Jadran Lenarčič
Michael M. Stanišić
Editors

Advances in Robot Kinematics

Motion in Man and Machine

Springer

Advances in Robot Kinematics: Motion in Man and Machine

Jadran Lenarčič • Michael M. Stanišić
Editors

Advances in Robot Kinematics: Motion in Man and Machine

Springer

Editors
Jadran Lenarčič
J. Stefan Institute
University of Ljubljana
Ljubljana, Slovenia
jadran.lenarcic@ijs.si

Michael M. Stanišić
Aerospace and Mechanical Engineering
University of Notre Dame
Fitzpatrick Hall 365
Notre Dame, IN 46556-5637
USA
stanisic.1@nd.edu

ISBN 978-90-481-9261-8 e-ISBN 978-90-481-9262-5
DOI 10.1007/978-90-481-9262-5
Springer Dordrecht Heidelberg London New York

Library of Congress Control Number: 2010928344

Table of Contents

Preface

The first International Meeting of Advances in Robot Kinematics, ARK, occurred in September 1988, by invitation to Ljubljana, Slovenia, of a group of 20 internationally recognized researchers, representing six different countries from three continents. There were 22 lectures and approximately 150 attendees. This success of bringing together excellent research and the international community, led to the formation of a Scientific Committee and the decision to repeat the event biannually. The meeting was made open to all individuals with a critical peer review process of submitted papers. The meetings have since been continuously supported by the Jožef Stefan Institute and since 1992 have come under patronage of the International Federation for the Promotion of Mechanism and Machine Science (IFToMM). Springer published the first book of the series in 1991 and since 1994 Kluwer and Springer have published a book of the presented papers every two years.

The papers in this book present the latest topics and methods in the kinematics, control and design of robotic manipulators. They consider the full range of robotic systems, including serial, parallel and cable driven manipulators, both planar and spatial. The systems range from being less than fully mobile to kinematically redundant to overconstrained. The meeting included recent advances in emerging areas such as the design and control of humanoids and humanoid subsystems, the analysis, modeling and simulation of human body motion, the mobility analysis of protein molecules and the development of systems which integrate man and machine.

We are grateful to the authors for their contributions and to a large team of reviewers for their critical and insightful recommendations. We are indebted to the staff of Springer and to Ms. Jolanda Karada (Karada Publishing Services) who brought this book together.

Jadran Lenarčič and Michael Stanišić, editors

PART 1

Calibration and Validation of a Rigid Body Kinematic Model of Flexure Hinges

R.J. Ellwood, D. Schütz, Annika Raatz and J. Hesselbach

Institute of Machine Tools and Production Technology, Braunschweig, Germany; e-mail: {j.ellwood, d.schuetz, a.raatz, j.hesselbach}@tu-bs.de

Abstract. In order to improve the precision of rigid body based kinematic models of flexure hinges, a review of applicable models is conducted. From these, a modification of the method proposed by Kimball and Tsai is proposed to decrease the number of calibration parameters while maintaining its versatility. The new model's validity is tested by first calibrating it using data obtained from a flexure hinge test bench. It is then shown that this modified method is able to replicate the actual path traversed by a flexure hinge, while decreasing the model and calibration complexity. A chart of the four methods is then used to allow a qualitative and quantitative comparison of the four different rigid body kinematic models of flexure hinges.

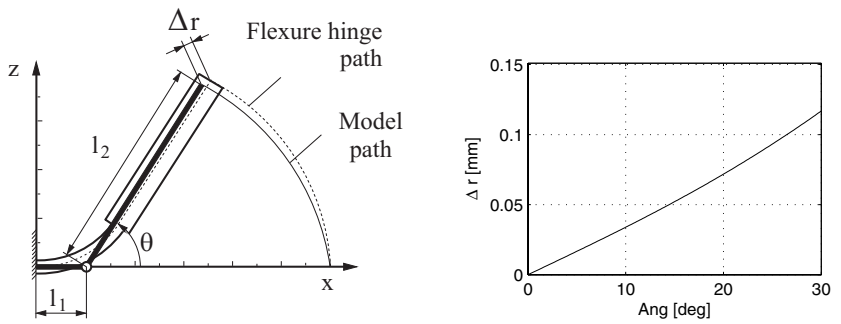
Key words: Kinematic model, flexure hinge, compliant mechanism, calibration.

1 Introduction

In order to tackle the complex tasks within precision engineering, the field of robotics is in a constant state of growth. One method to improve the precision of robots is through the elimination of the joint play found in typical passive joints. This can be achieved through the introduction of compliant mechanisms such as flexure hinges. Although extensive research has been done on the design of such elements, this research often overlooks the effects on the resulting kinematic model.

In most cases the kinematic model is either achieved using an idealized rigid body model or in a complex deflection model. In the case of the idealized rigid body model, it has been shown that this model is acceptable for small angles [1]. As can be seen in Figure 1(a), the error **r** increases as a function of the magnitude of the deflection angle. Here the error is found to be the difference between the two dimensional path **x** of the model and the actual system. In comparison to these, complexer mechanics based models have been shown to produce acceptable models, such as that shown in [2, 3]. Although these models are more accurate at larger angles, they are dependent on the material properties, manufacturing tolerances, and are substantially more complex.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_1, © Springer Science+Business Media B.V. 2010



(a) Flexure hinge deflection

(b) Simplified model error

Fig. 1 The deflection of a flexure hinge shown with the magnitude of the simplified model error.

$$\mathbf{r} = \mathbf{x}_{model} - \mathbf{x}_{actual}. \tag{1}$$

The goal of this paper is thus to find a rigid body kinematic model of a flexure hinge, which is simple yet still offers minimal error at large deflections. Here the emphasis is made on a simple flexure hinge model, as it is intended to be incorporated into the kinematic model of a robot. To accomplish this, a review of different rigid body models of compliant mechanisms which can be applied to the given flexure hinges is performed. This starts with an idealized model, then that of a leaf spring, followed by a method proposed by Kimball and Tsai [4], and ultimately a modified version of the method proposed by Kimball and Tsai. For each of the given methods, the kinematic equations which represent the end of a single mechanism are presented with respect to the x and z displacements.

It is then argued that although the original Kimball–Tsai method is able to produce acceptable results in the desired range, the introduction of 2 parameters per 1 degree of freedom (DOF) joint substantially increase the calibration task for larger models. In order to overcome this, the modified model holds one of the two parameters constant, reducing the calibration task by 1 parameter per flexure hinge. The validation of this model is then achieved by first calibrating the model using data obtained from the flexure hinge test bench. These optimized parameters within the modified model are then shown to improve the kinematic model without a substantial increase in calibration parameters.

The flexure hinges within the scope of this paper have a typical notched design [5], and are made from a memory shape alloy composed of a copper aluminum nickel iron (CuAlNiFe) alloy. These joints have a maximal deflection of $\pm 30^\circ$, and testing has shown an average of 1.5×10^5 cycles to failure [6]. Each flexure hinge has a length of 15 mm, with a radius of 15 mm notch cut so that the axis of rotation is as close to the middle of the beam as possible. The thinest portion of this notch has a cross section of 0.15 mm.

2 Rigid Body Kinematic Models of Flexure Hinges

The simplest kinematic representation of a flexure hinge can be achieved through an idealized rotation about the center of rotation of a compliant mechanism. Although these equations have been shown to give acceptable results in cases where there is a small angular deflection [1], the accuracy fades as the deflection angles increase. For completeness it is said that the path that the end of the hinge will traverse is

$$x = l_1 + l_2 \cos(\theta), \tag{2}$$
$$z = l_2 \sin(\theta). \tag{3}$$

Within these equations, the end effector is a function of the angle of the joint θ, and the lengths l_1 and l_2 which correspond with the parameters depicted in Figure 1(a). As can be seen in Figure 1(b), this model losses accuracy at larger angles. In a typical calibration process, both of these lengths need to be found. As all of the following methods have these lengths, or two comparable lengths, they will be considered the minimal amount of parameters needed for a calibration process. Throughout the rest of the paper, a comparison with regards to how many additional parameters to these two parameters will be used.

2.1 Leaf Spring Model

Another method which stems from flat or leaf springs, has been shown to better approximate fillet type flexure hinges which are characterized by a large thin section [7]. This model acts similar to two balls rolling against each other, with the angle representing the point of tangential interaction. Extending the model of the interaction presented in [8], results in

$$x = \frac{l_e}{\theta^2}(1 - \cos(\theta)) + l_2 \cos(\theta) + l_1, \tag{4}$$
$$z = \frac{l_e}{\theta^2}(\theta - \sin(\theta)) + l_2 \sin(\theta). \tag{5}$$

Within this equation, the length of the elastic region is said to be l_e. The length l_1 represents the region before the joint, while l_2 the region after the joint. Here the optimal length of the elastic portion of the model can be found as an additional parameter in the calibration process.

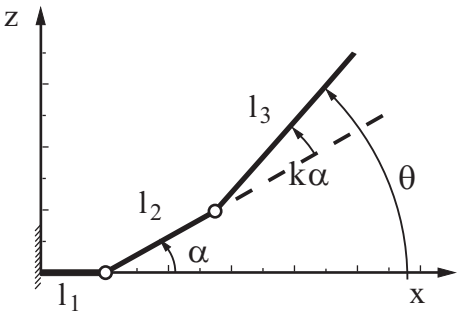


Fig. 2 Kimball–Tsai kinematic model.

2.2 Kimball–Tsai Method

Another method has been proposed by Kimball and Tsai, in which an 'equivalent' rigid body mechanism has been developed [4]. When this model (Figure 2) is compared to that of the initial model, there are two main differences. First, there is an additional length which is depicted in Figure 2 as l_2, which is the result of an additional pivot. In conjunction with this, the angles of the middle length l_2 and the end length l_3 are coupled by an angular correction factor k. This in turn causes the total angle θ to be a function of this factor and the intermediate angle α, as can be seen in Figure 2.

$$\theta = (1+k)\alpha. \tag{6}$$

The resulting kinematic model of the end effector of the flexure hinge are

$$x = l_1 + l_2\cos(\alpha) + l_3\cos(\theta), \tag{7}$$

$$z = l_2\sin(\alpha) + l_3\sin(\theta). \tag{8}$$

Although it is possible to extend this concept to the kinematic model of a robot, the additional length and angular correction factor must be found through a calibration process. In the case of a 1 DOF joint, the application of this method will result in 2 additional parameters, 1 length and 1 angular correction values, per joint. In the 2 DOF case, there will still be 1 length but 2 angular correction values. It can be seen that applying this method to a robot with many different passive joints, will cause a substantial increase in the number of parameters which need to be found through the calibration process.

2.3 Modified Kimball–Tsai Method

In order to limit the number of calibration parameters, a modification of the method proposed by Kimball has been developed in which the angular correction factor is

assumed to be $k = 1$. This modification reduces the overall parameters of the calibration process to one length parameter per joint, which is independent of number of DOF the joint has. This reduced model can then be said to be

$$x = l_1 + l_2 \cos(\alpha) + l_3 \cos(\theta), \tag{9}$$

$$z = l_2 \cos(\alpha) + l_3 \sin(\theta), \tag{10}$$

where

$$\theta = 2\alpha. \tag{11}$$

Although this modification decreases the number of calibration parameters, there is a decrease in the equations flexibility in comparison to the original equations. In order to quantitatively say how well this modified method is able to model the actual system, it will be verified against data obtained from a flexure hinge test bench. This is accomplished by first calibrating the model, then comparing the results. With a calibrated model, a quantitative comparison of how well each model is able to follow the system can be performed. As both the flat spring model and Kimball–Tsai method have already been validated, this process will not be repeated [4, 7].

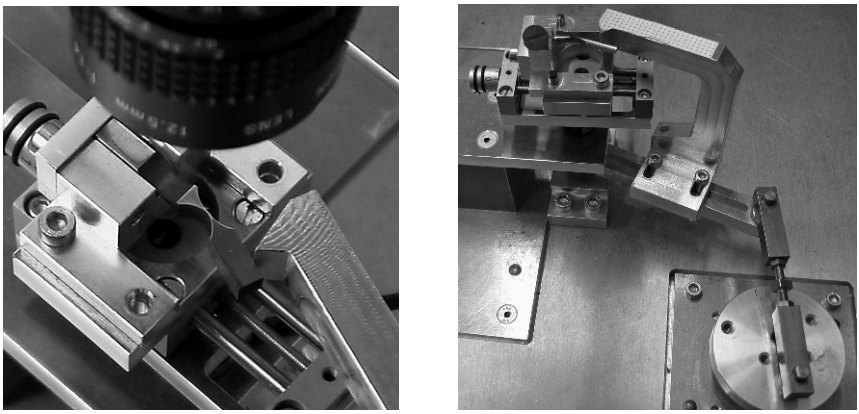
3 Flexure Hinge Test Bench

Collecting explicit kinematic data of a flexure hinge is no trivial task, as their deflection is dependent on the applied moment and loads. Such data is collected using the test bench designed within the institute, which can be seen in Figure 3. Here the flexure hinge is clamped in place while a triangular member applies a force perpendicular to the joint (Figure 3(a)). The task of keeping the applied force perpendicular to the hinge is achieved by a four bar linkage, with the joint of one member being placed as close to the middle of the flexure hinge as possible (Figure 3(b)). In the configuration shown, it is possible to flex the hinge in a cyclic manor between 0 and 30 degrees. Driving this four bar linkage is a stepper motor, which is attached the circular member in the bottom right corner in Figure 3(b).

As the joint is flexed, motion data is obtained from a machine vision system OPC (Optical Position Control) which has been developed at the Institute for Production Measurement Technology (IPROM) at the Technische Universität Braunschweig. With a camera placed perpendicularly above the joint, the 782×582 pixel sensor in conjunction with the OPC software is able to obtain a resolution of 0.1 μm [9].

4 Parameter Calibration of a Flexure Hinge

With the modified Kimball–Tsai model as well as motion data of the flexure hinge between 0 and 30 degrees, it is now possible to find the length parameter needed to test the models validity. This is achieved by first defining the parameter vector **k** as



(a) Close up of flexure hinge being actuated (b) Test bench linkage

Fig. 3 The flexure hinge test bench.

$$\mathbf{k} = \left[l_1, l_2, l_3\right]^T. \tag{12}$$

Next the residuum $\mathbf{r}(\mathbf{k})$ between the modeled data as a function of the parameter vector and the actual data is defined in equation (13). The cost function F is then developed as a function of the residuum, as seen in equation (14).

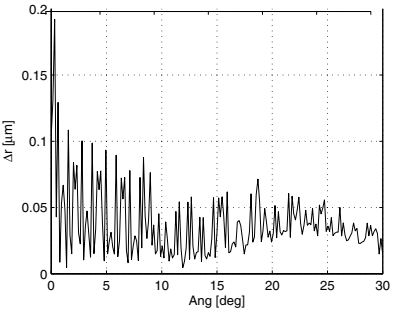
$$\mathbf{r}(\mathbf{k}) = \mathbf{x}_{model}(\mathbf{k}) - \mathbf{x}_{actual}, \tag{13}$$

$$F = \mathbf{r}^T \cdot \mathbf{r}. \tag{14}$$

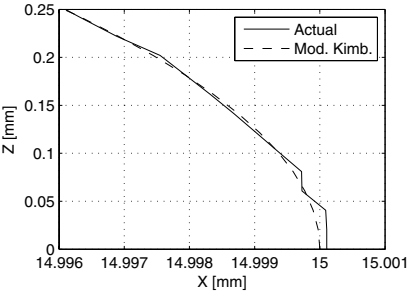
In order to find the optimized parameters, the cost function is minimized, which is achieved using the well documented Levenberg-Marquardt method [10].

5 Validation and Model Comparison

The parameters of the simplified rigid body model presented in Section 2.3 are calculated using the above mentioned calibration process. This resulted in l_1 being 6.5969 mm, l_2 being 1.8437 mm, and l_3 being 6.5594 mm. The error between the actual data and the model is found and can be seen in Figure 4(a). Within the error plot it can be seen that there is a proportionally large error at the beginning. Upon further investigation of this abnormality, it can be seen that there is a slight bump in the data found using the test bench. This can best be seen in the plot in Figure 4(b), in which both the modeled and actual paths have been plotted together. Other than this first peak, the error is on the order of the optical measurement system, 0.1 μm.



(a) Error between modified Kimball–Tsai method and actual data



(b) Plot of actual and modeled path zoomed in on area of interest

Fig. 4 Error between the modified Kimball–Tsai method and actual data.

Table 1 Comparing the rigid body kinematic models.

Models	Simp.	Flat-Spring	Kimb. Tsai	Mod. Kimb.
Complexity	++	- -	-	+
Calib. **k**				
1 DOF Joint	0	1	2	1
2 DOF Joint	0	1	3	1
Error **r** 5°	31 μm	17 μm	≤ 0.1 μm	≤ 0.1 μm
Error **r** 20°	128 μm	72 μm	≤ 0.1 μm	≤ 0.1 μm
Error **r** 30°	192 μm	117 μm	≤ 0.1 μm	≤ 0.1 μm

A comparison of the presented rigid body kinematic models can be seen in Table 1, in which a qualitative as well as quantitative comparison is conducted. For the qualitative comparison, $++$ is considered advantageous, while $--$ is considered less favorable. This table also shows the number of additional parameters needed within the calibration process, where it is said that at least two length parameters are needed.

6 Conclusion and Outlook

From the four presented kinematic models of flexure hinges, it is shown that the method proposed by Kimball and Tsai can be modified so that the angular correction value is assumed to be constant. This modified model reduces the number of additional parameters required in its calibration from 2 to 1 for a 1 DOF joint, and from 3 to 1 in a 2 DOF joint. This model is then calibrated with the Levenberg-Marquardt method using hinge data obtained from the flexure hinge test bench.

Lastly, the calibrated model along with the other models are compared to data from the actual system. It is thus shown that the reduced order model is able to reduce the error of the model for both small to large displacements. During model validation, it is shown that the model error is less then that of the sensor resolution of 0.1μm.

The validation of the single flexure hinge model using the modified Kimball–Tsai method has shed some light on what has often been an overlooked topic. This validation thus opens the door to applying this modified model to more complex systems. It is also important to mention that the chosen force loading shown within this paper is rather ideal. Although this is not the case within an actual robot system in which the loads and moments will be changing as a function of time, it is foreseen that an improvement in the calibrated kinematic model at increased angular deflections can be obtained. It is also anticipated that the resilience of the modified model to different loading scenarios will be dependent on the hinge stiffness.

Acknowledgements The research reported here has been supported by the German Research Foundation (DFG) within the scope of both the Collaborative Research Center SFB 516 and 562.

References

1. Howell, L. L.: *Compliant Mechanisms*. A Wiley-Interscience Publication, Wiley, New York (2001).
2. Lobontiu, N.: *Compliant Mechanisms: Design of Flexure Hinges*. CRC Press, Boca Raton, FL (2003).
3. Niaritsiry, T. F., Fazenda, N., Clavel, R.: Study of the sources of inaccuracy of a 3 dof flexure hinge-based parallel manipulator. In: *Proceedings of Robotics and Automation*, pp. 4091–4096, IEEE Operations Center, Piscataway, NJ (2004).
4. Kimball, C., Tsai, L. W.: Modeling of flexural beams subjected to arbitrary end loads. *Journal of Mechanical Design*, 223–235 (2002).
5. Paros, J. M., Weisbord, L.: How to design flexure hinges. *Machine Design*, 151–156 (1965).
6. Raatz, A., Hesselbach, J.: Performance of pseudo-elastic flexure hinges in parallel robots for micro-assembly tasks. *CIRP Annals – Manufacturing Technology*, **53**(1), 329–332 (2004).
7. Raatz, A.: Stoffschlüssige Gelenke aus pseudo-elastischen Formgedächtnislegierungen in Pararellrobotern. Techn. Univ., Diss., Braunschweig. Vulkan-Verl., Essen (2006).
8. Wuest, W.: Blattfedergelenk für Meßgeräte. Feinwerktechnik, **54**(7), 167–170 (1950).
9. Hesselbach, J., Ritter, R., Thoben, R., Reich, C., Pokar, G.: Visual control and calibration of parallel robots for micro assembly. In: *Microrobotics and Micromanipulation*, Vol. 3519, pp. 50–61, SPIE, Bellingham, WA (1998).
10. Scales, L. E.: *Introduction to Non-Linear Optimization*. Springer-Verlag, New York (1985).

Dynamic Jacobian Inverses of Mobile Manipulator Kinematics

K. Tchoń, J. Jakubiak and Ł. Małek

Institute of Computer Engineering, Control and Robotics,
Wrocław University of Technology, Wrocław, Poland;
e-mail: {krzysztof.tchon, janusz.jakubiak, lukasz.malek}@pwr.wroc.pl

Abstract. We derive a class of dynamic Jacobian inverses of mobile manipulator kinematics based on the Ważewski inequality. It is shown that all right Jacobian inverses, as well as the adjoint Jacobian, the adjugate dexterity matrix, and the singularity robust Jacobian inverse belong to this class. Performance of example Jacobian algorithms is illustrated by computer simulations.

Key words: Mobile manipulator, inverse kinematics, Jacobian, dynamic inverse.

1 Introduction

The idea of using a dynamic system in order to solve various problems in control theory is well known, to mention only dynamic observers or dynamic inverters of linear control systems. An extension of this idea to the inversion of nonlinear maps using state observers has been made in [1, 2]. A complementary approach, focused on the trajectory tracking in nonlinear control systems, and the continuous inverse kinematic problem in holonomic manipulators is set forth in [3]. An inspiring possibility of applying dynamical systems for solving problems of "discrete" nature has been discovered in [4], and developed in [5]. The dynamic Jacobian pseudo inverse has been used for motion planning of nonholonomic systems in [6] and [7].

This paper develops a construction of dynamic Jacobian inverses of kinematics of mobile manipulators, based on the Ważewski inequality [8]. The continuation method approach is taken as a point of departure. It is shown that not only the classical right Jacobian, but also the adjoint Jacobian, the adjugate dexterity matrix, and the singularity robust inverse Jacobian inverses belong to this class. The existence of global dynamic inverses, and the error convergence have been considered.

The paper is composed as follows. Section 2 defines the basic concepts including the kinematics, the Jacobian and the adjoint Jacobian, regular and singular configuration, and the continuation method. The concept of the dynamic Jacobian inverse is introduced in Section 3. Section 4 provides examples of the dynamic inverses. Computer simulations are included in Section 5. Section 6 contains conclusions.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_2, © Springer Science+Business Media B.V. 2010

2 Basic Concepts

We shall assume that a mobile manipulator consists of a nonholonomic mobile platform carrying on board a holonomic manipulator. The motion of the mobile platform is described by coordinates $q \in R^n$, and subject to nonholonomic phase constraints in the Pfaffian form. The joint positions of the on board manipulator are denoted by $x \in R^p$, and the end effector position and orientation are given by $y \in R^r$. In consequence, the kinematics of the mobile manipulator will be represented by a driftless control system with outputs

$$\begin{cases} \dot{q} = G(q)u = \sum_{i=1}^{m} g_i(q)u_i, \\ y = k(q,x) = (k_1(q,x), \ldots, k_r(q,x)). \end{cases} \tag{1}$$

2.1 Kinematics

The inputs $(u(\cdot),x)$ driving the system (1) include platform velocities $u(t) \in R^m$, and joint positions $x \in R^p$ of the on board manipulator. The platform control functions are chosen Lebesgue square integrable on an interval $[0,T]$; the joint positions are constant vectors. The control space $\mathscr{X} = L^2_m[0,T] \times R^p$ equipped with inner product

$$\langle (u_1(\cdot),x_1),(u_2(\cdot),x_2)\rangle_{RW} = \int_0^T u_1^T(t)R(t)u_2(t)dt + x_1^T W x_2, \tag{2}$$

where $R(t) = R^T(t) > 0$ and $W = W^T > 0$ are symmetric, positive definite weight matrices, becomes a Hilbert space called the endogenous configuration space of the mobile manipulator [9]. In what follows, wherever convenient, we shall use the notation $\mathrm{x} = (u(\cdot),x)$.

The output map of (1) takes values in the task space of the mobile manipulator, identified with R^r, and endowed with inner product

$$\langle y_1, y_2\rangle_Q = y_1^T Q y_2, \tag{3}$$

where Q denotes a symmetric, positive definite matrix $Q^T = Q$, $Q > 0$. The norm induced by (3) will be denoted by $||\cdot||_Q$.

Given an initial platform posture $q_0 \in R^n$, and an endogenous configuration $\mathrm{x} = (u(\cdot),x) \in \mathscr{X}$, we compute the trajectory $q(t) = \varphi_{q_0,t}(u(\cdot))$ of the platform, and the task space trajectory $y(t) = k(q(t),x)$. The end point map $K_{q_0,T} : \mathscr{X} \longrightarrow R^r$ of the control system (1), defined as

$$K_{q_0,T}(\mathrm{x}) = y(T) = k(\varphi_{q_0,T}(u(\cdot)),x), \tag{4}$$

will be identified with the kinematics of the mobile manipulator.

2.2 Jacobian

The differential kinematics are constituted by the mobile manipulator's Jacobian $J_{q_0,T}(\mathrm{x}) : \mathscr{X} \longrightarrow R^r$, defined as

$$J_{q_0,T}(\mathrm{x})(v(\cdot),w) = \frac{d}{d\alpha}\Big|_{\alpha=0} K_{q_0,T}(u(\cdot)+\alpha v(\cdot), x+\alpha w)$$
$$= C(T,x)\int_0^T \Phi(T,s)B(s)v(s)ds + D(T,x)w. \tag{5}$$

The matrices appearing in the Jacobian result from the linear approximation of the kinematics representation (1) along the pair (input, trajectory) $(u(t),x,q(t))$, so that

$$A(t) = \frac{\partial\big(G(q(t))u(t)\big)}{\partial q}, \quad B(t) = G(q(t)),$$
$$C(t,x) = \frac{\partial k(q(t),x)}{\partial q}, \quad D(t,x) = \frac{\partial k(q(t),x)}{\partial x}, \tag{6}$$

while $\Phi(t,s)$ denotes the transition matrix satisfying the evolution equation $\frac{\partial}{\partial t}\Phi(t,s) = A(t)\Phi(t,s)$, $\Phi(s,s) = I_n$. The linear, time dependent control system

$$\dot{\xi} = A(t)\xi + B(t)v, \quad \eta = C(t,x)\xi + D(t,x)w \tag{7}$$

determined by the data (6) will be referred to as the variational system associated with (1). It is easily seen that the Jacobian (5) corresponds to the input-output map of the variational system initialized at $\xi_0 = 0$.

2.3 Adjoint Jacobian

The dual map $J^*_{q_0,T}(\mathrm{x}) : \mathscr{X}^* \to (R^r)^*$ to the Jacobian is called the adjoint Jacobian of the mobile manipulator. Having identified the original linear spaces with their duals, and using the inner products (2) and (3), we obtain

$$\big(J^*_{q_0,T}(\mathrm{x})\eta\big)(t) = \big[R^{-1}(t)B^T(t)\Phi^T(T,t)C^T(T,x),\ W^{-1}D^T(T,x)\big]Q\eta. \tag{8}$$

2.4 Regular and Singular Configurations

An endogenous configuration $\mathrm{x} = (u(\cdot),x) \in \mathscr{X}$ of the mobile manipulator will be called regular, if the Jacobian (5) is surjective, otherwise the configuration is singular. The surjectivity of the Jacobian is equivalent to the output controllability of the

variational system (7), and implies that the original system (1) is locally output controllable. A necessary and sufficient condition for the regularity of a configuration is that the Gram matrix

$$\mathscr{D}_{q_0,T}(\mathrm{x}) = J_{q_0,T}(\mathrm{x})J^*_{q_0,T}(\mathrm{x}) = D(T,x)W^{-1}D^T(T,x) + C(T,x)\int_0^T \Phi(T,s)B(s)R^{-1}(s)B^T(s)\Phi^T(T,s)dsC^T(T,x) \quad (9)$$

of the variational system has rank r. In robotic terminology the matrix (9) is called the dexterity matrix of the mobile manipulator.

2.5 Continuation Method

Given a desirable output y_d, the inverse kinematic problem consists in determining an endogenous configuration $\mathrm{x}_d = (u_d(\cdot), x_d)$, such that $K_{q_0,T}(\mathrm{x}_d) = y_d$. A convenient way of solving inverse problems comes from the continuation method [6, 10]. In accordance with this method, for a parameter $\theta \in R$ and a $\gamma > 0$, we compose a homotopy map

$$\begin{aligned} H(\theta,\mathrm{x}) &= (1-\exp(-\gamma\theta))\big(K_{q_0,T}(\mathrm{x}) - y_d\big) + \exp(-\gamma\theta)\big(K_{q_0,T}(\mathrm{x}) - K_{q_0,T}(\mathrm{x}_0)\big) \\ &= K_{q_0,T}(\mathrm{x}) - \exp(-\gamma\theta)K_{q_0,T}(\mathrm{x}_0) - (1-\exp(-\gamma\theta))y_d, \end{aligned} \quad (10)$$

between the original problem (associated with $\theta = +\infty$) and a trivial problem: given x_0, find x such that $K_{q_0,T}(\mathrm{x}) = K_{q_0,T}(\mathrm{x}_0)$ (assigned to $\theta = 0$). By equating the homotopy map to 0, we get an implicitly defined curve x_θ of endogenous configurations, passing at $\theta = 0$ through x_0, that satisfies the following Ważewski–Davidenko differential equation [11, 12]

$$J_{q_0,T}(\mathrm{x}_\theta)\frac{d\mathrm{x}_\theta}{d\theta} = -\gamma\exp(-\gamma\theta)\big(K_{q_0,T}(\mathrm{x}_0) - y_d\big) = -\gamma\big(K_{q_0,T}(\mathrm{x}_\theta) - y_d\big). \quad (11)$$

Let $J^{\#}_{q_0,T}(\mathrm{x})$ denote any right inverse of the Jacobian, i.e. $J_{q_0,T}(\mathrm{x})J^{\#}_{q_0,T}(\mathrm{x}) = I_r$. Then, the equation (11) transforms into a Jacobian inverse kinematics algorithm with the underlying dynamics

$$\frac{d\mathrm{x}_\theta}{d\theta} = -\gamma J^{\#}_{q_0,T}(\mathrm{x}_\theta)\big(K_{q_0,T}(\mathrm{x}_\theta) - y_d\big). \quad (12)$$

A solution to the inverse kinematic problem is obtained as the limit

$$\mathrm{x}_d = (u_d(\cdot), x_d) = \lim_{\theta\to+\infty} \mathrm{x}_\theta. \quad (13)$$

Having defined the error as

$$e(\theta) = K_{q_0,T}(\mathrm{x}_\theta) - y_d, \tag{14}$$

it is easily seen that the Ważewski–Davidenko equation results in the error equation

$$\frac{de(\theta)}{d\theta} = -\gamma e(\theta), \tag{15}$$

and implies that the error converges to 0 exponentially with the rate γ.

3 Dynamic Inverses

Taking (15) as a point of departure, we shall request that the error dynamics obey a more general differential equation

$$\frac{d\,e(\theta)}{d\theta} = -\gamma S(\theta) e(\theta), \tag{16}$$

containing a certain $r \times r$ matrix $S(\theta)$ whose properties will be specified later. By differentiating the error formula (14), we get the Ważewski–Davidenko equation

$$J_{q_0,T}(\mathrm{x}_\theta)\frac{d\mathrm{x}_\theta}{d\theta} = -\gamma S(\theta) e(\theta). \tag{17}$$

Now, let us choose an operator $J^{\#S}_{q_0,T}(\mathrm{x}) : R^r \to \mathscr{X}$, and define the dynamic system

$$\frac{d\mathrm{x}_\theta}{d\theta} = -\gamma J^{\#S}_{q_0,T}(\mathrm{x}_\theta) e(\theta). \tag{18}$$

A substitution of (18) into the Ważewski–Davidenko equation yields the identity

$$S(\theta) = J_{q_0,T}(\mathrm{x}_\theta) J^{\#S}_{q_0,T}(\mathrm{x}_\theta). \tag{19}$$

Whenever a trajectory x_θ of (18) exists, the solution of the error equation (16) satisfies the Ważewski inequality [8]

$$||e(0)||_Q \exp\left(\int_0^\theta \underline{\lambda}_{M(\theta)} ds\right) \le ||e(\theta)||_Q \le ||e(0)||_Q \exp\left(\int_0^\theta \overline{\lambda}_{M(\theta)} ds\right), \tag{20}$$

with $\underline{\lambda}_M$ and $\overline{\lambda}_M$ denoting, respectively, the minimum and the maximum eigenvalue of the matrix

$$M(\theta) = -\frac{1}{2}\gamma\big(Q^{-1/2}(QS(\theta) + S^T(\theta)Q)Q^{-1/2}\big). \tag{21}$$

It is easily seen that, if x_θ is defined for every $\theta \ge 0$, and

$$\lim_{\theta\to+\infty}\int_0^{\theta}\overline{\lambda}_{M(s)}ds=-\infty,$$

then the error $e(\theta)$ vanishes asymptotically. When these conditions are met, the operator $J^{\#S}_{q_0,T}(\mathrm{x})$ will be called a dynamic inverse of the mobile manipulator's Jacobian. After plugging the dynamic inverse into the dynamic system (18), we transform this system into an inverse kinematics algorithm producing in the limit (13) a solution to the inverse kinematic problem.

By design, the dynamic inverse transforms an initial endogenous configuration $\mathrm{x}_0=(u_0(\cdot),x_0)$ into a solution $\mathrm{x}_d=(u_d(\cdot),x_d)$ of the inverse kinematic problem, exploiting as a vehicle the dynamics (18). In general, the dynamic inverse is defined only locally, for those endogenous configurations whose trajectories exist for all $\theta\geq 0$. This being so, a fundamental design problem of dynamic inverses consists in guaranteeing the completeness of (18), i.e. the existence of its solutions for every initial configuration x_0 and every $\theta\in R$.

4 Example Jacobian Inverses

In this section we present a collection of Jacobian dynamic inverses applicable to mobile manipulators, and demonstrate that diverse tools used for solving the inverse kinematic problem for mobile manipulators can be unified within the same concept of the dynamic Jacobian inverse. For the sake of space limitations we shall confine here only to very basic theoretical aspects of these inverses. Various computational issues have been addressed in [13–15].

4.1 Right Inverses

It has been established in [14] that every right inverse of the mobile manipulator Jacobian should have the following structure

$$J^{\#}_{q_0,T}(\mathrm{x})=L_{q_0,T}(\mathrm{x})R^{-1}_{q_0,T}(\mathrm{x}),\tag{22}$$

where $L_{q_0,T}(\mathrm{x}):R^r\to\mathscr{X}$ is a linear map, while $R_{q_0,T}(\mathrm{x})=J_{q_0,T}(\mathrm{x})L_{q_0,T}(\mathrm{x})$ denotes an $r\times r$ matrix. An endogenous configuration for which the matrix $R_{q_0,T}(\mathrm{x})$ has full rank will be referred to as an algorithmically regular configuration; when $R_{q_0,T}(\mathrm{x})$ becomes rank deficient, the corresponding configuration represents an algorithmic singularity. By design, every right Jacobian inverse is a dynamic inverse associated with the unit matrix $S(\theta)$, that is well defined outside algorithmic singularities. The convergence of the corresponding inverse kinematics algorithm is exponential.

Specifically, the Jacobian pseudo inverse $J^{\#P}_{q_0,T}(\mathrm{x}):R^r\to\mathscr{X}$ is defined as [9]

$$\left(J^{\#P}_{q_0,T}(\mathrm{x})\eta\right)(t) = \left[R^{-1}(t)B^T(t)\Phi^T(T,t)C^T(T,x), W^{-1}D^T(T,x)\right]\mathscr{D}^{-1}_{q_0,T}(\mathrm{x})\eta. \quad (23)$$

It is easily checked that the map $L_{q_0,T}(\mathrm{x}) = J^*_{q_0,T}(\mathrm{x})$, see (8). Consequently, $R_{q_0,T}(\mathrm{x}) = \mathscr{D}_{q_0,T}(\mathrm{x})$, therefore the algorithmic singularities of the Jacobian pseudo inverse coincide with Jacobian singularities. Only in a few specific cases the Jacobian pseudo inverse has been proved to be a global dynamic inverse [7]. Notice that all extended Jacobian inverses [13] also belong to the class of right inverses defined by (22).

4.2 Non-Right Inverses

4.2.1 Adjoint Jacobian

The adjoint Jacobian dynamic inverse $J^{\#*}_{q_0,T}(\mathrm{x}) = J^*_{q_0,T}(\mathrm{x}) : R^r \to \mathscr{X}$ has been defined by (8). Its associated matrix $S(\theta) = J_{q_0,T}(\mathrm{x})J^*_{q_0,T}(\mathrm{x}) = \mathscr{D}_{q_0,T}(\mathrm{x})Q$, and the convergence condition of the corresponding inverse kinematics algorithm can be formulated as

$$\lim_{\theta\to+\infty}\int_0^\theta \underline{\lambda}_{\mathscr{D}_{q_0,T}(\mathrm{x}_\alpha)Q}\, d\alpha = +\infty.$$

The adjoint Jacobian inverse can operate at singular configurations. A number of particular cases when the adjoint Jacobian becomes a global dynamic inverse have been examined in [15].

4.2.2 Adjugate Dexterity Matrix Inverse

This inverse comes from a generalization of the Newton method [16], and relies on the assumption that the kinematics (4) remain transverse to a ray in the task space [9]. The resulting dynamic inverse $J^{\#ADM}_{q_0,T}(\mathrm{x}) : R^r \to \mathscr{X}$ takes the form

$$\left(J^{\#ADM}_{q_0,T}(\mathrm{x})\eta\right)(t) = \left[R^{-1}(t)B^T(t)\Phi^T(T,t)C^T(T,x), W^{-1}D^T(T,x)\right]\mathrm{adj}\mathscr{D}_{q_0,T}(\mathrm{x})\eta,$$

where for an $n\times n$ matrix M, $\mathrm{adj}M$ denotes the adjugate (or adjoint) matrix, such that $\mathrm{adj}\,M\,M = M\,\mathrm{adj}\,M = I_n \det M$. The matrix $S(\theta) = J_{q_0,T}(\mathrm{x}_\theta)J^{\#ADM}_{q_0,T}(\mathrm{x}_\theta) = I_r \det\mathscr{D}_{q_0,T}(\mathrm{x}_\theta)$, whereas the error (14) converges, if

$$\lim_{\theta\to+\infty}\int_0^\theta \det\mathscr{D}_{q_0,T}(\mathrm{x}_\alpha)d\alpha = +\infty.$$

It can be shown that the adjugate dexterity matrix dynamic inverse vanishes at singular endogenous configurations. So far no globality conditions for this inverse are available.

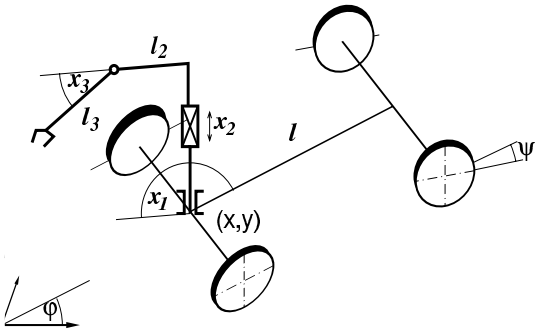


Fig. 1 Mobile manipulator.

4.2.3 Singularity Robust Inverse

The singularity robust Jacobian inverse $J^{\#SRI}_{q_0,T}(\mathrm{x}) : R^r \longrightarrow \mathscr{X}$, is defined as

$$\left(J^{\#SRI}_{q_0,T}(\mathrm{x})\eta\right)(t) = \left[R^{-1}(t)B^T(t)\Phi^T(T,t)C^T(T,x), W^{-1}D^T(T,x)\right]Q(\kappa I_r + \mathscr{D}_{q_0,T}(\mathrm{x})Q)^{-1}\eta,$$

where $\kappa > 0$ plays the role of a small regularizing parameter [9]. This inverse is well defined both at regular and at singular configurations. The associated matrix $S(\theta) = J_{q_0,T}(\mathrm{x}_\theta))J^{\#SRI}_{q_0,T}(\mathrm{x}_\theta) = \mathscr{D}_{q_0,T}(\mathrm{x}_\theta)Q(\kappa I_r + \mathscr{D}_{q_0,T}(\mathrm{x}_\theta)Q)^{-1}$. The error convergence condition takes the following form:

$$\lim_{\theta\to+\infty}\int_0^\theta \frac{\underline{\lambda}_{\mathscr{D}_{q_0,T}(\mathrm{x}_\alpha)Q}d\alpha}{\kappa + \underline{\lambda}_{\mathscr{D}_{q_0,T}(\mathrm{x}_\alpha)Q}} = +\infty.$$

Recently, it has been proved that the singularity robust inverse is a global dynamic Jacobian inverse [17, 18].

5 Computer Simulations

For illustration of theoretical developments, below we shall show sample results of solving an inverse kinematic problem for the mobile manipulator composed of kinematic car platform and RTR on-board manipulator, studied in [18], and displayed in Fig. 1.

Variable $q = (\mathrm{x}, \mathrm{y}, \varphi, \psi) \in R^4$ describes the platform position, orientation, and the heading angle of its front wheels. Variables $x = (x_1, x_2, x_3) \in R^3$ and $y = (y_1, y_2, y_3) \in R^3$ refer to the joint and to the end effector position of the on board manipulator. Under assumption $l = l_2 = l_3 = 1$ the control system (1) takes the following form

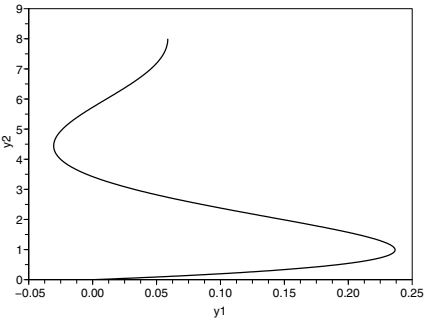


Fig. 2 Jacobian pseudo inverse: end effector path in the plane

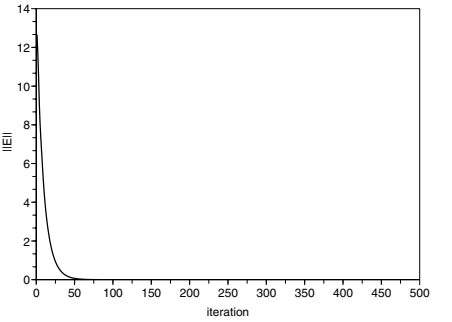


Fig. 3 Jacobian pseudo inverse: end effector error.

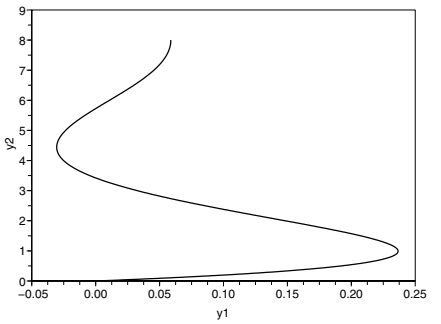


Fig. 4 Singularity robust inverse: end effector path in the plane.

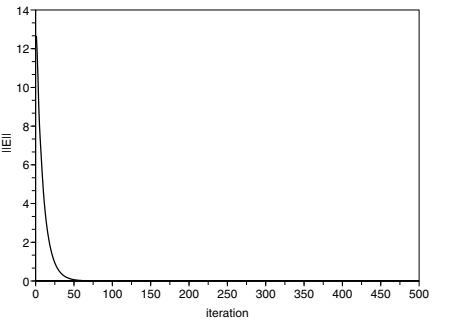


Fig. 5 Singularity robust inverse: end effector error.

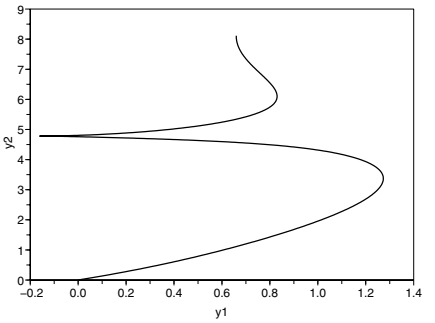


Fig. 6 Adjoint Jacobian inverse: end effector path in the plane.

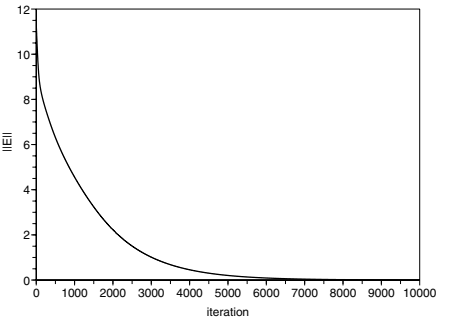


Fig. 7 Adjoint Jacobian inverse: end effector error.

$$\begin{cases} \dot{q}_1 = u_1 \cos q_3 \cos q_4, \;\; \dot{q}_2 = u_1 \sin q_3 \cos q_4, \;\; \dot{q}_3 = u_1 \sin q_4, \;\; \dot{q}_4 = u_2 \\ y = (q_1 + (1+\cos x_3)\cos(x_1+q_3), q_2 + (1+\cos x_3)\sin(x_1+q_3), x_2 + \sin x_3). \end{cases}$$

The Jacobian pseudo inverse, the singularity robust Jacobian inverse, and the adjoint Jacobian algorithms will be used. The inverse kinematic problem consists in reaching $y_d = (0,0,0)$ from initial platform's state $q_0 = (0,10,0,0)$ at initial configuration $x_0 = (1,1,1)$ of the on board manipulator, and over the time horizon $T = 1$. The control $u(t) = (u_1(t), u_2(t))$ has been synthesized from 500 element basis consisting of piecewise constant functions. Parameter γ is set to 0.1 for Jacobian pseudo inverse and singularity robust Jacobian inverse, and 0.0001 for adjoint Jacobian. Where applicable, $\kappa = 10^{-6}$ is adopted. Results of computations are displayed in Figs. 2–7.

6 Conclusions

This paper deduces a collection of Jacobian inverse kinematics algorithms for mobile manipulators from a common root of the dynamic Jacobian inverse. The presented results make a contribution to the theory of motion planning for mobile manipulators. Computer simulations illustrate performance and convergence of example algorithms based on right and non-right Jacobian inverses. In the paper we have defined the dynamic inverse by reference to the Ważewski inequality that provides a sufficient condition for the convergence of the inverse kinematics algorithms. It is a challenge of future research to seek for alternative, more general conditions.

Acknowledgements The research work reported here was supported by a statutory grant from Wrocław University of Technology. The authors are indebted to anonymous reviewers whose comments significantly influenced the final form of this paper.

References

1. Nicosia, S., Tornambé, A., Valigi, P.: A solution to the generalized problem of nonlinear map inversion. *Syst. Contr. Lett.*, **17**, 383–394 (1991).
2. Nicosia, S., Tornambé, A., Valigi, P.: Nonlinear map inversion via state observers. *Circuits, Syst. Signal Process.*, **13**, 571–589 (1994).
3. Getz, N. H.: Dynamic inversion of nonlinear maps with applications to nonlinear control and robotics, Doctoral Dissertation, UCB (1995).
4. Brockett, R. W.: Dynamical systems that sort lists, diagonalize matrices and solve linear programming problems. *Linear Algebra Appl.*, **146**, 79–91 (1991)
5. Helmke, U., Moore, J. B.: *Optimization and Dynamical Systems*. Springer-Verlag, Berlin (1994).
6. Sussmann, H. J.: A continuation method for non-holonomic path finding problems. In: *Proc. 32nd IEEE CDC*, pp. 2718–2723 (1993).
7. Chitour, Y.: A homotopy continuation method for trajectories generation of non-holonomic systems. *ESAIM: Control, Optim. Calc. Var.*, **12**, 139–168 (2006).

8. Ważewski, T.: Sur la limitation des intégrales des systemes d'équations différentielles linéaires ordinaires. *Studia Math.*, **10**, 48–59 (1948).
9. Tchoń, K., Jakubiak, J.: Endogenous configuration space approach to mobile manipulators: A derivation and performance assessment of Jacobian inverse kinematics algorithms. *Int. J. Control*, **76**, 1387–1419 (2003).
10. Richter, S. L., DeCarlo, R. A.: Continuation methods: Theory and applications. *IEEE Trans. Circuits Syst.*, **30**, 347–352 (1983).
11. Ważewski, T.: Sur l'évaluation du domaine d'existence des fonctions implicites réelles ou complexes. *Ann. Soc. Pol. Math.*, **20**, 81–120 (1947).
12. Davidenko, D. F.: On a new method of numerically integrating a system of nonlinear equations. *Dokl. Akad. Nauk SSSR*, **88**, 601–603 (1953).
13. Tchoń, K., Jakubiak, J.: A repeatable inverse kinematics algorithm with linear invariant subspaces for mobile manipulators. *IEEE Trans. Syst., Man, Cybern. Part B, Cybernetics*, **35**, 1051–1057 (2005).
14. Tchoń, K., Jakubiak, J., Małek, Ł.: A new inverse kinematics algorithm for nonholonomic mobile robots. In: *Proc. 2009 MMAR Conference*, IFAC Papers-Online, to appear.
15. Małek, Ł.: Convergence of Jacobian inverse kinematics algorithms based on the method of homotopy. Doctoral Dissertation, Wrocław University of Technology (2009) [in Polish].
16. Aubin, J.-P., Ekeland, I.: *Applied Nonlinear Analysis*. John Wiley, New York (1984).
17. Tchoń, K., Małek, Ł.: Singularity robust Jacobian inverse kinematics for mobile manipulators. In: J. Lenarcic, J., Wegner, Ph. (Eds.), *Advances in Robot Kinematics*, Springer, Dordrecht, pp. 155–164 (2008).
18. Tchoń, K., Małek, Ł.: On dynamic properties of singularity robust Jacobian inverse kinematics. *IEEE Trans. Autom. Control*, **54**, 1402–1406 (2009).

A Robust Forward Kinematics Analysis of 3-RPR Planar Platforms

Nicolás Rojas and Federico Thomas

Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain; e-mail: {nrojas, fthomas}@iri.upc.edu

Abstract. The standard forward kinematics analysis of 3-RPR planar parallel platforms boils down to computing the roots of a sextic polynomial. There are many different ways to obtain this polynomial but all of them include exceptions for which the formulation is not valid. Unfortunately, near these exceptions the corresponding polynomial exhibits numerical instabilities. In this paper, we provide a way around this inconvenience by translating the forward kinematics problem to be solved into an equivalent problem fully stated in terms of distances. Using constructive geometric arguments, an alternative sextic – which is not linked to a particular reference frame – is obtained in a straightforward manner without the need of variable eliminations nor tangent-half-angle substitutions. The presented formulation is valid, without any modification, for any planar 3-RPR parallel platform, including the special architectures and configurations – which ultimately lead to numerical instabilities – that cannot be directly handled by previous formulations.

Key words: 3-RPR parallel robots, coordinate-free formulations, Cayley–Menger determinants, bilateration, numerical stability.

1 Introduction

Much has been written about the 3-RPR planar parallel robot because of its practical interest, mechanical simplicity, and rich mathematical properties [1]. Such a robot consist of a moving platform connected to the ground through three revolute-prismatic-revolute kinematic chains. The prismatic joint of each chain is actuated and the forward kinematics problem consists in, given the prismatic joint lengths, calculating the Cartesian pose of the moving platform. This problem has at most six different solutions which are usually computed by solving a system of three non-linear equations in three unknowns [2, 3]. The major step in this resolution process is to find a so-called *univariate characteristic equation* in θ (the orientation of the moving platform), i.e., to eliminate all other variables from the system until an equation is obtained that contains only θ. Finally, a tangent-half-angle substitution is applied to translate sine and cosine functions of θ into rational polynomial expressions in a new variable $t = \tan(\theta/2)$. Merlet was probably one of the first researchers to observe that the computation of the roots of this 6th-degree leads,

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_3, © Springer Science+Business Media B.V. 2010

in some cases, to numerical instabilities and he also observed that the numerical robustness is improved by using a higher order polynomial [4].

The tangent-half-angle substitution poses two well-known problems. One results from the fact that $\tan(\theta/2)$ is undefined for $\pm\pi$. Moreover, it can become difficult to reconstruct other roots, occurring in conjunction with the root $\theta = \pm\pi$ [5]. The other problem is the introduction of extraneous roots. Both problems are well known and can be handled but it complicates notably subsequent calculations [6]. The use of the kinematic mapping, as in [7], avoids this substitution but the problem with $\pm\pi$ turns still remains if the used homogeneous coordinates are normalized to obtain the sextic polynomial. Other sources of numerical problems, specifically arising in 3-RPR parallel robots, are discussed in [8].

In order to simplify as much as possible the coefficients of the resulting 6th-degree polynomial, it is possible to express the coordinates of the base attachments according to a specific coordinate frame. For example, by making one coordinate axis to coincide with the baseline between two base attachments and/or locating the origin at one base attachment. Nevertheless, this kind of simplifications has an important drawback: the numerical conditioning of the resulting formulation depends on the chosen reference frame. This is why those formulations which are not linked to a particular reference frame – or coordinate-free formulations – are preferable.

In this paper, we provide a way around the above difficulties by casting the problem into an equivalent problem fully stated in terms of distances which is solved by a sequence of bilaterations. As a result, a 6th-degree characteristic polynomial, which is not linked to any particular reference frame, is obtained in a straightforward manner without variable eliminations nor tangent-half-angle substitutions. Moreover, the obtained polynomial is mathematically more tractable than the one obtained using other approaches because its coefficients are the result of operating with Cayley–Menger determinants with geometric meaning.

This paper is organized as follows. A coordinate-free formula for bilateration expressed in terms of Cayley–Menger determinants is presented in Section 2. It is the basic formula, used in Section 3, to derive a coordinate-free characteristic polynomial for the general 3-RPR planar parallel robot. Section 4 analyzes a numerical example in which the standard formulations miss a solution. Finally, Section 5 summarizes the main points and gives prospects for further research.

2 Cayley–Menger Determinants and Bilateration

Let P_i and $\mathbf{p}_i$ denote a point and its position vector in a given reference frame, respectively. Then, let us define

$$D(i_1,\ldots,i_n;\,j_1,\ldots,j_n) = \begin{vmatrix} 0 & 1 & \ldots & 1 \\ 1 & s_{i_1,j_1} & \ldots & s_{i_1,j_n} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & s_{i_n,j_1} & \ldots & s_{i_n,j_n} \end{vmatrix}, \tag{1}$$

with $s_{i,j} = \|\mathbf{p}_i - \mathbf{p}_j\|^2$, which is independent from the chosen reference frame. This determinant is known as the *Cayley–Menger bi-determinant* of the point sequences $P_{i_1}, \ldots, P_{i_n}$, and $P_{j_1}, \ldots, P_{j_n}$. When the two point sequences are the same, it will be convenient to abbreviate $D(i_1, \ldots, i_n; i_1, \ldots, i_n)$ by $D(i_1, \ldots, i_n)$, which is simply called the *Cayley–Menger determinant* of the involved points.

In terms of Cayley–Menger determinants, the squared distance between P_i and P_j can be expressed as $D(i, j)$ and the signed area[1] of the triangle $P_i P_j P_k$, as $\pm\frac{1}{2}\sqrt{D(i, j, k)}$. For a brief review of the properties of Cayley–Menger determinants, see [9].

The bilateration problem in $\mathbb{R}^2$ consists of finding the feasible locations of a point, say P_3, given its distances to two other points, say P_1 and P_2, whose locations are known. Then, according to Fig. 1, the position vector of the orthogonal projection of P_3 onto the line $P_1 P_2$ can be expressed as:

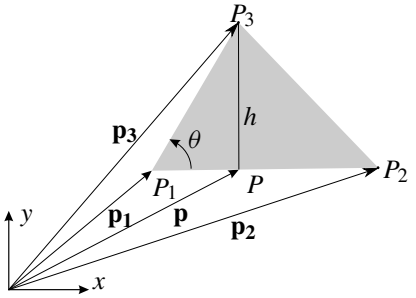


Fig. 1 The bilateration problem in $\mathbb{R}^2$.

$$\mathbf{p} = \mathbf{p}_1 + \sqrt{\frac{D(1,3)}{D(1,2)}}\cos\theta(\mathbf{p}_2 - \mathbf{p}_1) = \mathbf{p}_1 + \frac{D(1,2;1,3)}{D(1,2)}(\mathbf{p}_2 - \mathbf{p}_1). \tag{2}$$

Moreover, the position vector of P_3 can be expressed as:

$$\mathbf{p}_3 = \mathbf{p} \pm \frac{\sqrt{D(1,2,3)}}{D(1,2)}\mathbf{S}(\mathbf{p}_2 - \mathbf{p}_1), \tag{3}$$

where the $\pm$ sign accounts for the two mirror symmetric locations of P_3 with respect to the line defined by $P_1 P_2$, and $\mathbf{S} = \left[\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right]$. Then, substituting (2) in (3) and expressing the result in matrix form, we obtain

$$\mathbf{p}_3 = \mathbf{p}_1 + \mathbf{Z}(\mathbf{p}_2 - \mathbf{p}_1) \tag{4}$$

where

$$\mathbf{Z} = \frac{1}{D(1,2)}\begin{bmatrix} D(1,2;1,3) & \mp\sqrt{D(1,2,3)} \\ \pm\sqrt{D(1,2,3)} & D(1,2;1,3) \end{bmatrix}$$

[1] For a triangle $P_i P_j P_k$ in the Euclidean plane with area A, the *signed area* is defined as $+A$ (respectively, $-A$) if the point P_j is to the right (respectively to the left) of the line $P_i P_k$, when going from P_i to P_k.

3 Distance-Based Coordinate-Free Formulation

Figure 2 shows a general 3-RPR planar parallel robot. The center of the three grounded passive revolute joints define the base triangle $P_1P_2P_3$ and the three moving passive revolute joints centers, the moving triangle $P_4P_5P_6$. The active prismatic joint variables are the lengths ρ_1, ρ_2, and ρ_3.

Next, we derive a coordinate-free formula for the forward kinematics of this parallel platform. To this end, instead of directly computing the Cartesian pose of the moving platform, first we will compute the set of values of $T = \|\mathbf{p}_4 - \mathbf{p}_2\|^2$ compatible with ρ_1, ρ_2, and ρ_3 and the base and the moving platform side lengths, l_1, l_2, l_3, L_1, L_2, and L_3. Thus, this step is enterally posed in terms of distances.

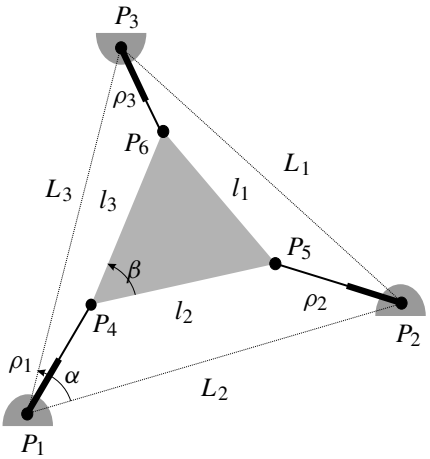


Fig. 2 A general planar 3-RPR parallel platform.

Let us take P_1 and P_2 as reference points. Then, by bilateration

1. $\mathbf{p}_3$ can be computed from $\mathbf{p}_1$ and $\mathbf{p}_2$. This has only one solution because the oriented area of triangle $P_1P_2P_3$ is constant and its sign is determined by the sign of α.
2. $\mathbf{p}_4$ can be computed from $\mathbf{p}_1$ and $\mathbf{p}_2$. This has two possible solutions which involve the unknown squared distance T.
3. $\mathbf{p}_5$ can be computed from $\mathbf{p}_4$ and $\mathbf{p}_2$. This also has two possible solutions which also involve the unknown squared distance T.
4. $\mathbf{p}_6$ can be computed from $\mathbf{p}_4$ and $\mathbf{p}_5$. This has only one solution because the oriented area of triangle $P_4P_5P_6$ is constant and its sign is determined by the sign of β.

Translating the above four bilaterations into algebraic terms using (4), the following system of vector linear equations is obtained:

$$\left.\begin{aligned} \mathbf{p}_3 &= \mathbf{p}_1 + \mathbf{A}(\mathbf{p}_2 - \mathbf{p}_1) \\ \mathbf{p}_4 &= \mathbf{p}_1 + \mathbf{B}(\mathbf{p}_2 - \mathbf{p}_1) \\ \mathbf{p}_5 &= \mathbf{p}_4 + \mathbf{C}(\mathbf{p}_2 - \mathbf{p}_4) \\ \mathbf{p}_6 &= \mathbf{p}_5 + \mathbf{D}(\mathbf{p}_4 - \mathbf{p}_5) \end{aligned}\right\}, \tag{5}$$

where

$$\mathbf{A} = \frac{1}{D(1,2)} \begin{bmatrix} D(1,2;1,3) & -\text{sign}(\alpha)\sqrt{D(1,2,3)} \\ \text{sign}(\alpha)\sqrt{D(1,2,3)} & D(1,2;1,3) \end{bmatrix},$$

$$\mathbf{B} = \frac{1}{D(1,2)} \begin{bmatrix} D(1,2;1,4) & \mp\sqrt{D(1,2,4)} \\ \pm\sqrt{D(1,2,4)} & D(1,2;1,4) \end{bmatrix},$$

$$\mathbf{C} = \frac{1}{D(4,2)} \begin{bmatrix} D(4,2;4,5) & \mp\sqrt{D(4,2,5)} \\ \pm\sqrt{D(4,2,5)} & D(4,2;4,5) \end{bmatrix},$$

$$\mathbf{D} = \frac{1}{D(5,4)} \begin{bmatrix} D(5,4;5,6) & \text{sign}(\beta)\sqrt{D(5,4,6)} \\ -\text{sign}(\beta)\sqrt{D(5,4,6)} & D(5,4;5,6) \end{bmatrix}.$$

Now, by properly manipulating the equations in the above linear system, it is possible to conclude that

$$(\mathbf{p}_6 - \mathbf{p}_3) = \boldsymbol{\Omega}(\mathbf{p}_2 - \mathbf{p}_1) \tag{6}$$

where $\boldsymbol{\Omega} = -\mathbf{A} + \mathbf{B} + \mathbf{C} - \mathbf{DC} - \mathbf{CB} + \mathbf{DCB}$. This matrix, when expanded in terms of Cayley–Menger determinants, leads to:

$$\boldsymbol{\Omega} = \frac{1}{D(5,4)D(4,2)D(1,2)} \begin{bmatrix} w_1 & -w_2 \\ w_2 & w_1 \end{bmatrix} \tag{7}$$

where

$$\begin{aligned} w_1 = {} & D(4,6;4,5)\left(\pm\sqrt{D(1,2,4)}\right)\left(\pm\sqrt{D(4,2,5)}\right) \\ & + D(4,2;4,5)\left(\text{sign}(\beta)\sqrt{D(5,4,6)}\right)\left(\pm\sqrt{D(1,2,4)}\right) \\ & - D(2,4;2,1)\left(\text{sign}(\beta)\sqrt{D(5,4,6)}\right)\left(\pm\sqrt{D(4,2,5)}\right) \\ & + D(4,6;4,5)D(2,4;2,1)D(4,2;4,5) \\ & + D(5,4)D(4,2)\left(D(1,2;1,4) - D(1,2;1,3)\right), \end{aligned}$$

and

$$\begin{aligned} w_2 = {} & \left(\text{sign}(\beta)\sqrt{D(5,4,6)}\right)\left(\pm\sqrt{D(1,2,4)}\right)\left(\pm\sqrt{D(4,2,5)}\right) \\ & + D(4,6;4,5)D(2,4;2,1)\left(\pm\sqrt{D(4,2,5)}\right) \\ & + D(4,2;4,5)D(2,4;2,1)\left(\text{sign}(\beta)\sqrt{D(5,4,6)}\right) \\ & - D(5,4)D(4,2)\left(\text{sign}(\alpha)\sqrt{D(1,2,3)}\right) \\ & + D(5,4)D(4,2)\left(\pm\sqrt{D(1,2,4)}\right) \\ & - D(4,2;4,5)D(4,6;4,5)\left(\pm\sqrt{D(1,2,4)}\right). \end{aligned}$$

Now, it can be checked that

$$\frac{\|\mathbf{p}_6 - \mathbf{p}_3\|^2}{\|\mathbf{p}_2 - \mathbf{p}_1\|^2} = \det(\mathbf{\Omega}).$$

Thus,

$$\frac{D(6,3)}{D(1,2)} = \frac{w_1^2 + w_2^2}{D^2(5,4)D^2(4,2)D^2(1,2)}, \tag{8}$$

which is a scalar equation in one unknown: T. Finally, by expanding all the involved Cayley–Menger determinants in the above equation in terms of distances, we obtain

$$\Phi_a + \Phi_b A_{124} + \Phi_c A_{425} + \Phi_d A_{124} A_{425} = 0 \tag{9}$$

where

$$A_{124} = \pm\frac{1}{2}\sqrt{\left((L_2+\rho_1)^2 - T\right)\left(T - (L_2-\rho_1)^2\right)}$$

and

$$A_{425} = \pm\frac{1}{2}\sqrt{\left((l_2+\rho_2)^2 - T\right)\left(T - (l_2-\rho_2)^2\right)}$$

are the signed areas of the triangles $P_1P_2P_4$ and $P_4P_2P_5$, respectively. Φ_a, Φ_b, Φ_c, and Φ_d are polynomials in T whose expressions can be found in the Appendix.

By properly twice squaring equation (9) to eliminate the two squared roots involving T, a 6th-degree polynomial in T is finally obtained. The roots of this polynomial determine the assembly-modes of the analyzed robot. Given the position vectors $\mathbf{p}_1$ and $\mathbf{p}_2$ in a particular reference frame, each root of this polynomial determines up to four possible values for $\mathbf{p}_6$ that can be obtained through a sequence of bilaterations. At least one of the values obtained from each of these sequences must satisfy, by construction, the distance constraint between P_3 and P_6 (i.e., $D(3,6) = {\rho_3}^2$). Observe that the sequence of bilaterations is not the same depending on whether the root is at the origin or not. Actually, it is important to realize that the obtained sextic permits to compute the solutions in which $T = 0$.

The presented formulation is coordinate-free because it is independent from the chosen reference frame. It is valid, without any modifications, for any planar 3-RPR parallel platform, including the special architectures discussed in [3] and [8] that cannot be directly solved by the formulation presented in [2], or that have been solved on an ad hoc basis [10].

4 Numerical Example

Let us study the planar 3-RPR parallel platform defined by $l_1^2 = 25$, $l_2^2 = 36$, $l_3^2 = 25$, $L_1^2 = 73$, $L_2^2 = 16$, $L_3^2 = 65$, $\alpha > 0$, and $\beta > 0$, with input variables

$\rho_1^2 = 1$, $\rho_2^2 = 121$, and $\rho_3^2 = 169$. If $\mathbf{p}_1 = (0, 0)^T$, $\mathbf{p}_2 = (4, 0)^T$, and $\mathbf{p}_3 = (1, 8)^T$, it can be verified that the characteristic polynomial of this mechanism, using the formulation derived in [2], reduces to:

$$1469440\,\Gamma^4 + 1755136\,\Gamma^3 + 4261376\,\Gamma^2 + 1140736\,\Gamma + 219136 = 0 \qquad (10)$$

with

$$\sin(\theta) = \frac{2\Gamma}{1+\Gamma^2} \quad \text{and} \quad \cos(\theta) = \frac{1-\Gamma^2}{1+\Gamma^2},$$

θ being the angle between the lines defined by P_1P_2 and P_4P_5. The solutions of equation (10), obtained using 32 decimal digits in the computations, are $-0.4573 - 1.5419i$, $-0.4573 + 1.5419i$, $-0.1399 - 0.1952i$, and $-0.1399 + 0.1952i$. Since none of them is real, it can be erroneously concluded that the platform under study cannot be assembled with the given leg lengths.

Alternately, using the formulation derived in [10], the following characteristic polynomial is obtained:

$$4408320Y^4 - 1744896Y^3 + 7788032Y^2 - 1464320Y + 3564544 = 0 \qquad (11)$$

where

$$\sin(\psi) = \frac{2Y}{1+Y^2} \quad \text{and} \quad \cos(\psi) = \frac{1-Y^2}{1+Y^2},$$

ψ being the angle between the lines defined by P_1P_4 and P_1P_2. The solutions of this equation obtained using 32 decimal digits in the computations, are $-0.0363 - 0.9243i$, $-0.0363 + 0.9243i$, $0.2342 - 0.9435i$, and $0.2342 + 0.9435i$. Again, since none of them is real, it can be erroneously concluded that the platform under study cannot be assembled with the given leg lengths thus confirming the results obtained using the formulation proposed in [2]. The formulation described in [7] leads to an analogous situation when one of the homogeneous coordinates is normalized to 1. Using the implementation for this formulation reported in [12], and choosing the moving reference frame such that $\mathbf{p}_4 = (0, 0)^T$ and $\mathbf{p}_5 = (6, 0)^T$ in it, the resulting polynomial is:

$$1469440\,{x_1}^4 + 1755136\,{x_1}^3 + 4261376\,{x_1}^2 + 1140736\,x_1 + 219136 = 0$$

where x_1 is a component of the kinematic image space coordinates. The roots of this equation are $-0.4573 - 1.5419i$, $-0.4573 + 1.5419i$, $-0.1399 - 0.1952i$, and $-0.1399 + 0.1952i$. Again, none of them is real. Nevertheless, substituting the parameters of the mechanism under study and the values of the input variables given above in the coordinate-free equation (9), the following characteristic polynomial is obtained:

$$B_6T^6 + B_5T^5 + B_4T^4 + B_3T^3 + B_2T^2 + B_1T^1 + B_0 = 0, \qquad (12)$$

where

$$B_6 = -3445000,\ B_5 = -185454160,\ B_4 = 304547661,$$
$$B_3 = 416912277408,\ B_2 = -32392769185400,$$
$$B_1 = 121165627563000,\ \text{and}\ B_0 = -15097679553125000.$$

The roots of this equation, calculated with the same numerical precision as above, are $-65.0725 - 41.0946i$, $-65.0725 + 41.0946i$, $13.1561 - 31.7921i$, $13.1561 + 31.7921i$, and a double root at 25.0000. It can be checked that the obtained double real root corresponds to a valid configuration of the analyzed 3-RPR parallel platform, in clear contradiction with what was concluded using the formulations proposed in [2, 10, 12]. In the platform pose associated with this double root, $\theta = 0$, $\psi = \pi$, and $\mathbf{p}_4 = (-1, 0)^T$. Figure 3 depicts this configuration.

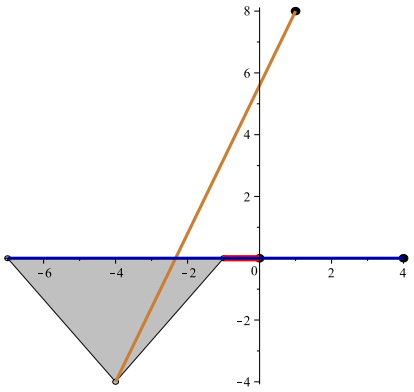


Fig. 3 This solution for the analyzed example cannot be obtained using the formulations presented in [2, 10, 12].

The obtained results confirm that the formulation of Gosselin et al. [2] and that of Kong and Gosselin [10] are not, in general, robust. Moreover, depending on the location of the chosen reference frames, the formulation of Husty [7] would also fail to provide all solutions. This is a highly relevant fact for the kinematic analysis and non-singular assembly-mode change studies of 3-RPR parallel manipulators [7, 11]. The presented coordinate-free formulation does not exhibit this kind of undesirable behavior.

5 Conclusions

Stating the forward kinematics analysis of 3-RPR parallel platforms directly in terms of poses introduces two major disadvantages: (a) a reference frame has to be introduced, and (b) all formulas involve translations and rotations simultaneously. This paper proposes a different approach in which, instead of directly computing the sought Cartesian poses, a problem fully posed in terms of distances is first solved. Then, the original problem can be trivially solved by sequences of bilaterations.

All proposed previous formulations have exceptions which translate into numerical instabilities when close to them. The approach proposed in this paper has no exceptions and, since all coefficients involved in the associated formulation are the result of operating with Cayley–Menger determinants with geometric meaning, it opens the door to a rational and exhaustive classification of all 3-RPR planar plat-

forms whose characteristic polynomials can be simplified either because they can be factorized or because their degrees are lower than six. This is certainly a point that deserves further attention.

Appendix

The polynomials Φ_a, Φ_b, Φ_c, and Φ_d in equation (9) can be expressed as:

$$\Phi_a = aT^2 + bT + c,\ \Phi_b = dT + e,\ \Phi_c = fT + g, \text{ and } \Phi_d = h,$$

where

$$\begin{aligned}
a &= -l_3^2L_1^2 + 3l_3^2L_2^2 - 3l_1^2L_2^2 - l_1^2L_3^2 + l_3^2L_3^2 + l_1^2L_1^2 - 3l_2^2L_3^2 - l_2^2L_2^2 + 3l_2^2L_1^2 \\
&\quad + \text{sign}(\alpha)\text{sign}(\beta)A_{123}A_{546}, \\
b &= -3l_2^2\rho_1^2L_2^2 + l_2^4L_3^2 - 5L_2^2l_2^2l_3^2 + l_2^2\rho_2^2L_1^2 - 3L_2^2l_3^2\rho_2^2 + 3L_2^2l_1^2\rho_2^2 \\
&\quad - 3l_2^2L_2^2\rho_2^2 - \rho_1^2L_2^2l_1^2 - l_2^4L_1^2 + 3l_2^4L_2^2 - 5l_2^2L_2^2L_1^2 - 3l_2^2L_2^2L_3^2 \\
&\quad + L_2^2l_1^2L_1^2 - L_2^2l_1^2L_3^2 + l_1^2l_2^2L_1^2 - l_1^2l_2^2L_3^2 - L_2^4l_3^2 + 3l_2^2\rho_1^2L_3^2 + l_1^2\rho_2^2L_3^2 \\
&\quad - 3L_2^2l_1^2l_2^2 - L_2^2l_3^2L_1^2 + L_2^2l_3^2L_3^2 - l_2^2l_3^2L_1^2 + l_2^2l_3^2L_3^2 + 8l_2^2L_2^2\rho_2^2 \\
&\quad + L_2^4l_1^2 - 3l_2^2\rho_1^2L_1^2 + l_3^2\rho_1^2L_1^2 - l_2^2\rho_2^2L_3^2 + \rho_1^2L_2^2l_3^2 - l_1^2\rho_1^2L_1^2 - l_1^2\rho_2^2L_1^2 \\
&\quad + l_1^2\rho_1^2L_3^2 - l_3^2\rho_1^2L_3^2 + 3l_2^2L_2^4 + l_3^2\rho_2^2L_1^2 - l_3^2\rho_2^2L_3^2 \\
&\quad + \left(-\rho_2^2 + l_2^2 + L_2^2 - \rho_1^2\right)\text{sign}(\alpha)\text{sign}(\beta)A_{123}A_{546} \\
c &= -(l_2 - \rho_2)(l_2 + \rho_2)(L_2 - \rho_1)(L_2 + \rho_1) \\
&\quad \left[-\left(l_2^2 + l_3^2 - l_1^2\right)\left(L_1^2 + L_2^2 - L_3^2\right) + \text{sign}(\alpha)\text{sign}(\beta)A_{1,2,3}A_{5,4,6}\right], \\
d &= 2\left(-l_3^2 + l_1^2 + 3l_2^2\right)\text{sign}(\alpha)A_{123} + 2\left(-L_2^2 - L_1^2 + L_3^2\right)\text{sign}(\beta)A_{546},
\end{aligned}$$

$$\begin{aligned}
e &= 2\left(l_2 - \rho_2\right)\left(l_2 + \rho_2\right) \\
&\quad \left[\left(-l_2^2l_3^2 + l_1^2\right)\text{sign}(\alpha)A_{123} + \left(-L_2^2 - L_1^2 + L_3^2\right)\text{sign}(\beta)A_{546}\right], \\
f &= 2\left(-l_3^2 + l_1^2 - l_2^2\right)\text{sign}(\alpha)A_{123} + 2\left(3L_2^2 - L_1^2 + L_3^2\right)\text{sign}(\beta)A_{546}, \\
g &= 2\left(L_2 - \rho_1\right)\left(L_2 + \rho_1\right) \\
&\quad \left[\left(-l_2^2l_3^2 + l_1^2\right)\text{sign}(\alpha)A_{123} + \left(-L_2^2 - L_1^2 + L_3^2\right)\text{sign}(\beta)A_{546}\right], \\
h &= 4\left(l_2^2 + l_3^2 - l_1^2\right)\left(L_1^2 + L_2^2 - L_3^2\right) - 4\text{sign}(\alpha)\text{sign}(\beta)A_{123}A_{546}.
\end{aligned}$$

Acknowledgements We are very much indebted to Philippe Wenger and his colleagues from IRCCyN who inspired this work. We also gratefully acknowledge the financial support of the

Autonomous Government of Catalonia through the VALTEC programme, cofinanced with FEDER funds, and the Colombian Ministry of Communications and Colfuturo through the ICT National Plan of Colombia.

References

1. Merlet, J-P.: *Parallel Robots*, Kluwer Academic Publishers, Dordrecht, (2000).
2. Gosselin, C., Sefrioui, J., Richard, M.J.: Solution polynomiale au problème de la cinématique directe des manipulateurs parallèles plans à 3 degrés de liberté. *Mechanism and Machine Theory*, **27**(2), 107–119 (1992).
3. Gosselin, C., Merlet, J-P.: On the direct kinematics of planar parallel manipulators: Special architectures and number of solutions. *Mechanism and Machine Theory*, **29**(8), 1083–1097 (1994).
4. Mertlet, J-P.: 5BARRES user's manual V. 0.1, available at http://www-sop.inria.fr/coprin/logiciels/RP/FK/3-RPR/notice-html.html
5. Lipkin, H., Duffy, J.: A vector analysis of robot manipulators. In: *Recent Advances in Robotics* Beni, G., Hackwood, S. J. (Eds.), Wiley, New York, pp. 175–242 (1985).
6. Kovács, P., Hommel, G.: On the tangent-half-angle substitution. In: *Computational Kinematics*, Angeles, J., Hommenl, G., Kovács, P. (Eds.), Kluwer Academic Publishers, Dordrecht, pp. 27–39 (1993).
7. Husty, M.: Non-singular assembly mode change in 3-RPR-parallel manipulators. In: *Computational Kinematics*, Kecskeméthy, A., Müller, A. (Eds.), Springer, Berlin/Heidelberg, pp. 51–60 (2009).
8. Wenger, P., Chablat, D., Zein, M.: Degeneracy study of the forward kinematics of planar 3-RPR parallel manipulators. *ASME Journal of Mechanical Design*, **192**(12), 1265–1268 (2007).
9. Thomas, F., Ros, L.: Revisiting trilateration for robot localization. *IEEE Trans. on Robotics*, **21**(1), pp. 93–101 (2005).
10. Kong, X., Gosselin, C.: Forward displacement analysis of third-class analytic 3-RPR planar parallel manipulators. *Mechanism and Machine Theory*, **39**(9), 1009–1018 (2001).
11. Zein, M., Wenger, P., Chablat, D.: Non-singular assembly-mode changing motions for 3-RPR parallel manipulators. *Mechanism and Machine Theory*, **43**, 480–490 (2008).
12. Hayes, M. J. D.: Kinematics of General Planar Stewart-Gough Platforms. PhD Thesis, Department of Mechanical Engineering and Centre for Intelligent Machines, McGill University (1999).

Hierarchical Decomposition and Kinematic Abstraction with Virtual Articulations

Marsette Vona

Northeastern University College of Computer and Information Science, Boston, MA 02115, USA; e-mail: vona@ccs.neu.edu

Abstract. We introduce a novel hierarchical model to partition a kinematic system into a set of nested subsystems. This is framed in a *mixed real/virtual* context, where some joints and links may exist in simulation only. We then use this capability to build a precise form of *kinematic abstraction*, where a potentially complex subsystem can be virtually replaced by a simpler "interface." Hierarchy and abstraction are interesting because they can help manage complexity in large (100+ DoF) mixed real/virtual mechanisms. We prove that checking if an abstraction is *proper* is PSPACE-hard, but show that even improper abstractions can be useful. Topological algorithms are presented for decomposing a hierarchical or abstracted kinematic system into subsystems that can be treated in isolation, thus speeding up kinematic computations. We demonstrate on a simulation of a hybrid serial/parallel modular tower with over 100 revolute joints.

Key words: Kinematic graphs, virtual joints, topological decomposition.

1 Introduction

Abstractions, where a complex implementation is hidden behind a simpler interface, are well known for managing complexity in computation. Is there any correlate in the domain of kinematics? We demonstrate the affirmative by introducing *structure abstraction* (Section 3). This novel and concrete technique can be used to effectively hide a complex kinematic subsystem behind a simpler "interface mechanism." Our approach is built on two foundations: *mixed real/virtual* models, where some links and joints are virtual; and *hierarchical linkages*.[1] By closing chains, virtual elements can be used to specify constraints (Fig. 3 gives an example). They also make structure abstraction more practical, as the interface mechanism can be virtual. We introduce hierarchical linkages, the second foundational technique, in Section 2.

Hierarchy and abstraction are interesting because they can help manage complexity in large systems, just as in other domains of computing and engineering. In particular, we are interested in high-DoF redundant mechanisms with 100+ DoF. Our original motivation arose from the study of kinematic simulation and control of

[1] We use "linkage" in this work to mean any kinematic system composed of links and joints.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_4,

linkage-type modular robots, where many modules can assemble in arbitrary topologies. One use for abstraction here is to help design virtual kinematic constraints – as chain-closing virtual joints – which guide a desired motion. Constraints can be defined for each subsystem and then placed below abstraction barriers. This is particularly useful given that high-DoF constructions often contain repeated substructure.

Abstractions can also lead to faster kinematic computation, e.g. numeric IK, by treating interface mechanisms as stand-in replacements for their more complex "implementations." The resulting motion can then be imposed on each implementation *in isolation*. We give the necessary algorithms in Section 4 to partition a hierarchical kinematic graph according to both (1) its biconnected components (a well-known decomposition) and (2) the imposed hierarchy. The latter ensures that interfaces are solved before implementations, as desired.

We implemented these algorithms as part of a general mixed real/virtual spatial kinematic simulation environment [1]. Section 5 demonstrates an example of interactive IK for a hybrid serial/parallel tower structure with over 100 revolute joints.

We find relatively little prior work in the area of kinematic abstraction. Davis [2] explored the idea of geometric abstraction, including a one-paragraph mention of "kinematic device as black box." Zanganeh and Angeles [3] studied partitioning of topologically large kinematic graphs, but did not separate interface from implementation. In [4], Williams and Mahew presented a tower structure similar to ours, but smaller in topological scale. More significantly, they developed an IK solving optimization related to structure abstraction, but only for one hand-decomposed instance. Our algorithms apply in the general case with no manual intervention.

2 Hierarchical Linkages

Figure 1 shows a kinematic graph L where the vertices are links (rigid bodies) and the edges are joints. Our implementation supports 12 different joint types. Cycles (closed chains) are allowed, but we always identify a spanning tree and distinct *closure* joints. The *parent* link of a joint is generally the one closer to the tree root. We allow the designer to identify this spanning tree by marking one closure joint in each cycle; it could also be automatically found. The pose of each link l is considered to be defined by the path of joint transforms from l to the tree root.

Such a flat graph L can be turned into a *hierarchical kinematic graph* $\mathscr{L}$ by imposing a properly nested set of disjoint edge cuts as shown in Fig. 2. We call each such demarcated subgraph a *sublinkage*. Further, we propose that a *disposition* of "driving," "driven," or "simultaneous" is assigned to each sublinkage. If P is the *parent* sublinkage immediately enclosing *child* sublinkage C, the disposition of C determines whether it is considered rigid with respect to P (driving), vice-versa (driven), or whether they are both mobile in the same context (simultaneous).

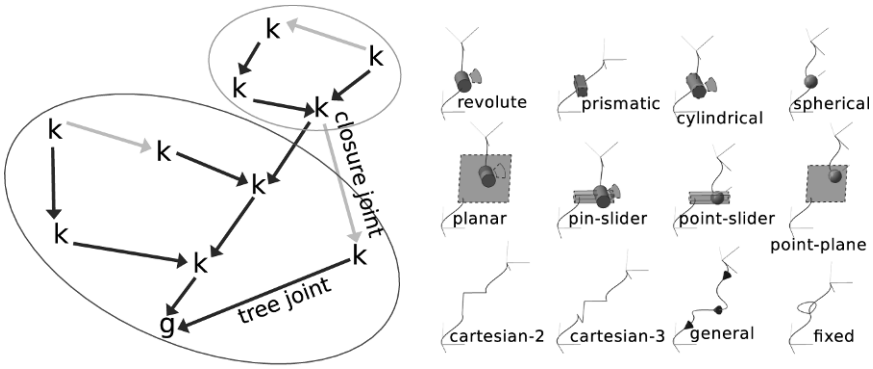


Fig. 1 The edges in a *kinematic graph* (left) correspond to joints and the vertices to links. We allow closed chains but always distinguish a spanning tree (dark edges). Twelve spatial joint types are available in our implementation (right), including all lower pairs except helical. Strongly connected components (circled, edge direction ignored) in the graph can be solved independently.

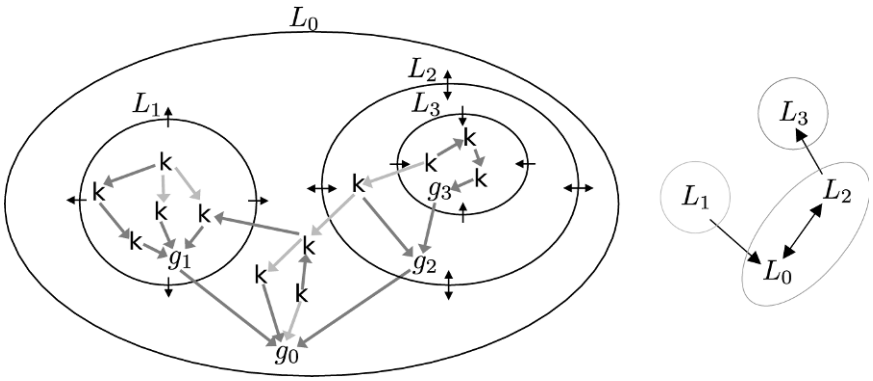


Fig. 2 To add hierarchy, partition a kinematic graph into a proper nesting of subgraphs. Each is specified as *driving* (e.g. L_1), *driven* (L_3), or *simultaneous* (L_2). The strongly connected components (circled) of a directed meta-graph of the subgraphs (right) are solvable independently.

For real linkages, such relationships may only be enforceable if fully actuated. For mixed real/virtual models, we can assume all virtual joints are actuatable; the real joints can be driven accordingly, again assuming no physical underactuation.

We do not actually remove the *crossing* joints in the edge cuts defining $\mathscr{L}$. Since the edge cuts are all disjoint, any crossing joint is cut by exactly one sublinkage boundary. Again assuming P to be the parent sublinkage of C, an *outcrossing* joint connects its child link in C to a parent link in P, and vice-versa for an *incrossing* joint. In the algorithms it is sometimes necessary find the innermost sublinkage containing a given joint or link. This is unambiguous except for crossing joints; we simply define them to be members of the inner of the crossed sublinkages.

Two additional topological restrictions are also imposed on crossing tree joints (there are no constraints on crossing closures): First, each sublinkage C always has

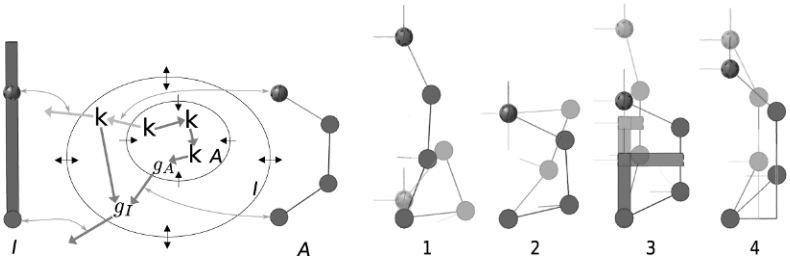


Fig. 3 In *structure abstraction* (left), an *implementation* linkage *A* is made a driven sublinkage of a new virtual *interface* linkage *I*. *I* replaces *A* in-context. The primary motion of interest in this example is the end-to-end stretching of *A* (1). Details, such as virtual prismatic joints (3) added to constrain an unwanted postural freedom (2), can be hidden underneath *I*. (4) shows a shorthand using a *Cartesian-2* joint that achieves the same effect as the virtual prismatic assembly in (3).

exactly one outcrossing tree joint (except at the top level) connecting the root of its spanning tree to a link in the parent sublinkage *P*. This ensures that each sublinkage has its own well-defined spanning tree. Second, incrossing tree joints are disallowed for a driven sublinkage *C*. Otherwise, changing the relative pose of links in *C* could change the relative pose of other links in the parent sublinkage *P*, violating the specified semantics that *P* should be rigid with respect to *C*.

3 Structure Abstraction

Structure abstraction is achieved by (a) encapsulating a connected part *A* of a linkage *L* s.t. *A* becomes demarcated as a simultaneous sublinkage of of *L*, and (b) substituting a virtual linkage *I* for *A* in *L*, with *I* simultaneous in *L* and *A* a driven sub-linkage of *I*. Figure 3 shows an example. The concept parallels traditional abstraction in computing: *I* can be simpler than *A*, but it should capture all of the behavior of *A* that would be relevant to the surrounding mechanism.

Making *A* a driven sub-linkage of *I* is not the only possible way to define abstraction in kinematics. For example, keeping *A* simultaneous could also make sense. However, the design choice to make *A* driven creates a fairly strong form of abstraction: motion of *L*, with *I* substituting for *A*, is independent of the motion of *A*. This can be helpful both (1) as a simplifying design aid (e.g. to allow virtual constraints as in Fig. 3 to to be abstracted away) and (2) to allow decomposition of the overall kinematic system into independently solvable subsystems, e.g. for IK computation.

This decoupling power comes with a trade-off: there is no built-in constraint to ensure that *A* can reach every configuration to which it may be driven by *I*. We say that *I* is a *proper abstraction* of *A* if all of the reachable configurations of *I*, when embedded in the surrounding linkage *L*, drive reachable configurations of *A*. It would be desirable to have an efficient algorithm that could determine, for any *L*, *I*, and *A*, whether *I* is a proper abstraction. This may be possible in some special

cases, but unfortunately, the general case is easily shown to be hard.

Theorem 1 *Determining whether an abstraction is proper is PSPACE-hard.*

Proof. By reduction from the *reachability problem* for 2D revolute linkages. Kempe's theorem [5] establishes that I could be constructed s.t. the links which drive A could move to arbitrary poses. But the problem of determining the reachability of arbitrary configurations of A from a starting configuration is known to be PSPACE-hard even for the restricted case of 2D revolute linkages [6, 7].

Thus, if a proper abstraction is required, it is up to the designer of that abstraction to ensure it. Sometimes this is easily done by construction. However, even improper abstractions can be useful when combined with task-priority IK (e.g. [8]) – if the kinematic constraints in A are given higher priority than those of, say, the crossing joints between I and A, then the motion of A will remain feasible. It will not exactly match the driving links in I, but because they are virtual, this can be acceptable.

4 Decomposition Algorithms

Practical algorithms for simulation and kinematic control of arbitrary topology linkages have polynomial runtime per iteration; e.g. least-squares IK with the SVD is typically quadratic in the number of DoF [1]. Thus it can be faster to run algorithms on disjoint subsystems, provided that their motion is actually independent. The overall motion is then the union of the motions of all sublinkages.

One well-known decomposition [9] finds biconnected components of the kinematic graph (Fig. 1), on the intuition that overlapping closed chains must be solved simultaneously. Our hierarchical model permits this, but also imposes additional lines of decomposition. Specifically, a driving sublinkage is solved before its parent, and vice-versa for a driven sublinkage. In fact, this ordering is one way to *enforce* the imposed driving/driven relationship. Because we also allow simultaneous sublinkages, this decomposition is not necessarily the same as the edge cuts defining the hierarchy. We must rather consider the strongly connected components of a directed meta-graph whose vertices correspond to the sublinkages and whose edges are directed according to sublinkage disposition. Figure 2 shows an example. (Note that a simultaneous sublinkage is connected to its parent with a bi-directional edge.)

In this section we present algorithms which decompose a given hierarchical linkage $\mathscr{L}$ into a set of disjoint components respecting both the biconnected components of the *flattening* of $\mathscr{L}$ (i.e. the underlying kinematic graph disregarding edge cuts) and also the strongly connected components of the associated meta-graph. The top-level call, DECOMPOSE (Alg. 1), has three phases. First, ASSIGNROUNDS (Alg. 2), labels the strongly connected components in $\mathscr{L}$ and also assigns a *solve round* to every SCC in $O(|\mathscr{L}|)$. (SCC and solve round labels are considered "inherited" by both the sublinkages within an SCC as well as the individual joints within them.) A sort by increasing round, performed at the end of DECOMPOSE in $O(|\mathscr{L}|\log|\mathscr{L}|)$, gives a solve ordering respecting all driving/driven relations.

The second phase of DECOMPOSE calls TRACESUPPORTS (Alg. 5) to identify the set of tree joints whose motion can affect the state of each closure joint. TRACESUPPORTS is $O(|T \cup C| + |T||C|)$ where T is the set of tree joints and C the set of closures in the flattening of $\mathscr{L}$. The helper function LOCKEDWRT(t,c) (Alg. 6) is key here, as it will ensure that tree joints will be considered locked with respect to closures unless both are in the same SCC.

The final phase calls FINDCOMPONENT (Alg. 7) to extract independent components by tracing transitive overlaps of closure supports. This phase is $O(|T||C|)$ because the $O(|T|)$ support of each closure is traversed once, and likewise for the $O(|C|)$ set of *supported-closures* for each of the $O(|T|)$ supporting tree joints.

Algorithm 1: DECOMPOSE($\mathscr{L}$)

Input: hierarchical linkage $\mathscr{L}$ with top-level sub-linkage L_0
Output: ordered partition $\mathscr{C}$ of closure joints
corresponding ordered set $\mathscr{T}$ of sets of unlocked supporting tree joints
unlocked support chains $S{\downarrow_c}, S{\uparrow_c}$ for each closure

ASSIGNROUNDS($\mathscr{L}$);
let T, C be the tree/closure partition of the joints in the flattening of $\mathscr{L}$;
foreach $j \in C$ **do** $S{\downarrow_c} \leftarrow \emptyset$, $S{\uparrow_c} \leftarrow \emptyset$, mark j unassigned;
foreach $j \in T$ **do** let *supported-closures*$_j \leftarrow \emptyset$, mark j unassigned;
let $U \leftarrow \emptyset$ ▷*the closures will be collected here*
;
$U \leftarrow$ TRACESUPPORTS(g_0, U) ▷g_0 *is the ground link in* L_0
;
let $\mathscr{C} \leftarrow \emptyset$, $\mathscr{T} \leftarrow \emptyset$;
while $U \neq \emptyset$ **do**
  let c be the first element of U, $C \leftarrow \emptyset$, $T \leftarrow \emptyset$;
  FINDCOMPONENT(c, U, C, T);
  add (C, T) to $(\mathscr{C}, \mathscr{T})$;
sort $(C, T) \in (\mathscr{C}, \mathscr{T})$ in order of increasing solve round of C;
return $(\mathscr{C}, \mathscr{T})$

Algorithm 2: ASSIGNROUNDS($\mathscr{L}$)

Input: hierarchical linkage $\mathscr{L}$ with top-level sub-linkage L_0
Output: each sub-linkage is marked with its solve round and its SCC in $\mathscr{L}$
foreach sub-linkage $L \in \mathscr{L}$ **do** $round_L \leftarrow -1$, $scc_L \leftarrow -1$;
let set of source sub-linkages $U \leftarrow \emptyset$, next SCC id $n \leftarrow 0$;
FINDSOURCESFROM(L_0, U);
foreach $L \in U$ **do** $n \leftarrow$ ASSIGNROUNDSFROM$(L, 0, n, n+1)$;

The full DECOMPOSE algorithm is thus $O(|T \cup C| + |T||C| + |\mathscr{L}| \log |\mathscr{L}|)$. A classical result in graph theory [10] is that biconnected components can be found in $O(|T \cup C|)$; DECOMPOSE is asymptotically slower, but also supports hierarchical

Algorithm 3: FINDSOURCESFROM(L,U)

Input: sub-linkage L, collected source sub-linkages U
Output: true if L drives its parent, updated U
let $dbd \leftarrow$ false ▷*driven by descendant*
;
foreach child sub-linkage M of L **do**
  if FINDSOURCESFROM(M,U) **then** $dbd \leftarrow$ true ;
if $\neg dbd$ and $((L = L_0)$ or (L is driving)) **then** add L to U;
return (L is driving) or (dbd and (L is simultaneous));

Algorithm 4: ASSIGNROUNDSFROM(L,r,i,n)

Input: sub-linkage L, solve round r, id i of SCC containing L, next unused SCC id n
Output: the new next unused SCC id
$round_L \leftarrow r$, $scc_L \leftarrow i$;
foreach child sub-linkage M of L **do**
  if ($round_M < 0$) and (M not driving) **then**
    if M driven **then** $n \leftarrow$ ASSIGNROUNDSFROM$(M, r+1, n, n+1)$;
    else $n \leftarrow$ ASSIGNROUNDSFROM(M,r,i,n) ▷*M is simultaneous*
    ;
    ;
if ($L \neq L_0$) and ($round_{parent_L} < 0$) and (L not driven) **then**
  if L driving **then** $n \leftarrow$ ASSIGNROUNDSFROM$(parent_L, r+1, n, n+1)$;
  else $n \leftarrow$ ASSIGNROUNDSFROM$(parent_L, r, i, n+1)$ ▷*L is simultaneous*
  ;
return n

Algorithm 5: TRACESUPPORTS(l,U)

Input: start link l, collected closures U
Output: U updated with newly found closures, and support chains for such
foreach j s.t. $parent_j = l$ **do**
  let $breadcrumb_l \leftarrow j$, $i \leftarrow child_j$;
  if TREE?(j) **then** $U \leftarrow$ TRACESUPPORTS(i,U);
  else ▷ j is a closure joint
    add j to U;
    repeat ▷ *trace down to least common ancestor*
      $p \leftarrow parent_i$, $i \leftarrow parent_p$;
      if $\neg$LOCKEDWRT?(p,j) **then** append p to $S{\downarrow_j}$, add j to $supported\text{-}closures_p$
    until $breadcrumb_i \neq \emptyset$;
    ▷*i is now the LCA of* $child_j$ *and* $parent_j$
    ;
    repeat ▷ *trace up from LCA*
      $c \leftarrow breadcrumb_i$, $i \leftarrow child_c$;
      if $\neg$LOCKEDWRT?(c,j) **then** append c to $S{\uparrow_j}$, add j to $supported\text{-}closures_c$
    until $c = j$;
$breadcrumb_l \leftarrow \emptyset$;
return U, support chains as side-effect

Algorithm 6: LOCKEDWRT?(t,c)

Input: tree joint t in sub-linkage L_t, closure joint c in sub-linkage L_c
Output: whether t is to be considered locked with respect to c
if t is explicitly locked **then return** true;
let $i \leftarrow scc_{L_c}$;
if CROSSING?(c) and (L_c driving) **then** $i \leftarrow scc_{parent_{L_c}}$;
return $(i \neq scc_{L_t})$;

Algorithm 7: FINDCOMPONENT(c,U,C_i,T_i)

Input: closure c to add to C_i, unexplored closures U
sets C_i and T_i of coupled closures and supporting unlocked tree joints
Output: updated U, C_i, and T_i
remove c from U, add c to C_i, mark c assigned;
foreach unlocked supporting tree joint t in $(S{\downarrow_c}, S{\uparrow_c})$ **do**
  if t unassigned **then**
    add t to T_i, mark t assigned;
    foreach unassigned closure u in $supported\text{-}closures_t$ **do**
    FINDCOMPONENT(u,U,C_i,T_i);

decomposition. Also, since many iterative numerical algorithms are at least quadratic in the number of DoF (which is at worst proportional to $|T|$), DECOMPOSE does not typically increase the overall computational complexity.

5 Scaling Results

We have implemented these algorithms as part of a new mixed real/virtual spatial kinematic simulator [1]. This environment supports general open- and closed-chain models using joints selected from the catalog in Fig. 1, and includes an interactive constraint solver based on task-priority iterative damped least-squares. Model topology can change on-line, and DECOMPOSE is automatically invoked as necessary.

In this section we demonstrate the scalability of our approach by showing interactive IK control of a simulated hybrid serial/parallel tower with 120 revolute joints and over 150 additional virtual joints (Fig. 4). Two layers of structure abstraction are applied, breaking up the operator's motion specification task and also enabling hierarchical decomposition to speed IK solving. We initially explored such a tower in [11], but that work was hand-coded and did not use the DECOMPOSE algorithm.

The tower is constructed of a chain of self-similar blocks. The actual joints comprising each block are redundant and can move in a variety of ways, but the operator intends only a subset of this motion. Extrinsically, the block should only have two DoF: it should be able to tilt left and right and to expand up and down. This forms the highest-level abstraction (**C** in the figure). Within this, a secondary constraint is that each 4-bar leg should effectively act like a piston, with the middle link re-

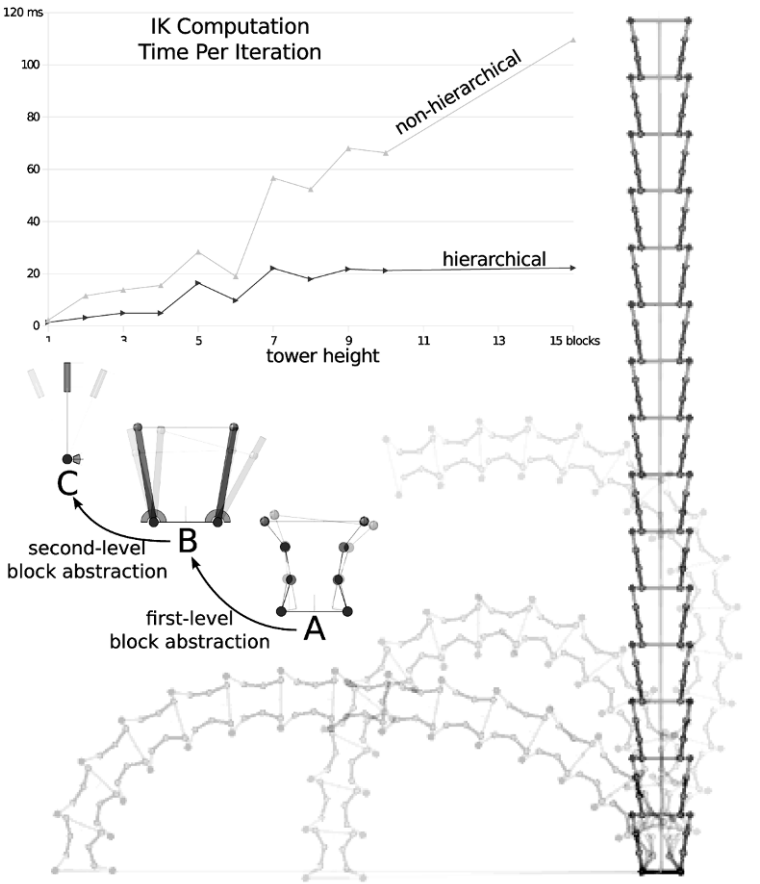


Fig. 4 Simulated interactive IK of a large (> 100 joint) hybrid serial/parallel tower is greatly accelerated by the introduction of two levels of abstraction (and hence hierarchy) for each block. The block legs **A** are virtually constrained and abstracted as in Fig. 3, producing an intermediate virtual model **B**. An additional level of abstraction covers **B** as a limited-travel RP chain **C**. An operator may drag any joint or link in the "backbone" chain of level-**C** interface mechanisms; the system interactively computes a corresponding motion for the tower respecting all constraints.

maining parallel to the axis of the piston. Virtual joints are added to enforce this local posture constraint, and then abstracted below a 3DoF RPR virtual interface (**B**). Fig. 3 shows the same construction. In this case, the abstractions can be kept proper by limiting the range of motion of the joints in the interfaces **B** and **C**.

These particular motion constraints, and this particular set of abstractions, are merely the designs of the operator. Other constraints and abstractions are possible: the idea is that the operator may express a desired set of motion constraints by designing constraints and structure abstractions.

Blocks can be strung together for towers of varying height. This final assembly is done at abstraction level **C**, so that the top-level linkage is simply a linear chain along the "backbone" of the tower (cf. [12]). The operator can then interactively specify a motion, e.g. by click-and-drag interaction, for any **C**-level link or joint.

To check the speedup afforded by hierarchical decomposition, we conducted an experiment with towers of varying heights (up to 15 blocks). Each was tested either as a flat linkage or structured with the above two levels of abstraction. The resulting measured computation times (on a typically loaded workstation) are comparatively plotted in Fig. 4. The hierarchical models remain at about 20 ms per iteration, which is acceptable for interactive response. But performance degrades to over 100 ms for a 15 block non-hierarchical tower, which results in very sluggish behavior.

These timings of course depend on the speed of the workstation. Furthermore, the hierarchical decomposition does not change the asymptotic cost of the IK computation, which is still quadratic in the number of joints solved in any single system (and the number of DoF in the level-**C** backbone still scales linearly with tower height). But lower constant factors for the hierarchical case mean that larger systems can be handled in practice before reaching the limits of interactivity.

6 Conclusions

In this paper we introduced a new hierarchical way to structure mixed real/virtual kinematic systems, and used it to define a novel form of abstraction for kinematics. As in other areas of computing and engineering, hierarchy and abstraction can help deal with large systems. We demonstrated our approach in a simulation of a tower structure with over 100 DoF. However, our algorithms and their implementation [1] are general across a broad class of spatial open- and closed-chain mechanisms.

Acknowledgements This work was performed at MIT, and was funded under the NSF EFRI program. Additional funding was provided under a Caltech/NASA/JPL DRDF grant.

References

1. M. A. Vona. Virtual articulation and kinematic abstraction in robotics. PhD thesis, EECS, Massachusetts Institute of Technology (2009).
2. E. Davis. Approximation and abstraction in solid object kinematics. Technical Report TR1995-706, New York University, Department of Computer Science (1995).
3. K. E. Zanganeh and J. Angeles. A formalism for the analysis and design of modular kinematic structures. *The International Journal of Robotics Research*, **17**(7), 720–730 (1998).
4. R. L. Williams and J. B. Mayhew. Control of truss-based manipulators using virtual serial models. In *The 1996 ASME Design Eng. Tech. Conf. and Comp. in Eng. Conf.* (1996).
5. A. B. Kempe. On a general method of describing plane curves of the *n*th degree by linkwork. *Proceedings of the London Mathematical Society*, **7**, 213–216 (1876).

6. R. Connelly and E. D. Demaine. Geometry and topology of polygonal linkages. In J. E. Goodman and J. OŔourke (Eds.), *CRC Handbook of Discrete and Computational Geometry*, 2nd ed. CRC Press, pp. 197–218 (2004).
7. J. Hopcroft, D. Joseph, and S. Whitesides. Movement problems for 2-dimensional linkages. *SIAM Journal of Computing*, **14**, 315–333 (1985).
8. P. Baerlocher and R. Boulic. An inverse kinematics architecture enforcing an arbitrary number of strict priority levels. *The Visual Computer*, **20**, 402–417 (2004).
9. C. Welman. Inverse kinematics and geometric constraints for articulated figure manipulation. Master's Thesis, Simon Fraser University (1993).
10. R. Tarjan. Depth-first search and linear graph algorithms. *SIAM J. Comp.*, **1**, 146–160 (1972).
11. C. Detweiler, M. Vona, K. Kotay, and D. Rus. Hierarchical control for self-assembling mobile trusses with passive and active links. In *IEEE International Conference on Robotics and Automation*, pp. 1483–1490 (2006).
12. G. S. Chirikjian and J. W. Burdick. The kinematics of hyper-redundant robot locomotion. *IEEE Transactions on Robotics and Automation*, **11**(6), 781–793 (1995).

Researching into Non-Singular Transitions in the Joint Space

M. Urízar, V. Petuya, O. Altuzarra and A. Hernández

Department of Mechanical Engineering, University of the Basque Country, Alameda de Urquijo s/n, 48013 Bilbao, Spain;
e-mail: {monica.urizar,victor.petuya,oscar.altuzarra,a.hernandez}@ehu.es

Abstract. It is known that *cuspidality* phenomenon appears in some parallel manipulators, called cuspidal manipulators, being able to perform non-singular transitions between different assembly modes. In this paper, the authors will present a methodology for obtaining the locus of cusp points in the joint space, which will be applied to generic 3-RPR planar parallel manipulators. This will permit analyzing non-singular transitions in a slice of the joint space and in the 3-dimensional joint space. It will be shown that as well as encircling a cusp point, analyzing the coalescence of solutions in the singular curves will be necessary so as to perform non-singular transitions.

Key words: Parallel manipulator, assembly mode, non-singular transition, cusp point.

1 Introduction

The workspace in parallel manipulators is usually complex and internally divided by the Direct Kinematic Problem (DKP) singularity locus. Additionally, the singularities produced by the Inverse Kinematic Problem (IKP) establish the boundary of the workspace. Hence many researches deal with obtaining both size and shape of the workspace, as well as the spatial distribution of the singularity loci [1, 2].

In general, parallel manipulators have multiple DKP and IKP solutions, called assembly modes and working modes respectively. In [3], the authors showed that it is feasible to perform transitions between different DKP solutions without crossing any singularity, hence guaranteeing the control of the robot all along the trajectory. Other authors have proved that it is possible to join different DKP solutions via paths totally free of singularities [4–8]. This fact suggests the possibility of a wider use of the workspace.

In [4], it was shown that when a DKP singularity is reached, several DKP solutions coalesce at the same pose. They demonstrated that some special singularities exist where three DKP solutions coalesce simultaneously. For three-degree-of-freedom (DoF) parallel manipulators, those special positions appear as cusp points in the DKP singular curves represented in the joint space for a constant value of one input. Performing non-singular transitions by surrounding cusp points has been ex-

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_5, © Springer Science+Business Media B.V. 2010

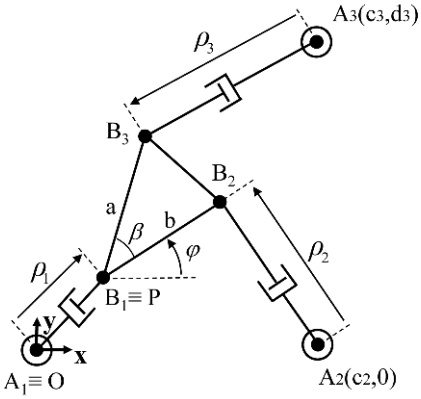


Fig. 1 3-*RPR* parallel manipulator.

tensively studied by several authors [7,9,10]. In [10], a new class of 3-RPR analytic manipulator was found that contrary to the four type of analytic manipulators studied in [11], this aforementioned manipulator is able to perform non-singular transitions. In [12], the authors obtained analytically the conditions for the existence of cusp points in the joint space for the RPR-2PRR planar parallel manipulator.

In this paper, the locus of cusp points for the general 3-RPR planar parallel manipulator will be obtained in a numeric way in the 3-dimensional joint space basing on the conditions developed in [12]. This way, it will be feasible to plan non-singular trajectories between assembly modes by varying the three input variables. Besides, it will be shown that not any path encircling a cusp point enables non-singular assembly mode change but that additional information about the singular curves which are crossed along the path is required.

2 Kinematic Problems

The 3-DoF parallel manipulator represented in Fig. 1 will be studied. It is a general 3-RPR planar parallel manipulator, having three limbs and a triangular moving platform. The three input variables correspond to prismatic joints that establish the limbs' lengths, named ρ_i for $i = 1,2,3$. The output variables are the coordinates (x,y) of the moving platform's point P, which is the joining-node with limb 1, and the orientation φ of the moving platform with respect to the fixed frame. Point A_1 is located at the origin of the fixed frame, the x-axis passing through point A_2. The geometric parameters a, b and angle β define the moving platform's characteristics. The three constraint equations are established as follows:

$$\rho_1^2 = x^2 + y^2 \tag{1}$$

$$\rho_2^2 = [x + b\cos(\varphi) - c_2]^2 + [y + b\sin(\varphi)]^2 \tag{2}$$

$$\rho_3^2 = [x + a\cos(\varphi + \beta) - c_3]^2 + [y + a\sin(\varphi + \beta) - d_3]^2 \tag{3}$$

A system of two linear equations in variables x and y can be obtained as proposed in [13], by subtracting Eq. (1) from Eqs. (2) and (3), hence obtaining:

$$Rx + Sy + Q = 0 \tag{4}$$

$$Ux + Vy + W = 0 \tag{5}$$

where

$$R = 2b\cos(\varphi) - 2c_2, \qquad S = 2b\sin(\varphi)$$
$$Q = \rho_1^2 + \rho_2^2 + c_2^2 + b^2 - 2bc_2\cos(\varphi)$$
$$U = 2a\cos(\varphi + \beta) - 2c_3, \qquad V = 2a\sin(\varphi + \beta) - 2d_3$$
$$W = \rho_1^2 + \rho_3^2 + c_3^2 + d_3^2 + a^2 - 2a[c_3\cos(\varphi + \beta) + d_3\sin(\varphi + \beta)]$$

Using Cramer's rule, x and y can be solved when the determinant $\Delta = RV - SU$ is different from zero. The conditions under which this determinant Δ vanishes are studied in detail in [10, 13]. A new class of an analytic manipulator was found in [10] by studying these aforementioned conditions, this type of analytic manipulator possessing the *cuspidality* phenomenon.

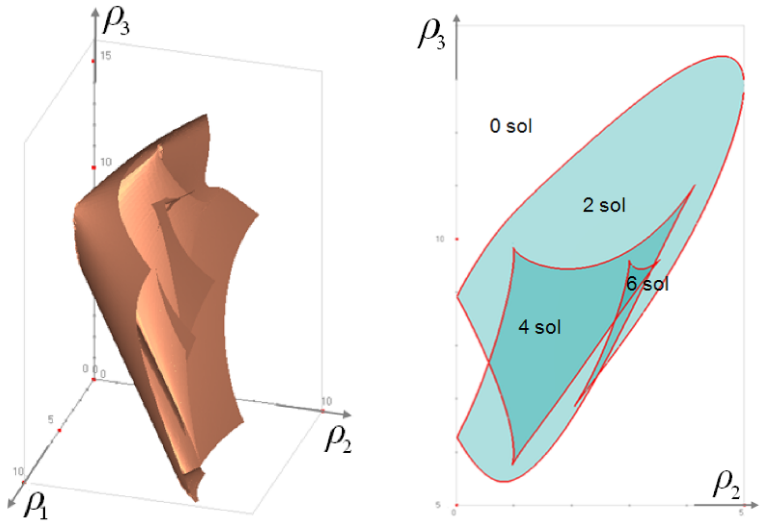
In this paper, the authors focus on the general case in which determinant Δ does not vanish. Then, substituting the corresponding values of x and y into Eq. (1), the characteristic polynomial is obtained, named $g(\rho_1, \rho_2, \rho_3, \varphi) = 0$. The expression of g is given by:

$$(SW - VQ)^2 + (QU - RW)^2 - \Delta^2\rho_1^2 = 0 \tag{6}$$

Performing the substitution $t = \tan(\varphi/2)$ into the characteristic polynomial in Eq. (6) yields a sixth-degree polynomial in t, meaning that general 3-RPR planar parallel manipulators have a maximum of six real DKP solutions.

2.1 Case Study

The 3-RPR planar parallel manipulator with the following design parameters will be used. The parameters of the fixed points A_i are: $c_2 = 2$, $c_3 = 8$, $d_3 = 5$, and the moving platform's geometry is defined by parameters: $a = 2$, $b = 1$ and angle $\beta = 60°$. The manipulator defined by these geometric parameters has at most six real DKP solutions. The DKP singularity surface represented in the joint space is depicted in Fig. 2(a), and a section of the joint space correspondent to $\rho_1 = 2$ is shown in Fig. 2(b). The different regions associated with 2, 4 or 6 DKP solutions can be differentiated in Fig. 2(b), the location of the six cusp points being clearly identified.



(a) (b)

Fig. 2 (a) DKP singularity locus in the joint space. (b) Joint space section for $\rho_1 = 2$.

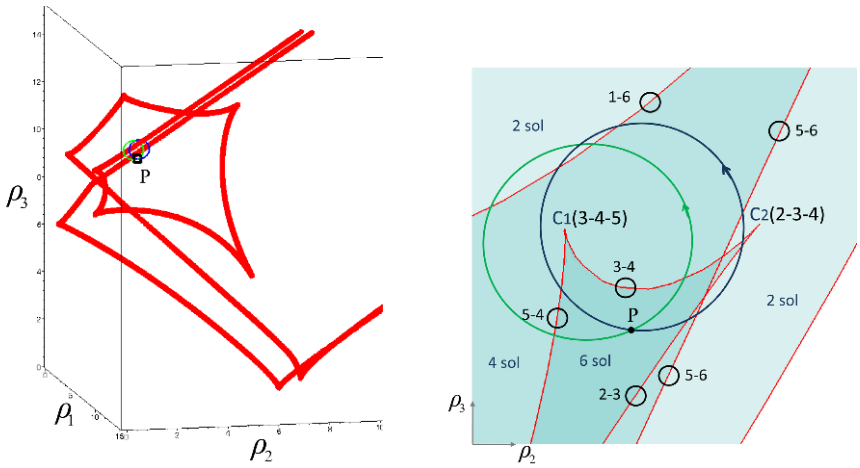
3 Locus of Cusp Points

The conditions for obtaining the cusp points in the joint space have been developed in [12]. These latter conditions are the following:

$$g = 0, \qquad h = \frac{\partial g}{\partial \varphi} = 0, \qquad \frac{\partial h}{\partial \varphi} = \frac{\partial^2 g}{\partial \varphi^2} = 0 \tag{7}$$

The first condition, $g = 0$, establishes that cusp points belong to the workspace, that is, they are DKP solutions of the manipulator and thus, they verify the expression of the characteristic polynomial. The second requirement, states that cusp points must belong to the DKP singularity surface, which implies verifying that at these points the expression $h = \partial g/\partial \varphi$ vanishes. Finally, the third condition establishes that a cusp point is a triple root of the characteristic polynomial.

Solving numerically the system formed by the three aforementioned conditions yields the locus of cusp points in the joint space entity, which is depicted in red color in Fig. 3(a). Intersecting the locus of cusps with a plane associated with one input constant, e.g. ρ_1, enables obtaining the number of cusp points and their location in a section of the joint space, i.e., the 2-dimensional slice (ρ_2, ρ_3). For very small values of input ρ_1 its associated plane does not intersect the locus of cusps at any point, while increasing input ρ_1 varies the number of intersection points, in the case of the manipulator under study, starting with four cusp points, increasing to six cusps and



(a) (b)

Fig. 3 (a) Locus of cusp points and loops encircling $C1$. (b) Loops in the planar joint space (ρ_2, ρ_3).

finally stabilizes to four cusp points. These results coincide with the ones obtained by Zein *et al.* in [14], where the different patterns that acquire the singular curves associated with different slices of the joint space were investigated for a manipulator with another values of the dimensional parameters which has a maximum of eight cusps.

4 Non-Singular Transitions in the Joint Space

The locus of cusp points in the joint space permits performing non-singular transitions between different DKP solutions by changing the three inputs. For example, a plane associated with input $\rho_1 = 2$ intersects the locus of cusps at six cusp points, the ones shown in Fig. 2(b). Starting at a point in the joint space belonging to the region of six DKP solutions, point P represented in Figs. 3(a) and 3(b) (coordinates $\rho_{1P} = 2$, $\rho_{2P} = 3$ and $\rho_{3P} = 9$), we will carry out two different loops belonging to the plane $\rho_1 = 2$ by surrounding one of the curves of the locus of cusps, which is equivalent to encircle the cusp at the intersection of the plane with the locus of cusps. Performing a trajectory around cusp $C1$, the green loop represented in Figs. 3(a) and 3(b), enables making up a non-singular transition from solution $S5$ to $S3$, as it is shown in Fig. 4(a). The evolution of the six solutions along the trajectory is depicted in Fig. 4(a), starting with 6 solutions, decreasing to 4 and 2, and then again increasing to 4 and 6. This means that the executed trajectory crosses the regions with six, four and two solutions of the DKP, as can be observed in Fig. 3(b).

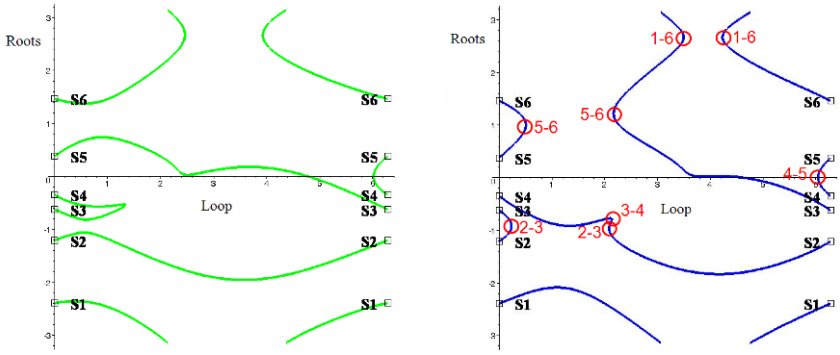
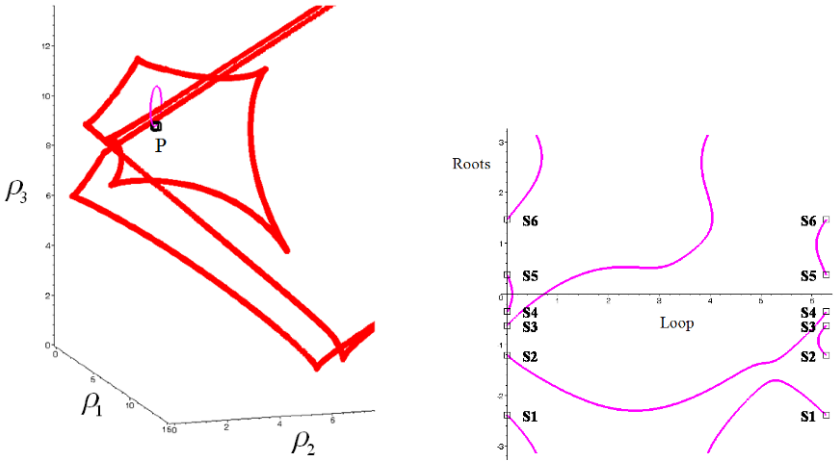


Fig. 4 Evolution of roots encircling $C1$ along the green loop (a) and along the blue loop (b).

Nevertheless, let us make the loop, also contained in plane $\rho_1 = 2$, depicted in blue shown in Figs. 3(a) and 3(b). Since cusp $C1$ is inside the loop it is supposed that a non-singular assembly mode change will occur. However, in the evolution of the roots shown in Fig. 4(b), it can be observed that none of the solutions performs a non-singular transition as all the paths that join two different solutions cross a singularity. In Fig. 4(b), the points where two DKP solutions coalesce, which means that the path crosses a singularity (the tangent to the curve is vertical), are marked with a red circle together with the numbering of both coalescing solutions.

This fact means that more information is required so as to plan a non-singular trajectory. This information is given by researching into the singular curves of the planar joint space depicted in Fig. 3(b). Each one of the singular curves produces the coalescence between two DKP solutions associated with a different sign of the DKP Jacobian. In a cusp point (see cusps $C1$ and $C2$ in Fig. 3(b)), two singular curves with a common solution join, thus coalescing three solutions. So as to perform a non-singular transition between two solutions i and j, the trajectory made must encircle a cusp point and all along the trajectory it can only cross singular curves which are joined together in that cusp or the ones that do not produce the coalescence of solutions i and j with another solution. In our example, starting at point P in Fig. 3(b), the green loop which is valid to make a non-singular transition between solutions $S5$ and $S3$, crosses the two singular curves that join in cusp $C1$ and the singular curve which makes the coalescence between solutions $6-1$. On the contrary, it is not possible to perform a non-singular transition from $S5$ to $S3$ by carrying out the blue loop as we are crossing the singular curves that produce the coalescence of solutions 2–3 and 5–6.

Our interest focuses on planning non-singular transitions in the 3-dimensional joint space, hence having the ability of varying the three input variables. It will be executed the 3D trajectory depicted in pink color in Fig. 5(a), which corresponds to the following input variations:



(a) (b)

Fig. 5 (a) Loop varying the three inputs. (b) Evolution of the roots along the 3D loop.

$$\rho_1 = \rho_{1P} + [r\sin(\alpha)\sin(\gamma) - r(1-\cos(\alpha))\cos(\beta)\cos(\gamma)] \quad (8)$$

$$\rho_2 = \rho_{2P} + [r\sin(\alpha)\cos(\gamma) + r(1-\cos(\alpha))\cos(\beta)\sin(\gamma)] \quad (9)$$

$$\rho_2 = \rho_{3P} + [r(1-\cos(\alpha))\sin(\beta)] \quad (10)$$

where ρ_{iP} for $i = 1,2,3$, correspond to the coordinates of point P, $r = 0.8$ is th e radius of the loop, the angles $\beta = \pi/2$ and $\gamma = 2\pi/3$ are the angles of the loop's plane with respect to the horizontal and vertical plane respectively, and angle α varies from 0 to 2π. The trajectory obtained with these aforementioned expressions of the three inputs surrounds two curves contained in the locus of cusp points, as shown in Fig. 5(a), and it can be observed in Fig. 5(b) that a non-singular transition has been made from $S2$ to $S4$ (the path joining $S3$ to $S1$ crosses a singularity surface as two points with vertical tangent can be observed). However, not all 3D trajectories surrounding one or more curves of the locus of cusp points will allow us performing a non-singular transition because, as it has been explained for the 2-dimensional slice of the joint space, the coalescence of DKP solutions in the singular curves (which will be singularity surfaces in the 3-dimensional joint space) must be analyzed in order to characterize the singularity surfaces.

5 Conclusions

In this paper the locus of cusp points has been numerically obtained for the 3-RPR parallel manipulator. This allows us performing non-singular transitions in both 2D

and 3D joint space by surrounding one or more curves of the locus of cusps. Nevertheless, it has been demonstrated for the 2D joint space, that not any path encircling a cusp point permits making a non-singular transition, but that additional information related to the singular curves is required. The next step will be characterizing the singularity surfaces in the 3D joint space in order to assess which trajectories enable assembly mode changing.

Acknowledgements The authors of this paper wish to acknowledge the financial support received from the Spanish Government via the Ministerio de Educación y Ciencia (Project DPI2008-00159) and the University of the Basque Country (Project GIC07/78).

References

1. J. Wang and C.M. Gosselin. Singularity loci of a special class of spherical 3-DOF parallel mechanisms with prismatic actuators. *Journal of Mechanical Design*, **126**, 319–326 (2004).
2. I.A. Bonev and C.M. Gosselin. Singularity loci of spherical parallel mechanisms. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, pp. 2968–2973 (2005).
3. C. Innocenti and V. Parenti-Castelli. Singularity free evolution from one configuration to another in serial and fully-parallel manipulators. *Journal of Mechanical Design*, **120**(1), 73–79 (1998).
4. P. McAree and R. Daniel. An explanation of never-special assembly changing motions for 3-3 parallel manipulators. *The International Journal of Robotics Research*, **18**(6), 556–575 (1999).
5. P. Wenger and D. Chablat. Workspace and assembly modes in fully-parallel manipulators: A descriptive study, in *Advances in Robot Kinematics and Computational Geometry*, pp. 117–126. Kluwer Academic Publishers (1998).
6. H. Bamberger, A. Wolf, and M. Shoham. Assembly mode changing in parallel mechanisms. *IEEE Transactions on Robotics*, **24**(4), 765–772 (2008).
7. M. Zein, P. Wenger, and D. Chablat. Non-singular assembly mode changing motions for 3-RPR parallel manipulators. *Mechanism and Machine Theory*, **43**(4), 480–490 (2008).
8. M.L. Husty. Non-singular assembly mode change in 3-RPR-parallel manipulator. In *Proceedings of the 5th International Workshop on Computational Kinematics*, Kecskeméthy, A., Müller, A. (Eds.), pp. 43–50, Springer (2009).
9. E. Macho, O. Altuzarra, C. Pinto, and A. Hernandez. Transitions between multiple solutions of the Direct Kinematic Problem. In *11th International Symposium Advances in Robot Kinematics*, France, June 22–26 (2008).
10. P. Wenger and D. Chablat. Kinematic analysis of a class of analytic planar 3-RPR parallel manipulator. In *Proceedings of the 5th International Workshop on Computational Kinematics*, Kecskeméthy, A., Müller, A. (Eds.), pp. 43–50, Springer (2009).
11. X. Kong and C.M. Gosselin. Forward displacement of third-class analytic 3-RPR planar parallel manipulators. *Mechanism and Machine Theory*, **36**, 1009–1018 (2001).
12. A. Hernandez, O. Altuzarra, V. Petuya, and E. Macho. Defining conditions for nonsingular transitions of assembly mode. *IEEE Transactions on Robotics*, **25**(6), 1438–1447 (2008).
13. P. Wenger, D. Chablat, and M. Zein. Degeneracy study of the forward kinematics of planar 3-RPR parallel manipulators. *Journal of Mechanical Design*, **129**, 1265–1268 (2007).
14. M. Zein, P. Wenger, and D. Chablat. Singular curves and cusp points in the joint space of 3-RPR parallel manipulators. *Robotica on Geometry in Robotics and Sensing* (special issue), **25**(6), 717–724 (2007).

MARIONET, A Family of Modular Wire-Driven Parallel Robots

J.-P. Merlet

INRIA Sophia Antipolis – Méditerranée, Sophia Antipolis, France;
e-mail: jean-pierre.merlet@sophia.inria.fr

Abstract. We present several prototypes of wire-driven parallel robots that we have recently designed and which use two different actuation schemes. Two of them have been completed and submitted to extensive tests. These tests have allowed to determine interesting open problems related to kinematics that are presented.

Key words: Kinematics, wire-driven parallel robots.

1 Introduction

Major drawbacks of parallel robots are the limited workspace and difficulty in adapting the robot to the tasks at hand. Using wires instead of rigid legs is a natural solution to the workspace problem. Indeed major limitations for the workspace of parallel robots are due to the limited stroke of linear actuators and to the limited range of motion allowed by the passive joints that are used. For a wire system the amount of possible leg length change may be very large while passive joints may be avoided or an appropriate mechanical design may push their influence much farther than classical mechanical joints.

After the pioneering work of Landsberger [11] wire-driven parallel robots have been extensively studied with various applications in mind. Without claiming to be extensive we may mention: crane [1, 8], motion tracking [2, 6] and metrology [18], haptic interface [3,5], surgery [7], rehabilitation [9,13,14] and sport training [12,17], telescope [15] and rescue robotics [4, 16].

However they are still various unsolved issues regarding wire-driven robots:

- *accuracy*: wire elasticity and sag, unmodeled wire deformation of the wires at the platform anchor points and measurement errors in the wire length induce inaccuracy in the platform location
- *kinematics and statics:* although it is sometimes claimed that parallel robot may be fully constrained, provided that they have at least $n+1$ wires for controlling n d.o.f. of the platform, this is true only if all wires are in tension. But being

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_6,

given the wire lengths a fully constrained robot may perfectly end up in a pose in which not all of its wires are under tension. Conversely although a solution of the inverse kinematics with positive tension may have been computed there is no guarantee that the platform will reach the desired pose when applying this solution to the robot. Hence direct kinematics remains to be investigated first by determining all possible poses satisfying the mechanical equilibrium condition that can be reached with 1 to n wires under tension, and then determining the current pose which is among all the solutions

- *modularity management*: wire-driven robot may be designed so that changing their geometry (i.e. the location of the anchor points of the wires) is an easy task. But determining the possible location of the anchor points so that the robot will fulfill a set of requirements is still an open problem.

To investigate these issues both theoretically and experimental we have decided to design and build a whole family of parallel wire-driven robots, based on two different actuation schemes.

2 Actuation Scheme

Classically wire-driven robots use drums that are actuated by a rotary motor. The wires coil on the drum and the measurements of the motors rotations allows theoretically to determine the wire lengths. Although implementing this actuation scheme is easy (and is used in some of our robots) it has various drawbacks, especially regarding the determination of the wire lengths. We may indeed calculate this length from the motor rotation under the following assumptions: the wires coil on the drum always in the same manner so that we can exactly determine the number of layers and the wire diameter is constant. In practice both assumptions have to be verified or a better measurement method has to be designed. This actuation scheme will be denoted Drum/Rotary Motor (DRM).

In order to improve the wire length measurement we have investigated another approach that is based on the use of a linear actuator and a pulleys system. One extremity of the wire is fixed to the ground and from this extremity the wire goes alternatively to a pulley on the mobile part of the actuator and then to a pulley on the fixed part of the actuator. The last pulley is fixed and the wire get out of the system through a hole whose location is also fixed with respect to the ground (Fig. 1).

The pulleys system is designed in such way that all wire strands in the system are parallel. The pulleys system allows to produce a length change of the wire that is a multiple of the stroke of the actuator: a system with m pulleys has a multiplication factor of m. Such actuation scheme will be called Linear Actuator/Pulleys System (LAPS). The LAPS principle has several advantages compared to DRM:

- the measurement of the motion of the linear actuators allows to measure the wire length with an error that is at most m time the error on the actuator measurement.
- the velocity of the wire is m times the velocity of the actuator

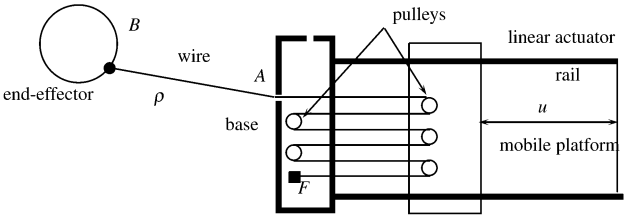


Fig. 1 Using a linear actuator to change wire length.

- the system may be designed in such way that the number of pulleys may be changed, hence allowing to adapt the multiplication factor. This offers an additional flexibility

LAPS has however some drawbacks: the maximal tension available in the wire is equal to the maximal force of the actuator divided by m and consequently there is a limit on the maximal number of pulleys as each additional pulley add a small amount of friction that decreases the available tension in the wire. Hence LAPS allows only limited length changes compared to the virtually unlimited range provided by DRM (however we will see that the motion range of LAPS is still large). Consequently LAPS should be used for fast and accurate robots with limited load and relatively limited workspace while DRM should be preferred for large load and workspace and for tasks for which accuracy is not a major issue.

3 The MARIONET family

The MARIONET family is a set of four wire-driven robots with different size, actuation scheme and applicative purposes:

- MARIONET-REHAB (MR): a 2.2 m $\times$ 1.2 m $\times$ 2 m robot using LAPS that is intended to be used for rehabilitation tasks and fast pick-and place operation
- MARIONET-CRANE (MC): a 15 m $\times$ 15 m $\times$ 15 m robot using DRM that is intended to be used for rescue operations and the manipulation of large load in a large workspace
- MARIONET-ASSIST (MA): a 3 m $\times$ 5 m $\times$ 3 m robot using DRM whose main task will be to act as a at-home lifting crane for assistance robotics, especially for elderly and handicapped people
- MARIONET-VR (MV): a 6 m $\times$ 5 m $\times$ 3 m robot using LAPS that will be used in a virtual reality environment as a motion provider and haptic device

Currently MARIONET-REHAB and MARIONET-CRANE have been built and are fully functional, while MARIONET-ASSIST and MARIONET-VR are at the design stage and will be available at the end of 2010.

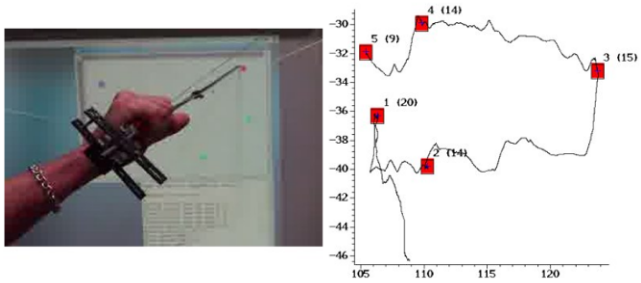


Fig. 2 Trajectory followed by a patient during a rehabilitation test: he should move in a straight line between the successive location of the square mark.

3.1 MARIONET-REHAB

The development of this robot with 7 wires (Fig. 3) started in 2004 and is the major test bed for the study of LAPS. It uses Copley Motion linear actuator with a stroke of 40 cm, a maximum force of 48 N, a positioning accuracy of 1 µm and a maximal velocity of 10 m/s. The LAPS may use up to 10 pulleys: with a maximal wire velocity of 100 m/s the platform velocity may theoretically exceed the speed of sound at some poses of its workspace, although we have tested it only up to 8 m/s. Our accuracy tests have shown that this robot was very accurate with a positioning error less than 0.1 mm.

We have recently tested MR for a rehabilitation task. After a coronary stroke patient may suffer a loss of arm coordination. A classical protocol to correct this coordination problem is to ask the patient to extend his arm and to draw with his finger the straight line between the successive location of a colored mark that is moving on a computer screen. The pratician then evaluates visually the arm coordination. We have first used MR in a passive mode where the robot just records the patient motion, thereby allowing an objective assessment of the quality of the motion (Fig. 2). It has then be observed that this protocol is very demanding for the patient as working with an extended arm leads quickly to arm fatigue. We have therefore used MR in a semi-active mode where the robot is still passive except in the vertical direction where it exerts a vertical force to relieve the patient from the weight of his arm (the apparent weight of the arm is reduced to approximately 15% of the real weight), allowing for longer rehabilitation session.

We intend now to use MR to measure precisely human joint motion. For that purpose we will use the 7 wires of the robot, together with several 3D accelerometers (Fig. 3).

3.2 MARIONET-CRANE

MARIONET-CRANE (MC) is a very large DRM 6 d.o.f. robot that is intended to be used for rescue operations. Our motivation to develop this robot may be illustrated

Fig. 3 MARIONET-REHAB used in rehabilitation tasks. The wires and the 3D accelerometers will be used to precisely measure human joint motion during a rehabilitation task.

Fig. 4 The outdoor implementation of MARIONET-CRANE.

by quoting Skynews after the 5/15/2008 earthquake in China: *The Chinese government has made an emergency appeal for cranes and heavy lifting equipment amid warnings that time is running out for survivors of Monday's 7.9-magnitude quake.* Hence we have designed the robot as a portable, fully autonomous device. It has 6 wire systems whose weight is about 10 kg with a drum capacity of between 50 and 100 m. Each wire may sustain up to a 1.5 tons load, leading to a robot that may lift up to 2 tons in most of its workspace. Optional tripods may be used to increase the height of the wire systems. MC has been deployed outdoor during 3 months at the end of 2009. The 6 tripods have been installed on top of three buildings, surmounting a 20 m $\times$ 20 m $\times$ 12 m work area (Fig. 4).

One role of MC is to free victims from the rumbles: this may require the use of all d.o.f. of the robot. To identify the necessary freeing motion we use a small mobile robot with a web cam so that these wires may be attached to the most appropriate anchor points (Fig. 5).

Then the victim may be lifted toward the surface (Fig. 4). An originality of this experiment is that the mobile robot is lifted with the victim so that the medical team on the surface may get physiological information during the transfer. Furthermore a small smart communicating object allows to get other measurements such as temperature and heart rate.

Fig. 5 MARIONET-CRANE used from freeing a victim from rumbles.

3.3 MARIONET-ASSIST *and* MARIONET-VR

MARIONET-ASSIST (MA) is a reduced version of MC that is intended to be used in a flat as a low-cost lifting crane and walking aid for elderly and handicapped people. At rest it lies on the ceiling of the room and is almost invisible. It is deployed on request and is designed to provide a lifting capacity of 150 kg. MARIONET-VR (MV) is a LAPS robot using linear drives with toothed belt whose stroke is 2 m and a pulleys system. It will be deployed in front of a 5 m immersive wall for rehabilitation, motion training and entertainment.

4 Lessons Learned

4.1 Improving the DRM

A major issue with large parallel robot is the determination of the ground anchor points of the wire system. We have successfully solved this problem by using a laser distance meter that measure the distances between a mark close to the anchor points and (1) the origin of the reference frame and (2) two points located on the x, y axis of the reference frame. A simple triangulation method allows then to determine quite accurately the location of the anchor points. We have then measured the accuracy of the robot when performing a vertical motion by using a plumb-line that allows one to determine the deviation of the robot center along the x, y axis. For a vertical motion of 8m the maximum deviation on these axis was less than 2cm, which is quite acceptable.

But the extensive tests of MC have shown that a major problem for DRM is the inaccuracy of the wire length measurement. Although MC has a guiding system for coiling the wires we have observed changes in the coiling process after several hours of use, leading to an error in the wire lengths that may exceed 50 cm. Hence on a regular basis it was necessary (especially for steel wire) to completely uncoil the wires and then coil them in a controlled manner. To overcome this drawback we will test a method that will allow to measure from time to time the wire lengths

almost exactly. We are currently being investigating two methods based on the same principle but using different sensors:

- *magnetic*: small strips of magnetic tape are glued at known distance on the wire. A Hall sensor in the wire system is able to detect the presence of one strip, allowing to determine the current wire length
- *optical*: colored marks that may be detected by an IR optical sensor

Between two marks we may interpolate the measurement using the drum rotation. Preliminary tests have shown that both type of marks are detectable. The magnetic method seems to offer a better resolution but is more difficult to implement and may disturb the coiling process.

4.2 Kinematics

Tests of both MC and MR have shown that kinematics is the most important issue for an efficient control of wire-driven parallel robot. Although all of our robots are so-called fully constrained one we have noticed discrepancies between our solution of both the direct and inverse kinematics and the observed robot pose. For the 6-wires MC the direct kinematics is equivalent to solving the one of a classical parallel robot and to retain the solution(s) that satisfies the mechanical equilibrium condition with positive tension in the wires. To the best of the author's knowledge there is no known bound on the maximum number of such solution. But even if such a solution exists the current robot pose may be different. Indeed let us consider the set of robots derived from the MC by suppressing from one up to five wires and compute their forward kinematic solutions. For a robot with m wires we have m constraint equations that relate the wire lengths to the pose parameters and 6 equilibrium equations for $6+m$ unknowns (6 pose parameters and m tensions in the wires). Among all these solutions we retain only the one such that the length(s) of the disconnected wire(s) is at least equal to the distance(s) between their anchor points. After this process we get a set of possible solutions, one of them being the current pose of the platform. But nothing guarantee that the current pose will be the solution of the fully-constrained robot. Hence even for a fully-constrained robot we have to investigate the forward kinematics of under-constrained robots. A similar study has to be performed for the inverse kinematics. These studies will be even more complex if elasticity and sagging of the wires [10] are taken into account.

4.3 Singularity

Singularity is related in depth to the static analysis of wire robot. Indeed the mechanical equilibrium constraint is equivalent to having the lines associated to the wires and the vertical line going through the platform center spanning a linear complex.

As linear complexes may be characterized geometrically such formulation allows one to write the equilibrium constraint without involving the wire tensions.

Using wires also allows the robot to move from one *aspect* to another one more easily than for parallel robots with rigid legs, because of the flexibility of the wires. We have been fortunate to observe and record such phenomena. Going through a singularity with a wire driven robot is also an interesting open problem.

5 Conclusion

Experimental tests allow to discover new theoretical open problems. Our large experimental effort with the development of 4 new wire-driven robots have shown that the kinematics of such robots is still an open issue. They have especially shown that for fully constrained robots the kinematics cannot be restricted to the case where all wires are under tension as the kinematics of all under-constrained systems have also to be solved.

References

1. Albus, J., Bostelman, R., and Dagalakis, N. The NIST SPIDER, a robot crane. *Journal of Research of the National Institute of Standards and Technology*, **97**(3), 373–385 (1992).
2. Andrade-Cetto, J. and Thomas, F. Wire-based tracking using mutual information. In *Advances in Robot Kinematics: Mechanisms and Motion*, J. Lenarčič and B. Roth (Eds.), pp. 3–14, Springer, Dordrecht (2006).
3. Arcara, P. et al. Perception of depth information by means of a wire-actuated haptic interface. In *IEEE Int. Conf. on Robotics and Automation*, pp. 3443–3348, San Francisco, 24–28 April (2000).
4. Ceccarelli, M., Ottaviano, E., and Tavolieri, C. Experimental activity on cable-based parallel manipulators: Issues and results at LARM in Cassino. In *2nd Int. Congress, Design and Modelling of Mechanical systems*, Monastir, 19–21 March (2007).
5. Dominjon, L., Perret, J., and Lécuyer, A. Novel devices and interaction techniques for human-scale haptics. *Visual Computer* **23**(4), 257–266 (2007).
6. Geng, Z. and Haynes, L.S. A 3-2-1 kinematic configuration of a Stewart platform and its application to six degree of freedom pose measurements. *Robotics and Computer-Integrated Manufacturing*, **11**(1), 23–34, (1994).
7. Hamid, S.A. and Simaan, N. Design and synthesis of wire-actuated universal-joint wrists for surgical applications. In *IEEE Int. Conf. on Robotics and Automation*, Kobe, 14–16 May, pp. 1807–1813 (2009).
8. Higuchi, T., Ming, A., and Jiang-Yu, J. Application of multi-dimensional wire crane in construction. In *5th Int. Symp. on Robotics in Construction*, Tokyo, pp. 661–668 (1988).
9. Homma, K. and Arai, T. Upper limb motion assist system with parallel mechanisms. In *2nd Japan-France Congress on Mechatronics*, Takamatsu, 1–3 November, pp. 388–391 (1994).
10. Kozak, K. et al. Static analysis of cable-driven manipulators with non-negligible cable mass. *IEEE Trans. on Robotics*, **22**(3), 425–433 (2006).
11. Landsberger, S.E. and Sheridan, T.B. A new design for parallel link manipulator. In *Proc. Systems, Man and Cybernetics Conf.*, Tucson, pp. 812–814 (1985).
12. Morizono, T., Kurahashi, K., and Kawamura, S. Realization of a virtual sports training system with parallel wire mechanism. In *IEEE Int. Conf. on Robotics and Automation*, Albuquerque, 21–28 April, pp. 3025–3030 (1997).

13. Ottaviano, E., Ceccarelli, M., and Palmucci, F. Experimental identification of kinematic parameters and joint mobility of human limbs. In *2nd Int. Congress, Design and Modelling of Mechanical Systems*, Monastir, 19–21 March (2007).
14. Rosati, G., Gallina, P., and Masiero, S. Design, implementation and clinical test of a wire-based robot for neurorehabilitation. *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, **15**(4), 560–569, (2007).
15. Su, Y.X. et al. Development of a large parallel-cable manipulator for the feed-supporting system of a next-generation large radio telescope. *J. of Robotic Systems*, **18**(11), 633–643 (2001).
16. Tadokoro, S. et al. A portable parallel manipulator for search and rescue at large-scale urban earthquakes and an identification algorithm for the installation in unstructured environments. In *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, Kyongju, 17–21 October, pp. 1222–1227 (1999).
17. von Zitzewitz, J. et al. A versatile wire robot concept as a haptic interface for sport simulation. In *IEEE Int. Conf. on Robotics and Automation*, Kobe, 14–16 May, pp. 313–318 (2009).
18. Williams II, R.L., Albus, J., and Bostelman, R. 3D cable-based cartesian metrology system. *J. of Robotic Systems*, **21**(5), 237–257 (2004).

Using Cosserat Point Theory for Estimating Kinematics and Soft-Tissue Deformation During Gait Analysis

A. Wolf[1], I. Sharf[2] and M.B. Rubin[3]

[1]*Technion, Israel, e-mail: alonw@tx.technion.ac.il*
[2]*McGill University, Canada, e-mail: inna.sharf@mcgill.ca*
[3]*Technion, Israel, e-mail: mbrubin@tx.technion.ac.il*

Abstract. Analyzing the biomechanics of human or animal motion requires a kinematical characterization of the motion of body segments as a function of time. Common noninvasive techniques for motion tracking involve placing markers on the skin of the body part and using a motion capturing system to track the marker positions. Determining the underlying joint motion requires processing the acquired marker coordinates. This is typically done using the assumption of rigid body motion of the segment even though soft tissues deform and marker acquisition is imperfect. In this paper, a new method is presented based on a continuum mechanics approach to estimate motion of body segments while assuming a non-rigid body transformation. The new method, based on the theory of a Cosserat Point, explicitly accounts for the fact that the marker kinematics include not only translation and rotation but also the associated and possibly large deformation of the body part. The method was validated with an experimental setup and the results were compared to ground truth results also obtained during the experiment.

Key words: Non-rigid body deformation, Cosserat point, motion estimation, soft-tissue deformation.

1 Introduction

Typical joint movement during normal gait involves a complex set of coupled translation and rotation. Consequently, in order to monitor and measure joint motion one needs to estimate the three-dimensional (3D) motion, composed of position and orientation of bone segments. Stereo-photogrammetry is one of the most popular, non-invasive methods used for gait analysis. During measurement, markers are attached to the skin surface of the body segment that is being tracked. A tracking system composed of a set of cameras detects the light which is reflected from the markers and reconstructs their 3D trajectories. These trajectories are used to calculate the pose of the underlying bone, with the erroneous assumption that markers and bone segments are rigidly connected, while it is well-known that markers on the surface (i.e., skin) of the body move with respect to the underlying bones, and the motion is task dependent [1–5]. It was also observed that attempts to remove soft-tissue artifacts through traditional filtering techniques can result in loss of in-

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_7, © Springer Science+Business Media B.V. 2010

formation [1, 5, 6]. Consequently, the soft-tissue artifact has been recognized as the major source of error in human motion analysis and is the primary factor limiting the resolution of detailed joint movement using skin-based systems.

The majority of studies describing three dimensional in vivo segment motions do not account for errors associated with non-rigid body movement. Several investigators have described methods that were designed to reduce the associated errors, however, these techniques in general model the limb segment as a rigid body, and then apply various estimation algorithms to obtain an optimal estimate of underlying skeletal motion, subject to a rigid body constraint [7,8]. For example, the motion of the skin markers can be minimized by the least squares methods [9–12], or the state function of the rigid body motion can be estimated using numerical differentiation [13] or Kalman filtering [14]. However, an exact solution has not yet been developed.

The objective of this work is to develop an alternative approach for characterizing the kinematics of a body segment by taking a non-rigid body deformation approach. Using the theory of a Cosserat point [15, 16], the presented method explicitly accounts for the fact that the marker kinematics include not only translation and rotation, but also the associated and possibly large deformation of the body part.

2 Motion and Deformation of a Tetrahedron

The main idea behind the proposed approach is to separate the collection of M markers into N groups of four makers that are used to define the four nodes of N tetrahedrons. Figure 1 shows a sketch of the I-th tetrahedron which is defined by the four vectors ${}_I\mathbf{x}_i(t)$ $(i = 0,1,2,3)$ which locate the positions of the nodes of the tetrahedron as a function of time t relative to a fixed position in space. For simplicity, the reference configuration of the tetrahedron will be characterized by the vectors ${}_I\mathbf{X}_i(t)$ $(i = 0,1,2,3)$ which are the values of ${}_I\mathbf{x}_i(t)$ at the initial time $t = 0$.

Treating the tetrahedron as a Cosserat point [15–17], it is possible to define the reference director vectors ${}_I\mathbf{D}_i$ and current configuration director vectors ${}_I\mathbf{d}_i(t)$ through the following equations:

$$\begin{aligned} {}_I\mathbf{D}_0 &= {}_I\mathbf{X}_0, \quad {}_I\mathbf{D}_i = {}_I\mathbf{X}_i - {}_I\mathbf{X}_0, \quad i = 1,2,3 \\ {}_I\mathbf{d}_0 &= {}_I\mathbf{x}_0, \quad {}_I\mathbf{d}_i = {}_I\mathbf{x}_i - {}_I\mathbf{x}_0, \quad i = 1,2,3 \end{aligned} \tag{1}$$

where the nodes have been specified so that ${}_I\mathbf{D}_i$ $(i = 0,1,2,3)$ and ${}_I\mathbf{d}_i$ $(i = 0,1,2,3)$ each form a right-handed triad of linearly independent vectors such that:

$${}_ID^{1/2} = {}_I\mathbf{D}_1 \times {}_I\mathbf{D}_2 \cdot {}_I\mathbf{D}_3 > 0, \quad {}_Id^{1/2} = {}_I\mathbf{d}_1 \times {}_I\mathbf{d}_2 \cdot {}_I\mathbf{d}_3 > 0 \tag{2}$$

In order to define rotation and deformation of the tetrahedron, it is convenient to introduce the reciprocal vectors ${}_I\mathbf{D}^i$ and ${}_I\mathbf{d}^i$ $(i = 1,2,3)$ through the expressions:

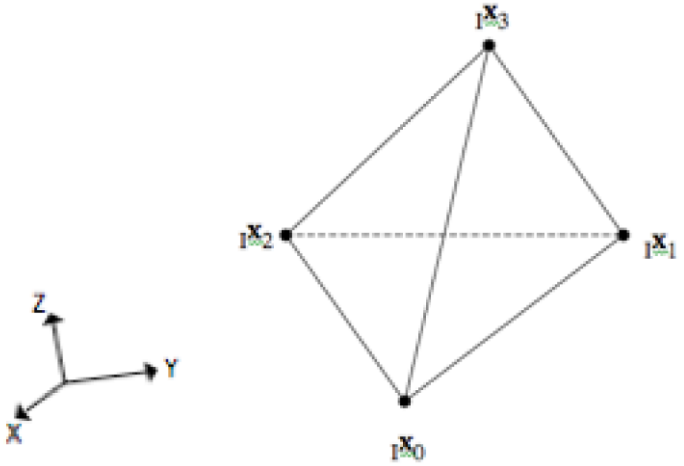


Fig. 1 Sketch of the present configuration of the I-th tetrahedron.

$$
\begin{aligned}
&{}_I\mathbf{D}^1 = {}_ID^{-1/2}({}_I\mathbf{D}_2 \times {}_I\mathbf{D}_3), \quad {}_I\mathbf{D}^2 = {}_ID^{-1/2}({}_I\mathbf{D}_3 \times {}_I\mathbf{D}_1), \quad {}_I\mathbf{D}^3 = {}_ID^{-1/2}({}_I\mathbf{D}_1 \times {}_I\mathbf{D}_2),\\
&{}_I\mathbf{d}^1 = {}_Id^{-1/2}({}_I\mathbf{d}_2 \times {}_I\mathbf{d}_3), \quad {}_I\mathbf{d}^2 = {}_Id^{-1/2}({}_I\mathbf{d}_3 \times {}_I\mathbf{d}_1), \quad {}_I\mathbf{d}^3 = {}_Id^{-1/2}({}_I\mathbf{d}_1 \times {}_I\mathbf{d}_2) \quad (3)
\end{aligned}
$$

which can be used to show that:

$$
{}_I\mathbf{D}^i \cdot {}_I\mathbf{D}_j = \delta^i_j, \qquad {}_I\mathbf{d}^i \cdot {}_I\mathbf{d}_j = \delta^i_j, \qquad i,j = 1,2,3 \tag{4}
$$

where δ^i_j is the Kronecker delta symbol. Moreover, it is convenient to introduce the deformation tensor ${}_I\mathbf{F}$ by:

$$
{}_I\mathbf{F} = \sum_{i=1}^{3} {}_I\mathbf{d}_i \otimes {}_I\mathbf{D}^i \tag{5}
$$

where $\mathbf{a} \otimes \mathbf{b}$ denotes the tensor product between two vectors $\mathbf{a}$ and $\mathbf{b}$. By definition ${}_I\mathbf{F}$ is a nonsingular deformation tensor which includes both rotation and stretch. In continuum mechanics, the polar decomposition theorem [18,19] is used to introduce the proper orthogonal rotational tensor ${}_I\mathbf{R}$ and the positive definite symmetric stretch tensor ${}_I\mathbf{M}$, such that:

$$
{}_I\mathbf{F} = {}_I\mathbf{R}\,{}_I\mathbf{M}, \quad {}_I\mathbf{R}^T\,{}_I\mathbf{R} = \mathbf{I}, \quad \det({}_I\mathbf{R}) = +1, \quad {}_I\mathbf{M}^T = {}_I\mathbf{M} \tag{6}
$$

where $\mathbf{I}$ is the second order unity tensor. Moreover, the associated symmetric deformation ${}_I\mathbf{C}$ and Lagrangian strain tensors ${}_I\mathbf{E}$ are defined by

$$
{}_I\mathbf{C} = {}_I\,\mathbf{F}^T\,{}_I\mathbf{F}, \qquad {}_I\mathbf{E} = \frac{1}{2}({}_I\mathbf{C} - \mathbf{I}) \tag{7}
$$

Specifically, given ${}_I\mathbf{F}$, it is easy to calculate ${}_I\mathbf{C}$. The stretch tensor ${}_I\mathbf{M}$ can be obtained as the matrix square root of ${}_I\mathbf{C}$ and then its inverse is used to determine the

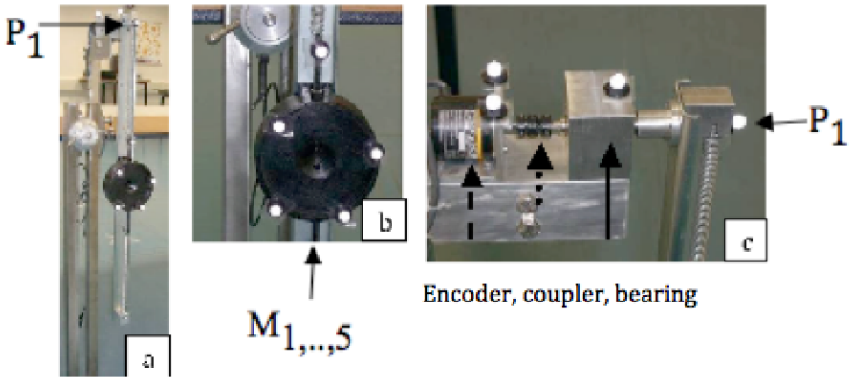


Fig. 2 Experimental setup.

rotation tensor ${}_I\mathbf{R} = {}_I\mathbf{F}_I\mathbf{M}^{-1}$. For the case of markers placed on an ideal *rigid* body, the rotation tensors, ${}_I\mathbf{R}$, are equal and represent the rotational motion of the body, while the strains ${}_I\mathbf{E}$ vanish. For a deformable body, the resulting rotation tensors represent an 'average' rotational motion of the deforming tetrahedra.

3 Experimental Setup

A series of experiments were conducted with a pendulum consisting of a heavy rigid mass oscillating on a spring, with the spring deforming perpendicular to the pendulum axis of rotation. During the experiments, the positions of five markers (Fig. 2b: $M_{1,..,5}$) were measured using a Vicon Motion Capture System (Mx13) [20]. As well, a number of stationary markers were affixed to the pendulum including one at the pendulum hinge (Fig. 2a,c: P_1). The reported capture system error is 0.1 pixels, which is equivalent to 0.1–0.2 mm (in 3D) [20]. Moreover, the joint rotation angle was measured by an encoder (Fig. 2c) whose readings were collected simultaneously with the marker positions.

Although this system (Fig. 2) does not completely simulate the mechanical properties of the human body, it can be used to examine the effectiveness and accuracy of the proposed method with respect to measurements of pure rigid body motion, i.e., the pendulum motion.

4 Results

For the analysis of the measurements and validation of the suggested method, the markers are divided into two groups:

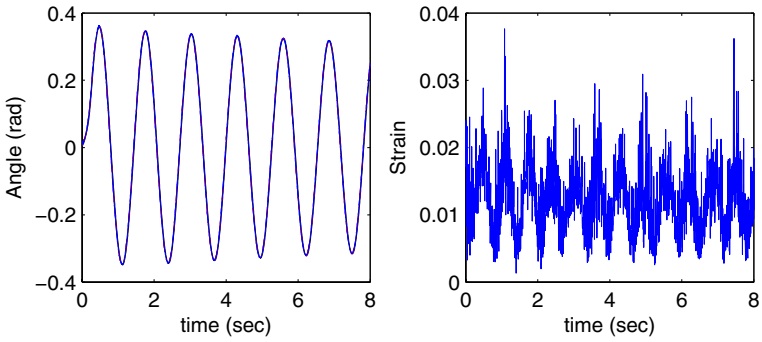


Fig. 3 Rigid tetrahedra results (a) rotation angles; (b) strain profile ($\|\mathbf{E}\|$) for one tetrahedron.

Group 1 Results in the first group are obtained with tetrahedra formed from the markers placed on the rigid mass only, i.e., four out of M_1–M_5. Therefore, these tetrahedra do not experience any deformation during motion (within the rigid-body assumption and the accuracy of the position measurement system). Since there were five markers available on the mass, five different tetrahedra could be formed. One of these however, is nearly planar and fails to produce reasonable results, leaving four rigid tetrahedra to be used.

Group 2 For the results in the second group, one of the tetrahedron vertices was the fixed marker at the pendulum hinge (P_1) and the other three vertices were formed from the markers on the rigid mass (three of M_1–M_5). Therefore, the resulting tetrahedra undergo deformation during the pendulum/mass motion. In this case, a total of 10 tetrahedra could be formed with the available markers.

The measured positions of the vertices of the tetrahedra defined in the two groups above were processed using the Cosserat point algorithm presented in Section 2. In particular, the rotation tensor $\mathbf{R}$ and the Lagrangian strain tensor $\mathbf{E}$ were calculated for each tetrahedron, at each sample time of the measured pendulum motion. For the tetrahedra in group 1 above, the rotation tensors were used to calculate the associated rotation angles, $\varphi(\mathbf{R})$ [21], to be compared to the rotation angle measurements from the encoder. For the deformable tetrahedra in group 2, the rotation difference measure is defined as $\varphi(\mathbf{R}^T\mathbf{R}_{encod})$ where $\mathbf{R}_{encod}$ is the rotation tensor defined by the encoded angle. This angular measure combines errors due to both the rotation angle and rotation axis predicted by the Cosserat point algorithm. As a measure of strain, use has been made of the tensor norm of the Lagrangian strain tensor $\mathbf{E}$ computed with the Cosserat point algorithm.

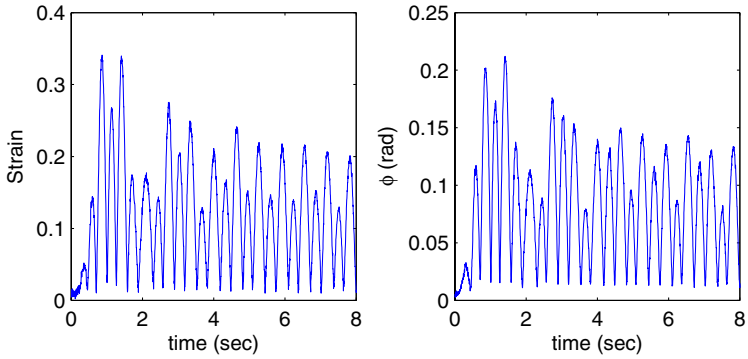


Fig. 4 Flexible tetrahedron results: (a) $\|\mathbf{E}\|$ and (b) φ.

4.1 Group 1 Results (Rigid Tetrahedra)

Figure 3 displays the time histories of the 4 angles for the first 8 seconds of motion (calculated with the four tetrahedra) and the angle measured by the encoder, which is considered to be accurate. As one can see, the rotation angles of the pendulum obtained with these tetrahedra are in excellent agreement with each other (correlation 1.0000) and with the encoder angle (correlation 0.9982). The magnitude of the strain from one of the rigid tetrahedra is plotted in Figure 3b. As can be seen, the magnitudes of the Lagrangian strain tensor $\mathbf{E}$ do not exceed 4% (similar results are obtained for other tetrahedra) and are on average 1.5%. The average rotation vectors were calculated using the Riemannian averaging procedure [21], over the four tetrahedra at every time instant, and they show the rotation axis to be primarily aligned with Y-direction (as expected). Finally, a measure of error of the peak angle was calculated. This was done for the average rotation angle obtained by averaging rotations of the four tetrahedra. The error measure gives the percentage relative error between the peaks of the encoded angle and those of the averaged estimates. The calculated error was 1.6% (maximum value).

4.2 Group 2 Results (Deformable Tetrahedra)

Results from one deformable tetrahedron are shown in Figure 4 which displays the magnitude of the strain and the rotation difference measure $\varphi(\mathbf{R}^T\mathbf{R}_{encod})$ defined earlier. The two profiles in Figure 4 are in excellent correlation, indicating that the magnitude of the strain can be used to identify those tetrahedra which produce accurate estimation of the underlying rigid-body motion of the segment.

Analysis of the experimental data produced four "good" tetrahedra with magnitudes of the strain below 15%. In all four, there is a very good correlation between the rotation difference measure φ and the magnitude of the strain (correlation coef-

ficients of 0.7996, 0.9848, 0.9743, 0.9867 respectively for the four tetrahedra). Most importantly, when the magnitude of the strain is small, the rotation difference predicted by the proposed algorithm is also small. In this regard, it is interesting to note that even though the deformation of the spring in the experiments is uniaxial the deformation states in the tetrahedral elements are not given by uniaxial strain.

5 Conclusions

The algorithm presented in this paper performs very well in terms of predicting the rotation in experimental data for a pendulum with an oscillating mass when the tetrahedra are formed from the markers on the rigid mass. In the mass-spring experiments where deformable tetrahedra are used, it has been shown that the error in the estimate of rigid-body rotation is small when the magnitude of the strain is small. Consequently, the magnitude of the strain can be used to identify when the body part is moving nearly as a rigid body and when the deformations are too large to treat the part as rigid. Future work will focus on the application of the algorithm for determining joint motions and soft-tissue deformations in the context of human biomechanics investigations.

Acknowledgements The assistance of M. Mor in generating the experimental results is gratefully acknowledged.

References

1. Leardini, A., Chiari, L., Croce, U.D., and Cappozzo A.: Human movement analysis using stereophotogrammetry Part 3. Soft tissue artifact assessment and compensation. *Gait and Posture*, **21**, 212–225 (2005).
2. Reinschmidt, C., Van Den Bogert, A.J., Lundberg, A., Nigg, B.M., and Murphy, N.: Effect of skin movement on the analysis of skeletal knee joint motion during running. *Journal of Biomechanics*, **30**(7), 729–732 (1997).
3. Manal, K., McClay, I., Galinat, B.,Stanhope, S.: The accuracy of estimating proximal tibial translation during natural cadence walking: Bone vs. skin mounted targets. *Clinical Biomechanics*, **18**, 123–131 (2003).
4. Ryu, J.H., Kouchi, M., Mochimaru, M., and Lee, K.H.: Analysis of skin movements with respect to bone motions using MR images. *International Journal of CAD/CAM*, **3**(2), 61–66 (2003).
5. Cappello A., Cappozzo A., Lucchetti L., La Palombara P.F., Leardini, A.: Multiple anatomical landmark calibration for optimal bone pose estimation. *Human Movement Science*, **16**, 259–274 (1997).
6. Ball, K.A., Pierrynowski, M.R.: Modeling of the pliant surfaces of the thigh and leg during gait. *SPIE Proceedings Series* **3254**, 435–446 (1998).
7. Alexander, E.J., Andriacchi, T.P.: Correcting for deformation in skin-based marker systems. *Journal of Biomechanics*, **34**(3), 355–361 (2001).

8. Wolf, A., Mor, M.: Rigid-body motion estimation using statistical solid dynamics method. In *Proceedings of the ASME Engineering Systems Design and Analysis Conference (ESDA 2008)*, Haifa, Israel (2008).
9. Camomilla, V., Donati, M., Stagni, R., Cappozzo, A.: Non-invasive assessment of superficial soft tissue local displacements during movement: A feasibility study. *Journal of Biomechanics*, **42**(7), 931–937 (2009).
10. Cerveri, P., Rabuffetti, M., Pedotti, V., Ferrigno, G.: Real-time human motion estimation using biomechanical models and non-linear state-space filters. *Medical and Biological Engineering and Computing*, **41**, 109–123 (2003).
11. Choset, H., Lynch, K.M, Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L.E., Thrun., S.: Kalman filtering. In *Principles of Robot Motion: Theory, Algorithms, and Implementation*, Massachusetts Institute of Technology, pp. 269–301 (2005).
12. Ryu, T., Choi, H.S., Chung, M.K.: Soft tissue artifact compensation using displacement dependency between anatomical landmarks and skin markers: A preliminary study. *International Journal of Industrial Ergonomics*, **39**(1), 152–158 (2009).
13. Fioretti, S., Jetto, L.: Accurate derivative estimation from noisy data: A state-space approach. *International Journal of Systems Science*, **20**(1), 33–53 (1989).
14. David A.W.: *Biomechanics and Motor Control of Human Movement*, John Wiley and Sons, Waterloo, Ontario, Canada, pp. 33–45 (1930).
15. Rubin, M.B.: On the theory of a Cosserat point and its application to the numerical solution of continuum problems. *J. Appl. Mech.*, **52**, 368–372 (1985).
16. Rubin, M.B.: On the numerical solution of one-dimensional continuum problems using the theory of a Cosserat point. *J. Appl. Mech.*, **52**, 373–378 (1985).
17. Rubin, M.B.: *Cosserat Theories: Shells, Rods and Points*, Solid Mechanics and Its Applications, Vol. 79. Kluwer Academic Publishers, Dordrecht (2000).
18. Malvern, L.E.: *Introduction to the Mechanics of a Continuous Media*. Prentice-Hall (1969).
19. Rubin, M.B.: A simplified implicit Newmark integration scheme for finite rotations. *Computers and Mathematics with Applications*, **53**, 219–231 (2007).
20. Vicon Motion Systems and Peak Performance Inc., http://www.vicon.com (2007).
21. Sharf, I., Wolf, A., Rubin, M.B.: Arithmetic and geometric solutions for average rigid-body rotation. *Mechanism and Machine Theory*, to appear (2010).

PART 2

Mechanical Generators of 2-DoF Translation along a Ruled Surface

Chung-Ching Lee[1] and Jacques M. Hervé[2]

[1]*National Kaohsiung University of Applied Sciences, Taiwan R.O.C.; e-mail: cclee@cc.kuas.edu.tw*
[2]*Ecole Centrale Paris, France; e-mail: jacques.herve07@orange.fr*

Abstract. The theorem that any subset of translations is normal or invariant by conjugation in the group of a spatial 3-DoF translation is applied to the synthesis of a novel type of parallel manipulators, which generate 2-DoF translation along a ruled helicoid. A parallel manipulator generating 2-DoF translation along a revolute hyperboloid of one sheet is a special case. The revolute one-sheet hyperboloid is a doubly ruled surface. The double geometric generation of that hyperboloid by the rotation of a straight line is further exploited and leads to more architectural types of parallel manipulators, which produce 2-DoF translation of the end-effector.

Key words: Group conjugation, 2-DoF translation, ruled helicoid, revolute hyperboloid of one sheet, parallel mechanism, doubly ruled surface.

1 Introduction

Parallel manipulators, which generate 3-degree-of-freedom (3-DoF) translation of the end-effector have attracted researchers for a long time [1–9]. However, the synthesis of parallel generators of 2-DoF translation along a surface that is neither a plane nor a sphere is a novel topic. In [10], a theorem derived from the group-algebraic properties of the 6-dimensional (6D) set of displacements is stated: any subset of translations is normal or invariant by conjugation in the commutative (or Abelian) Lie 3D group $\mathscr{T}$ of translations. By virtue of that theorem, any 2-DoF translation along a surface that may be a ruled helicoid or a hyperboloid of one sheet is normal in the group $\mathscr{T}$. Using the theorem, a novel parallel generator of 2-DoF translation along a ruled helicoid is synthesized. From a special case, parallel generators of 2-DoF translation along a revolute hyperboloid of one sheet are also obtained.

In the first step of the synthesis, we introduce a single-closed-loop chain, which includes a rigid body and moves by translation along the surface of a ruled helicoid. Hereinafter, P and H designate respectively a prismatic pair and a helical pair (screw joint) and the underline indicates the parallelism of $\underline{\text{H}}$ axes. The closed-loop chain can be considered to be the parallel arrangement of a PPP chain generating the group

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_8, © Springer Science+Business Media B.V. 2010

$\mathscr{T}$ and a HPH open chain with parallel screw axes. The HPH chain generates a 3D sub-manifold of a 4D group $\mathscr{X}$ of Schoenflies (or Schönflies) motions. A vector calculation is done for verification. In the second step, we add a second HPH-$\|$-PPP chain, which is conjugate or congruent by translation, to the first HPH-$\|$-PPP chain. In the third step, we rigidly connect the two end-effectors and remove the superfluous PPP sub-chains. Finally, we obtain a HPH-$\|$-HPH parallel manipulator whose end-effector translates along a surface, which is generally a ruled helicoid. In a special case, the surface is a doubly ruled hyperboloid and more parallel manipulators are derived.

2 2-DoF Translation along a Ruled Helicoid

The parallel layout of a PPP generator of the group $\mathscr{T}$ with a HPH open chain makes up a HPH-$\|$-PPP mechanism, which is depicted in Figure 1. (**i**, **j**, **k**) is an orthonormal vector base. The points A and B are located on the H axes, which are both parallel to **k**. $\mathscr{H}(A,\mathbf{k},p_1)$ denotes the 1D group of helical motions with the axis $(A,\mathbf{k})$ and the pitch p_1; $\mathscr{T}(\mathbf{s})$ is the 1D group of rectilinear translations parallel to **s**. The parallel kinematic chain embodies $\mathscr{H}(A,\mathbf{k},p_1)\mathscr{T}(\mathbf{s})\mathscr{H}(B,\mathbf{k},p_2)\cap \mathscr{T}(\mathbf{j})\mathscr{T}(\mathbf{i})\mathscr{T}(\mathbf{k})$. The chain is movable with two DoFs. As a matter of fact, it is a trivial chain associated with the 4D Schoenflies group $\mathscr{X}(\mathbf{k})$ [11] and it is made up of six 1-DoF pairs; $6-4=2$. The displacement set $\mathscr{H}(A,\mathbf{k},p_1)\mathscr{T}(\mathbf{s})\mathscr{H}(B,\mathbf{k},p_2)\cap \mathscr{T}(\mathbf{j})\mathscr{T}(\mathbf{i})\mathscr{T}(\mathbf{k})$ is equal to $\mathscr{H}(A,\mathbf{k},p_1)\mathscr{T}(\mathbf{s})\mathscr{H}(B,\mathbf{k},p_2)\cap\mathscr{T}$ and is a 2D submanifold of $\mathscr{T}$. In a 2-DoF translation, any point of the translating body moves on a surface that is a 2D submanifold of the 3D Euclidean affine space. Two points belonging to the same translating body move on two surfaces that are congruent through the translation that transforms one of the two points into the other one.

In a HPH-$\|$-PPP manipulator, let M be any point belonging to the end-effector. The parallel H axes have the reduced pitches $k_1=p_1/2\pi$ and $k_2=p_2/2\pi$. $p_1=p_2=0$ is a possible special case and then the Hs are revolute R pairs. The PPP limb is a generator of $\mathscr{T}$. The 3D group $\mathscr{T}$ is the set of point transformations

$$M\rightarrow M'=M+a_1\mathbf{i}+b_1\mathbf{j}+c_1\mathbf{k}, \tag{1}$$

which is also expressed by $(\mathbf{MM}')=a_1\mathbf{i}+b_1\mathbf{j}+c_1\mathbf{k}$ where $(\mathbf{MM}')=M'-M$ is the vector obtained from the points M and M'. The parameters a_1, b_1 and c_1 are the scalar values of the rectilinear translations produced by the P pairs where the variables are unknown. In Figure 1, (**AN**) is perpendicular to (**NB**), which is parallel to the unit vector **s**; $(\mathbf{AN})=r\mathbf{i}$, $(\mathbf{NB})=b\mathbf{s}$, $\mathbf{s}=\sin\alpha\mathbf{j}+\cos\alpha\mathbf{k}$ where the angle α is a given constant. In order to facilitate the understanding, the point B does not coincide with N and b is not zero in the figure. Because displacements (or rigid-body motions) represent changes of rigid-body positions, the geometric description of an initial home configuration of a mechanism is a datum that is necessary for the characterization of all the relative motions. With no loss of generality and for

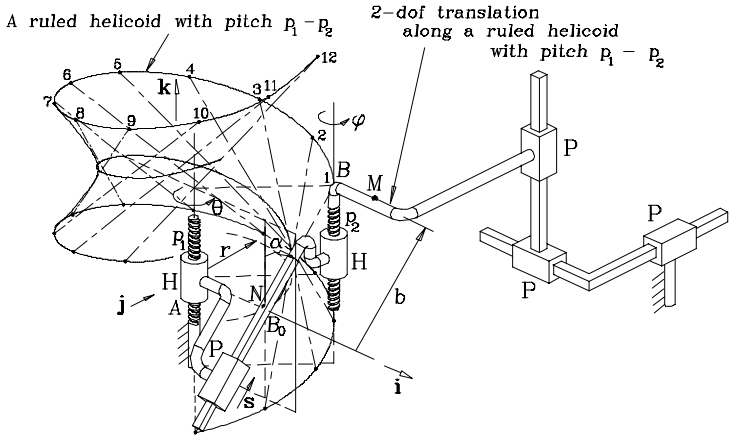


Fig. 1 A HPH-||-PPP generator of 2-DoF translation along a ruled helicoid: general configuration.

the sake of simple calculation, at the home configuration shown in Figure 2, the position B_0 of the end-effector point B is assumed to coincide with N. Hence, $B_0 = N(\neq B)$ in the following calculation. Moreover, the position of M is M_0 at the home configuration. The displacements produced by the HPH limb can be attained by a succession of three 1-DoF motions. In the first stage, the second H moves with an angle ϕ around the axis $(B_0, \mathbf{k})$ while the first H and the P do not move ($\theta = 0$ and $b = 0$); M_0 becomes M_i, which is expressed as

$$M_0 \rightarrow M_i = B_0 + \phi k_2 \mathbf{k} + \exp(\phi \mathbf{k}\times)(\mathbf{B_0M_0}) \text{ or } (\mathbf{B_0M}_i) = \phi k_2 \mathbf{k} + \exp(\phi \mathbf{k}\times)(\mathbf{B_0M_0}) \tag{2}$$

In the second stage, the second H is locked (angle ϕ keeps its value), the first H keeps its home position ($\theta = 0$) and the P provides a translation parallel to $\mathbf{s}$, with an amplitude b. Then, the point M_i becomes M_i'

$$M_i \rightarrow M_i' = M_i + b\mathbf{s} \tag{3}$$

With $(\mathbf{B_0M}_i') = b\mathbf{s} + \phi k_2 \mathbf{k} + \exp(\phi \mathbf{k}\times)(\mathbf{B_0M_0})$ and $(\mathbf{NM}_i') = b\mathbf{s} + \phi k_2 \mathbf{k} + \exp(\phi \mathbf{k}\times)(\mathbf{B_0M_0})$ because of the choice $B_0 = N$, we have

$$(\mathbf{NM}_i') = b\sin\alpha \mathbf{j} + (b\cos\alpha + \phi k_2)\mathbf{k} + \exp(\phi \mathbf{k}\times)(\mathbf{NM_0}), \tag{4}$$

In the third stage, the second H and the P are locked and, therefore, ϕ and b keep their values. The first H moves with angle θ around the fixed axis $(A, \mathbf{k})$ and M_i' is transformed into M'

$$M_i' \rightarrow M' = A + \theta k_1 \mathbf{k} + exp(\theta \mathbf{k}\times)(\mathbf{AM}_i') or (\mathbf{AM}') = \theta k_1 \mathbf{k} + exp(\theta \mathbf{k}\times)(\mathbf{AM}_i') \tag{5}$$

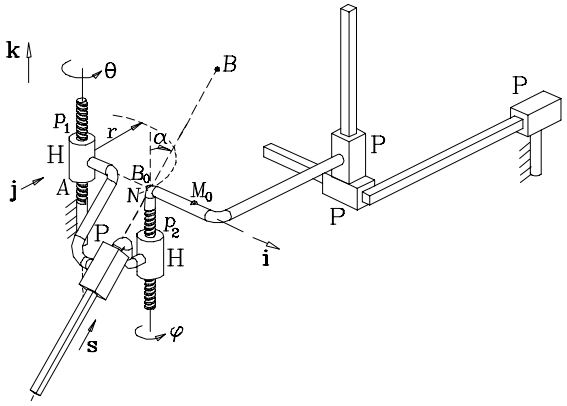


Fig. 2 A HPH-||-PPP generator of 2-DoF translation along a ruled helicoid:initial posture of parametrization

Using $(\mathbf{AM}_i') = (\mathbf{AN}) + (\mathbf{NM}_i') = r\mathbf{i} + b\sin\alpha\mathbf{j} + (b\cos\alpha + \phi k_2)\mathbf{k} + \exp(\phi\mathbf{k}\times)(\mathbf{B}_0\mathbf{M}_0)$ yields

$$(\mathbf{AM}') = \theta k_1\mathbf{k} + \exp(\theta\mathbf{k}\times)[r\mathbf{i} + b\sin\alpha\mathbf{j} + (b\cos\alpha + \phi k_2)\mathbf{k} + \exp(\phi\mathbf{k}\times)(\mathbf{B}_0\mathbf{M}_0)]$$

$$= (\theta k_1 + \phi k_2 + b\cos\alpha)\mathbf{k} + \exp(\theta\mathbf{k}\times)(r\mathbf{i} + b\sin\alpha\mathbf{j}) + \exp[(\theta+\phi)\mathbf{k}\times](\mathbf{B}_0\mathbf{M}_0)$$

Notice that $\exp(\theta\mathbf{k}\times)(r\mathbf{i}) = r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j})$ and $\exp(\theta\mathbf{k}\times)(b\sin\alpha\mathbf{j}) = b\sin\alpha(-\sin\theta\mathbf{i} + \cos\theta\mathbf{j})$,

$$\begin{aligned}(\mathbf{AM}') = {} & (\theta k_1 + \phi k_2 + b\cos\alpha)\mathbf{k} + (r\cos\theta - b\sin\alpha\sin\theta)\mathbf{i} \\ & + (r\sin\theta + b\sin\alpha\cos\theta)\mathbf{j} + \exp[(\theta+\phi)\mathbf{k}\times](\mathbf{B}_0\mathbf{M}_0) \end{aligned} \tag{6}$$

This is an expression of the product $H(A,\mathbf{k},p_1;\theta)T(\mathbf{s};b)H(B,\mathbf{k},p_2;\phi)$, which is an element of the set $\mathscr{H}(A,\mathbf{k},p_1)\mathscr{T}(\mathbf{s})\mathscr{H}(B,\mathbf{k},p_2)$. In the HPH-||-PPP mechanism, for any point M, the displacements of the moving platform have to satisfy Eq. (6) and to be a translation. This is achieved for $\phi = -\theta$. As a matter of fact, (**BM**) must keep unchanged, that is, $(\mathbf{B}'\mathbf{M}') = (\mathbf{B}_0\mathbf{M}_0)$. Then, Eq. (6) becomes

$$\begin{aligned}(\mathbf{AM}') = {} & [\theta(k_1 - k_2) + b\cos\alpha]\mathbf{k} + [r(cos\theta - 1) - b\sin\alpha\sin\theta]\mathbf{i} \\ & + (r\sin\theta + b\sin\alpha\cos\theta)\mathbf{j} + (\mathbf{AM}_0), \end{aligned} \tag{7}$$

It is straightforward to verify that the end-effector point B that coincides with B_0 at the home configuration moves on a ruled helicoid with the reduced pitch $(k_1 - k_2)$ and the axis $(A,\mathbf{k})$. This helicoid has the parametric representation:

$$\begin{aligned}(\mathbf{AB}') &= r(\cos\theta\mathbf{i} + \sin\theta)\mathbf{j} + b\sin\alpha(-\sin\theta\mathbf{i} + \cos\theta\mathbf{j}) + b\cos\alpha\mathbf{k} + \theta(k_1 - k_2)\mathbf{k} \\ &= (\mathbf{AN}_\theta) + b\mathbf{s}_\theta + \theta(k_1 - k_2)\mathbf{k} \end{aligned} \tag{8}$$

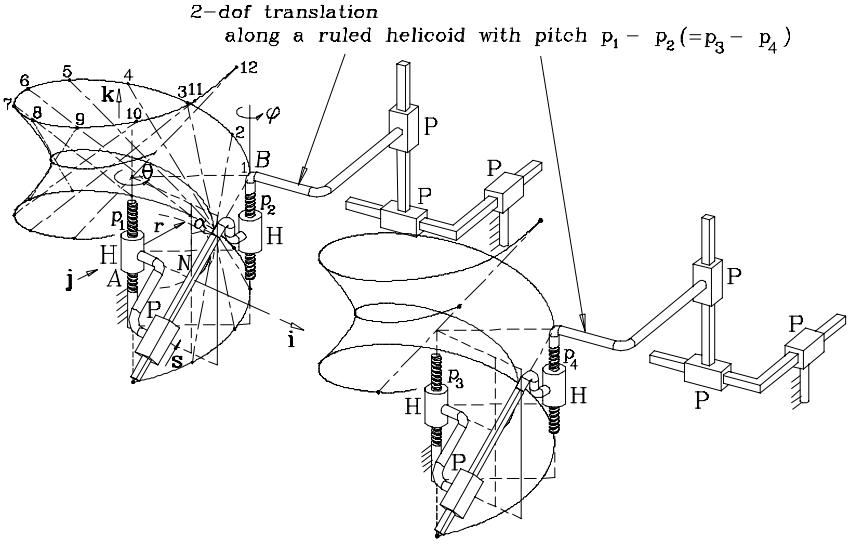


Fig. 3 Two HPH-||-PPP mechanisms generating translation along the same ruled helicoid.

where N_θ and $\mathbf{s}_\theta$ are respectively the outcomes of N and $\mathbf{s}$ by a rotation of angle θ around the axis $(A, \mathbf{k})$.

In the translation expressed by Eq. (7), any point of the end-effector moves on a ruled helicoid, which is congruent by translation with the helicoid of B. In other words, the end-effector moves by a 2-DoF translation along a ruled helicoid with pitch $p = p_1 - p_2$; in the general case $p_1 \neq p_2$. In a special case, $p_1 = p_2 = p$. Then, the end-effector translates along a ruled helicoid of pitch 0, which is a revolute hyperboloid of one sheet. This hyperboloid is a doubly ruled surface. Each family of straight lines of the hyperboloid is called a regulus. One can turn one regulus into the other by reflecting in a plane that contains the axis of revolution.

3 Mechanical Generators of 2-DoF Translation along a Ruled Helicoid

Two mechanical generators of 2-DoF translation, which are congruent by any given translation, produce the same translational motion of their end effectors [10]. In the two congruent manipulators, the two end-effectors move with the same motion and, consequently, can be rigidly connected. More than two congruent limbs can connect the fixed base to the end effector.

In Figure 3, two HPH-||-PPP mechanisms produce the same ruled-helicoid translation with the pitch $p_1 - p_2 = p_3 - p_4$. That way, rigidly connecting the two end-

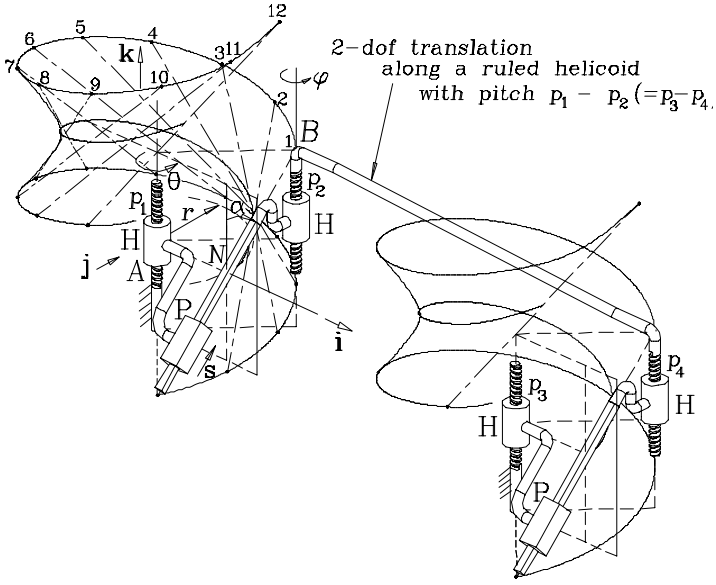


Fig. 4 HPH-||-HPH generator of 2-DoF translation along a ruled helicoid.

effectors and removing the PPP subchains lead to a parallel HPH-||-HPH generator of 2-DoF translation along the surface of a ruled helicoid, Figure 4. This is the general case with $p_1 - p_2 = p_3 - p_4$. One may notice that locking one or two of the two Ps, the 2-DoF mechanism becomes a Delassus HHHH parallelogram, which is movable with one DoF [12].

4 Mechanical Generators of 2-DoF Translation along a Revolute Hyperboloid

The HPH-||-PPP chain with equal H pitches ($p_1 = p_2 = p$) is the special case. Then, the ruled heloicoid degenerates into a revolute one-sheet hyperboloid. Two HPH-||-PPP mechanisms generating the translation along the same revolute one-sheet hyperboloid are considered. The four H pitches have to satisfy $p_1 = p_2 = p$ and $p_3 = p_4 = q$. A new HPH-||-HPH manipulator that is depicted in Figure 5 is synthesized through the gluing of the end effectors and the removal of the superfluous PPP limbs.

As any revolute surface, the revolute hyperboloid of one sheet is symmetric with respect to any plane containing its axis of revolution. Consequently, that hyperboloid is a doubly ruled surface. From that, the double mechanical generation of the 2-DoF translation along a revolute hyperboloid of one sheet is derived and a new architectural type of HPH-||-HPH parallel manipulator is synthesized and shown

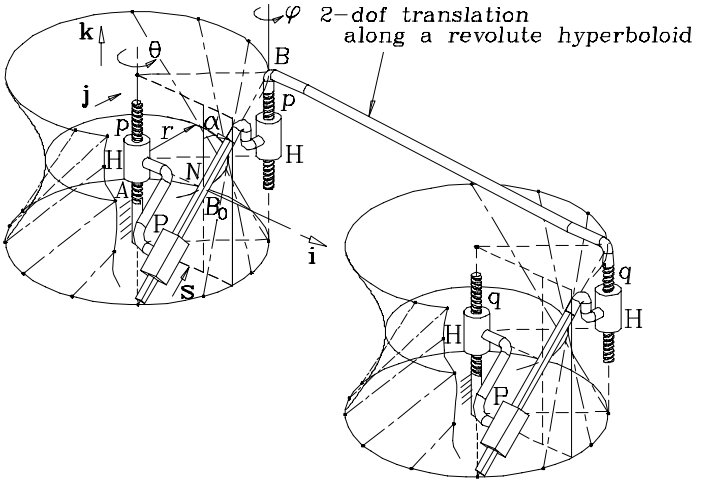


Fig. 5 HPH-||-HPH mechanism generating hyperboloidal 2-DoF translation.

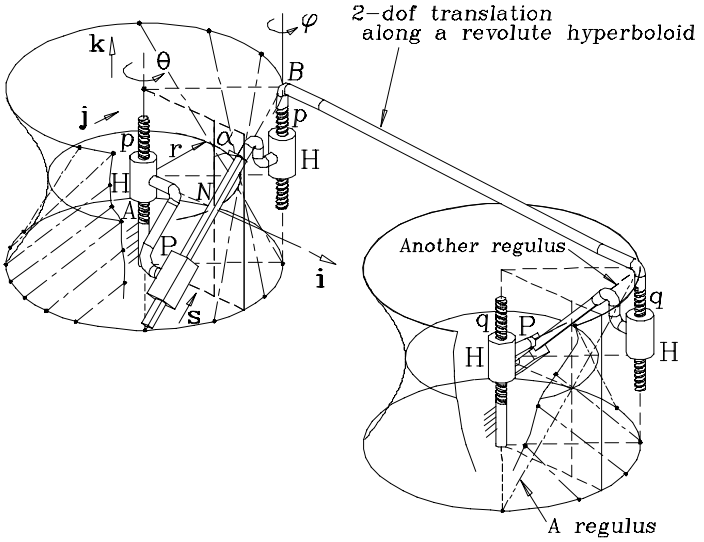


Fig. 6 New type of HPH-||-HPH generator of 2-DoF hyperboloidal translation.

in Figure 6. It is worth noticing that, in the new manipulator of Figure 6, the four pitches satisfy $p_1 = p_2 = p$ and $p_3 = p_4 = q$. In practical applications, the four H pitches can be zero and we obtain a RPR-||-RPR parallel manipulator. This 2-DoF manipulator can be actuated in the two fixed R pairs whereas the manipulators of Figures 4 and 5 cannot.

5 Conclusions

A parallel HPH-||-HPH manipulator generating, in the general case, 2-DoF translation along a ruled helicoid is synthesized via the invariance by conjugation of any submanifold of translations in the commutative Lie group $\mathscr{T}$. In a special case of the screw pitches, a mechanical generator of 2-DoF translation along a revolute hyperboloid of one sheet is found out. More new generators of 2-DoF translation along a hyperboloid are synthesized using the double geometric generation of the revolute one-sheet hyperboloid by the rotation of a straight line. In further work, the mechanisms revealed in the paper will lead to a new derivation of the Delassus plane-symmetric HPHP linkage, which is movable with one DoF.

Acknowledgements The authors are very thankful to the National Science Council for supporting this research under grant NSC 98-2221-E-151-019.

References

1. Clavel, R.: Dispositif pour le déplacement et le positionnement d'un éléments dans l'espace. Swiss Patent No. 672089A5 and International Patent (PCT) No. WO 87/03538 (1987).
2. Hervé, J. M., Sparacino, F.: Structural synthesis of parallel robots generating spatial translation. In: *Proc. 5th IEEE Int. Conf. on Advanced Robotics*, Pisa, Italy, pp. 808–813 (1991).
3. Wenger, P., Chablat, D.: Kinematic anlaysis of a new parallel machine tool: The orthoglide. In: Lenarcic, J., Stanisic, M. M. (Eds.), *Advances in Robot Kinematics*, pp. 305–314. Kluwer Academic Publishers (2000).
4. Gao, F., Li, W., Zhao, X., Jin, Z., Zhao, H.: New kinematic structures for 2-, 3-, 4-, and 5-DOF parallel manipulator designs. *Mech. Mach. Theory*, **37**, 1395–1411 (2002).
5. Kong, X.-W., Gosselin, C.: Kinematics and singularity analysis of a novel type of 3-CRR 3-DOF translational parallel manipulator. *Int. J. Robot. Res.*, **21**, 791–798 (2002).
6. Angeles, J.: The qualitative synthesis of parallel manipulators. *ASME J. Mech. Design*, **126**, 617–624 (2004).
7. Lee, C.-C., Hervé, J. M.: Translational parallel manipulators with doubly planar limbs. *Mech. Mach. Theory*, **41**, 433–455 (2006).
8. Lee, C.-C., Hervé, J. M.: Cartesian parallel manipulators with pseudoplanar limbs. *ASME J. Mech. Design*, **129**, 1256–1264 (2007).
9. Lee, C.-C., Hervé, J. M.: On some applications of primitive Schönflies-motion generators. *Mech. Mach. Theory*, **44**, 2153–2163 (2009).
10. Hervé, J. M.: Conjugation in the displacement group and mobility in mechanisms. *Trans. Can. Soc. Mech. Eng.*, **33**, 3–14 (2009).
11. Lee, C.-C., Hervé, J. M.: Type synthesis of primitive Schoenflies-motion generators. *Mech. Mach. Theory*, **44**, 1980–1997 (2009).
12. Lee, C.-C., Hervé, J. M.: On the Delassus parallelogram. In: Lenarcic, J., Wenger, P. (Eds.), *Advances in Robot Kinematics: Analysis and Design*, pp. 439–449. Springer, Dordrecht (2008).

Worm-Like Robotic Locomotion in Flexible Environment

D. Zarrouk[1], I. Sharf[2] and M. Shoham[3]

[1]*Technion-I.I.T., Haifa, Israel; e-mail: zadavid@tx.technion.ac.il*
[2]*McGill University, Montreal, Canada; e-mail: inna.sharf@mcgill.ca*
[3]*Technion-I.I.T., Haifa, Israel; e-mail: shoham@technion.ac.il*

Abstract. Miniature crawling robots for medical purposes have been the issue of many studies in the past decade. The main challenge of this type of locomotion is the high flexibility of the tissue of biological vessels combined with the usually low friction coefficients. However, little has been done in the field of analyzing the interaction between the tissue and the robot. In a previous study, we presented the influence of the *local compliance* effects on the locomotion and developed the efficiency as a function of the number of cells, friction coefficients and tangential compliance. In this study we include the influence of the *structural compliance* on the locomotion analysis of two-cell worm robot.

Key words: Medical robotics, miniature worm robots, locomotion efficiency, local and structural compliance.

1 Introduction

Over the past decade, several research groups around the world have worked on developing small worm-like robots for medical purposes, for example [1–13]. The robots are designed to crawl inside the gastrointestinal (GI) tract, blood vessels and respiratory-system vessels. Due to the high compliance of the biological tissue, the locomotion efficiency defined as the actual advance divided by the stroke of the robot is substantially reduced.

Most studies propose a robot design and experimentally test its locomotion efficiency. Some, e.g. [3, 4, 8, 9, 11, 13], study the efficiency of locomotion based on a simplified model of the mechanical properties of the environment, in particular, a homogeneous medium, without hysteresis and simple modeling of the elasticity. Dario et al. [3,4] calculate the overall efficiency by multiplying the individual efficiencies of the main stages of locomotion. However, this analysis implies that the different parts of the cycle are completely independant. The *cumulative* loss of stroke is first addressed by Kwon et al. [7] who model the loss of stroke for a two cell worm in linearly elastic environment. Zarrouk et al. [13] developed the theory of locomotion of worms for multiple cells taking into account external resisting forces

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_9, © Springer Science+Business Media B.V. 2010

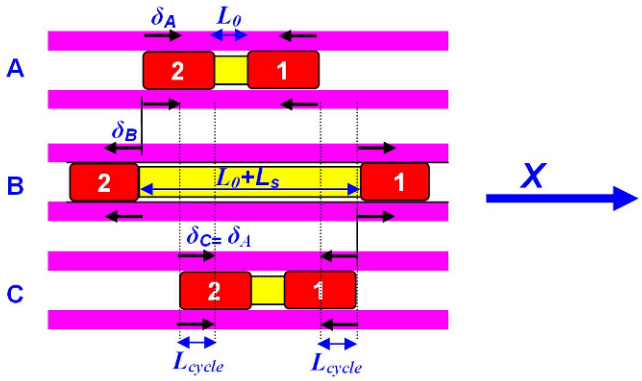


Fig. 1 Two-cell earthworm advancing in flexible environment.

and back sliding effects. However, both studies are based on *local compliance* analysis, i.e., the deformation of the tissue is local to every specific cell and does not influence the other cells.

In this paper we introduce the locomotion efficiency of a two cell worm in *structural analysis*, that is, the deformation of the tissue is large enough to reach and influence the deformation of the other cells.

2 Contact Compliance Analysis

In order to illustrate the problem of loss of efficiency of a worm when crawling in a compliant surface, consider one cycle of motion of two-cell earthworm, moving in a straight compliant environment. In this particular example, we assume that no back sliding takes place (higher friction coefficient in the back direction). We also adopt the linear tangential compliance model of the environment, which we write as:

$$F_t = k_t \delta \tag{1}$$

where δ is the tangential deflection, F_t the tangential force and k_t is the tangential compliance. We also assume the Coulomb model of friction, where the maximum static and sliding friction forces are given by:

$$F_t = \mu F_n \tag{2}$$

where μ is the friction coefficient and F_n is the normal force. The normal force F_n may be determined from the mechanical properties of the robot and the environment or by performing experiments. For typical earthworms whose diameter is smaller than that of the environment, it is the weight per cell of the robot.

The locomotion cycle of a two-cell earthworm as illustrated in Fig. 1 is composed of two main stages. Starting with the initial configuration (A) where L_0 represents

the initial distance between the two cells and the deflections are $\delta_A^{(1)} = -\delta_A^{(2)} = \mu F_n/k_t$, we have:

Stage A-B: Robot extends its body to advance the fore-cells by L_s. Now we see that the deflections have switched directions and caused a loss of efficiency. At the end of configuration (B), the deflections are $\delta_B^{(2)} = -\delta_B^{(1)} = \mu F_n/k_t$.

Stage B-C: Robot retracts its body to advance the back cells by L_s. This position is similar to configuration (A) but shifted right by a distance of L_{cycle}.

At the end of the cycle, the robot has effectively advanced through $L_{cycle} = L_s - 2\left|\delta_A\right| - 2\left|\delta_B\right|$. Since the deflections in this case are a function of the tangential contact force only (the deflections are independent), then $\left|\delta_A\right| = |\delta_B|$ and the theoretical advance is

$$\eta = \frac{\textit{effective advance}}{\textit{theoretical advance}} = 1 - \frac{4\mu F_n}{L_s k_t} \tag{3}$$

For the general case, where the robot may partially clamp its moving cells, the advance in a full cycle is [13]:

$$\eta \approx L_s - nr = L_s - \frac{2n}{k_t(n-m-1)}\left(F_n\mu_f - \frac{F_{ext} + mF_n\mu_b}{n-m}\right) \tag{4}$$

where n is the number of cells of the robot, m is the number of cells sliding back, F_{ext} is the external force acting on the robot, F_n is the normal contact force between the cell and the surface, μ_b and μ_f are respectively the backward and forward friction coefficients. The value of m is defined as the smallest integer satisfying the following inequality:

$$m > \frac{-2\delta_{ext} + n\delta_f + n\delta_b}{\delta_f - \delta_b} \tag{5}$$

where $\delta_b = \mu_b\, F_n/k_t$, $\delta_f = \mu_f\, F_n/k_t$, and $\delta_{ext} = F_{ext}/k_t$.

3 Structural Compliance Analysis

In structural analysis, we assume that the tissue deformation associated with each cell is not limited to the vicinity of that cell but can influence the deformations associated with the other cells. Therefore, the environment of the robot deforms as a continuous structure under the forces exerted by the robot. This problem is encountered in crawling locomotion inside the intestines.

To proceed with the analysis applicable to the intestinal tract, we consider the physical nature of the GI environment and distinguish between the intestinal walls and the mesenteries of the GI tract (see Fig. 2). Similarly to the treatments in [12, 13], we assume that the elastic properties of the intestines are linear and we neglect the effects of the normal deflections on the locomotion. Furthermore, we make use of a discrete model of the environment where the intestines and mesenteries are

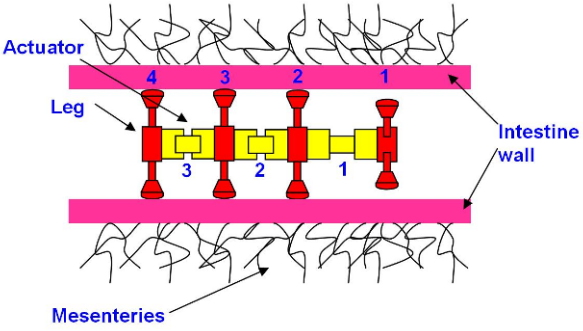


Fig. 2 A robot inside the intestines.

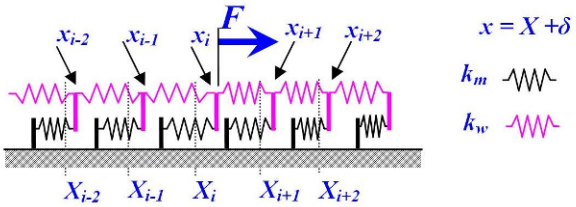


Fig. 3 Modeling of the intestines and mesenteries as two sets of springs.

modeled as two sets of springs (see Fig. 3). The first set of springs, representing the flexibility of the mesenteries, is connected on one end to a fixed rigid support, while the other end is connected to a second set of springs representing the walls of the intestines. The stiffness of the springs representing the mesenteries and the intestinal wall are respectively k_m and k_w.

3.1 Model of Intestines

We now develop a mathematical model of the intestine, specifically, a relationship for the tangential deflection δ of the intestinal wall as a function of the applied force F. For the analysis presented here, we employ X to denote the undeformed coordinate along the intestine so that X locates points on the tissue when there is no load or deformation; δ_i is the tangential deflection of the point at X_i and x_i is the deformed coordinate, i.e., the actual position of the tissue after deformation, $x_i = X_i + \delta_i$ (see Fig. 3).

Assuming that the force is applied at point i, the tangential deflection of point j can be calculated as a function of the neighboring deflections by enforcing equilibrium at j as:

$$-k_m\delta_j - k_w\left(\delta_j - \delta_{j-1}\right) + k_w\left(\delta_{j+1} - \delta_j\right) = 0 \tag{6}$$

In addition, the equilibrium of the whole structure requires that:

$$k_m \sum \delta_j = F \tag{7}$$

To transition from the discrete model above to a continuous description of the system, we observe that, the coefficient k_m can be calculated from:

$$k_m = K_m \Delta X \tag{8}$$

where K_m is the elasticity/distance of the mesenteries. By analogy with the axially loaded medium of sectional area A and Young modulus E, we can write the coefficient k_w as:

$$k_w = \frac{AE}{\Delta X} \tag{9}$$

Inserting Eqs. (8) and (9) into (6) we obtain:

$$\frac{1}{\Delta X}\left(\left(\frac{\delta_{j+1} - \delta_j}{\Delta X}\right) - \left(\frac{\delta_{-}\delta_{j-1}}{\Delta X}\right)\right) = \frac{K_m}{AE}\delta_j \tag{10}$$

and recognizing the expression on the left-hand side as a central difference approximation of the second derivative of δ, we obtain the continuous form of (10):

$$\frac{\partial^2 \delta}{\partial X^2} = \frac{K_m}{AE}\delta(X) \tag{11}$$

The only physical solution of the above differential equation is:

$$\delta(X) = Ce^{-\left(\frac{K_m}{AE}\right)^{1/2}|X|} \tag{12}$$

The continuous analogue of Eq. (7) becomes:

$$\int_{-\infty}^{\infty} K_m \delta(X)\, dX = F \tag{13}$$

from which, upon substitution for $\delta(X)$ from Eq. (12), the constant C can be extracted. This results in the final solution for the deformation as:

$$\delta(X) = \left(\frac{1}{AEK_m}\right)^{1/2} \frac{F}{2} e^{-\left(\frac{K_m}{AE}\right)^{1/2}|X|} \tag{14}$$

The above indicates that δ is a linear function of the force acting on the tissue and the *structural deformation* is an exponentially decaying function of the distance.

3.2 Superposition Principle

Since the developed model is linear, when more than one force is applied to the structure, say F_i at location X_i, the principle of superposition is applied, i.e., the tangential deflection is the sum of the tangential deflections caused by the individual forces.

$$\delta(X_j) = \sum_i \delta(X_i, X_j, F_i) \tag{15}$$

4 Two-Cell Robot Example

We now apply the analysis of the previous section to determine the deflections of and the forces acting on a two-cell crawling robot. This system, when moving quasi-statically in a flexible environment is statically indeterminate; therefore, we must combine the equilibrium relation for the system with the deformation "history" as derived in Section 3.1.

4.1 Analysis

Consider now a robot with two cells only, moving freely on structurally flexible environment where the tangential compliance of the surface is modeled as per Eq. (15) but now rewritten in a more compact form:

$$\delta(X) = \frac{F\, e^{-\beta|X|}}{k_t} \tag{16}$$

with the corresponding definitions of β and k_t. We also assume that the backward friction coefficient is much larger than the forward friction coefficient. In quasi-static analysis, the forces acting on the two cells, $F^{(1)}$ and $F^{(2)}$ are of opposite sign but of equal magnitude:

$$\left|F^{(1)}\right| = \left|F^{(2)}\right| = \mu_f F_n \tag{17}$$

The positions of the two cells are constrained by the kinematics of the robot, that is:

$$\left(X^{(1)} + \delta^{(1)}\right) - \left(X^{(2)} + \delta^{(2)}\right) = L_0 + L_s \tag{18}$$

when the actuator is extended or

$$\left(X^{(1)} + \delta^{(1)}\right) - \left(X^{(2)} + \delta^{(2)}\right) = L_0 \tag{19}$$

when the actuator is contracted, respectively. The tangential deflections for the two legs can then be evaluated from Eq. (17) and superimposing the effects of the two forces. This gives the following two non-linear equations:

$$\begin{bmatrix} \delta^{(1)} \\ \delta^{(2)} \end{bmatrix} = \frac{1}{k_t} \begin{bmatrix} 1 & e^{-\beta|X|} \\ e^{-\beta|X|} & 1 \end{bmatrix} \begin{bmatrix} F^{(1)} \\ -F^{(2)} \end{bmatrix} \tag{20}$$

which together with Eq. (18) (or Eq. (19)) can be solved numerically for the unknowns: $\delta^{(1)}$, $\delta^{(2)}$ and $X^{(2)}$ ($X^{(1)}$ can be arbitrarily set to zero).

4.2 Approximate Closed-Form Solution

To obtain an approximate but explicit value of the locomotion efficiency, we assume that the difference of the undeformed coordinates of the two cells can be approximated from Eqs. (18) and (19), with the deflections of the two cells set to their "local compliance" values, $\left|\delta^{(1)}\right| = \left|\delta^{(2)}\right| = \mu\ F_n/k_t$. Then for the two stages A and B (see Fig. 1):

$$\begin{aligned} X_A^{(1)} - X_A^{(2)} &= L_0 + \frac{2\mu_f\ F_n}{k_t} \\ X_A^{(1)} - X_A^{(2)} &= L_0 + L_s - \frac{2\mu_f\ F_n}{k_t} \end{aligned} \tag{21}$$

and substituting for the above into Eqs. (20) to compute the corresponding deflections, $\delta_A = |\delta_A^{(1)}| = |\delta_A^{(2)}|$ and $\delta_B = |\delta_B^{(1)}| = |\delta_B^{(2)}|$, we obtain the distance traveled during a cycle as:

$$\begin{aligned} L_{cycle} &= L_s - 2\delta_A - 2\delta_B \\ &\approx L_s - 4\frac{\mu_f F_n}{k_t} + 2\frac{\mu_f F_n}{k_t}\left(e^{-\beta\left(L_0 + 2\frac{\mu_f F_n}{k_t}\right)} + e^{-\beta\left(L_0 + L_s - 2\frac{\mu_f F_n}{k_t}\right)}\right) \end{aligned} \tag{22}$$

The above analytical result shows that compared to the local compliance analysis, which produces the first two terms in the expression on the right of Eq. (22), structural flexibility inmproves the efficiency of locomotion as it reduces the deflections under the cells.

4.3 Comparison to Exact Solution

In Fig. 4, the approximate value of the locomotion efficiency is compared to the exact solution based on Eqs. (17–20), as a function of β. As can be seen from Fig. 4, the two are in excellent agreement, the maximum difference between the two being 0.4%. When β becomes small, the robot behaves as if there is no flexibility at all, while when β increases, the locomotion efficiency approaches the value predicted with the local compliance analysis. However, as already noted, the structural com-

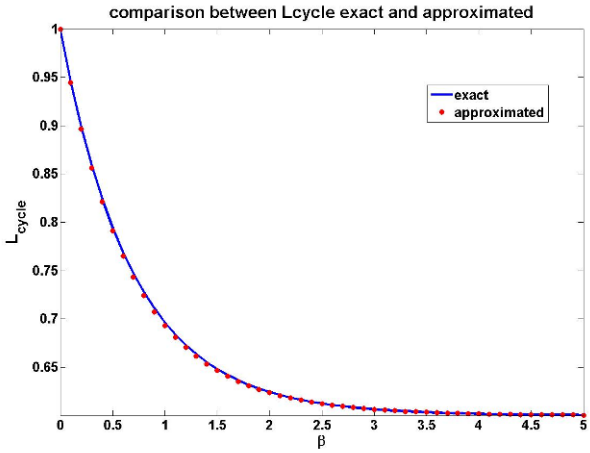


Fig. 4 Comparison between the exact solution of Eqs. (17–20) and the approximate solution (22).

pliance improves the efficiency of locomotion by reducing the deflections under the cells.

5 Conclusions

The research described in the present paper focuses on the analysis of worm robotic locomotion in a flexible environment. Differently from our previous analysis, we considered the influence of the structural compliance, rather than, the local compliance. The intestine-like environment (intestinal wall and mesentery) is modeled with two sets of linear springs and the model is applied to the two-cell worm locomotion.

The efficiency of the locomotion including the influence of the structural compliance is calculated. Interestingly, it was found that the structural compliance increases the efficiency of locomotion compared to the case where only local compliance is considered. An approximate closed form solution is also obtained and compared to exact value of the locomotion efficiency. The authors' future work will be devoted to generalizing the analysis of the behavior of robotic worms for general number of cells, under influence of external force and different friction coefficients.

References

1. Asari, V.K., Kumar, S., Kassim, I.M.: A Fully autonomous microrobotic endoscopy System. *Journal of Intelligent and Robotic Systems*, **28**, 325–341 (2000).
2. Chi, D., Yan, G.: From wired to wireless: A miniature robot for intestinal inspection. *Journal of Medical Engineering and Technology*, **27**(2), 71–76 (2003).

3. Dario, P., Ciarletta, P., Menciassi, A., Kim, B.: Modelling and experimental validation of the locomotion of endoscopic robots in the colon. In *Experimental Robotics VIII, STAR 5*, pp. 445–453, Springer-Verlag, Berlin/Heidelberg (2003).
4. Dario, P., Ciarletta, P., Menciassi, A., Kim, B.: Modeling and experimental validation of the locomotion of endoscopic robots in the colon. *International Journal of Robotics Research*, **23**(4–5), 549–556 (2004).
5. Kim, B., Lee, S., Park, J.H., Park, J.O.: Inchworm-like microrobot for capsule endoscope. In *Proceedings of the 2004 IEEE International Conference on Robotics and Biomimetics*, pp. 458–463 (2004).
6. Kim, B., Lee, M.G., Lee, Y.P., Kim, Y.I., Lee, G.H.: An Earthworm-like micro robot using shape memory alloy actuator. *Sensors and Actuators A*, **125**, 429–437 (2006).
7. Kwon, J., Park, S., Park, J., Kim, B.: Evaluation of the critical stroke of an earthworm-like robot for capsule endoscopes. In *Proceedings of the Institution of Mechanical Engineers, Part H (Journal of Engineering in Medicine)*, **221**, 397–405 (2007).
8. Menciassi, A., Accoto, D., Gorini, S., Dario,P.: Development of a biomimetic miniature robotic crawler. *Autonomous Robots*, **21**(2), 155–163 (2006).
9. Phee, L., Menciassi, A., Accoto, D., Stefanini, C., Dario, P.: Analysis of robotic locomotion devices for the gastrointestinal tract. In *Robotics Research, STAR 6*, pp. 467–483 (2003).
10. Phee, L., Accoto, D., Menciassi, A., Stefanini, C., Carrozza, M.C., Dario, P.: Analysis and development of locomotion devices for the gastrointestinal tract. *IEEE Transactions on Biomedical Engineering*, **49**(6), 613–616 (2002).
11. Wang, K., Yan, G., Jiang P., Ye, D.: A wireless robotic endoscope for gastrointestine. *IEEE Transactions on Robotics*, **24**(1), 206–210 (2008).
12. Wang, K., Yan, G.: Micro robot prototype for colonoscopy and in vitro experiments. *Journal of Medical Engineering and Technology*, **31**(1), 24–28 (2007).
13. Zarrouk, D., Sharf, I., Shoham, M.: Analysis of earthworm-like robotic locomotion on compliant surfaces. In *Proceedings IEEE International Conference on Robotics and Automation*, Alask, accepted for publication (2010).

Actuation Strategy Based on the Acceleration Model for the 3-PRPR Redundant Planar Parallel Manipulator

Maurizio Ruggiu[1] and Juan A. Carretero[2]

[1]*Department of Mechanical Engineering, University of Cagliari, Piazza d'Armi, 09123 Cagliari, Italy; e-mail: ruggiu@dimeca.unica.it*
[2]*Department of Mechanical Engineering, University of New Brunswick, Fredericton, NB, E3B 5A3, Canada; e-mail: juan.carretero@unb.ca*

Abstract. In this paper a new actuation strategy is proposed and then applied to the 3-PRPR[1] redundant planar parallel manipulator. The actuation strategy consists of selecting adequate actuators displacements such that the 2-norm of the joint space acceleration vector is minimised in each step of a given trajectory. The method is based on the use of constrained local optimisation algorithms. Given the redundant nature of the manipulator, in the optimisation routine three of the six prismatic actuator displacements are selected as design variables. Thus, in each step of the calculation, velocities and accelerations of the three selected actuators can be calculated while the displacements, velocities and accelerations of the remaining actuators are obtained by solving the inverse problems. Some results are shown by comparing the joint acceleration history for the 3-PRPR manipulator with a similar non-redundant manipulator (the 3-RPR).

Key words: Redundant parallel manipulator, optimisation, path planning, joint acceleration.

1 Introduction

Parallel manipulators are extensively studied due to their numerous advantages in comparison to their serial counterparts. The advantages can be summarised in high structural rigidity, high payload-to-weight ratio and relatively high accuracy, amongst other. However, it is well know that they also have numerous drawbacks such as complexity in the forward kinematic equations, relatively small workspace, singularities inside their workspace, to name a few. Some of the cited drawbacks, namely the singularities inside their workspace and a generally small workspace, may be overcome by the use of redundancy. In parallel manipulators, there are three basic types of redundancy: kinematic, actuation and branch redundancy. This paper deals with kinematic redundancy (see [1–6]) which consists of adding extra active joints and links to one or more branches of the mechanism.

In the case of kinematic redundancy, whether it is in serial or parallel manipulators, the added mobility of the mechanism allows for an enlarged workspace while helping avoid most singular configurations [1, 3, 7]. Conversely, the motion

[1] P denotes an actuated prismatic joint and R denotes a passive revolute joints.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_10, © Springer Science+Business Media B.V. 2010

planning problem of kinematically redundant manipulators increases significantly in complexity. Indeed, in this case there are infinite number of solutions for the inverse displacement problem providing infinite number of combinations of actuators displacements which guarantee a specific end-effector pose (i.e., position and orientation). Therefore, when dealing with a trajectory, the choice of actuation strategy can follow different criteria as examined in [2, 4, 8, 9], to name a few.

Redundancy in serial manipulators has been used, amongst others, to select trajectories free of collisions (see for instance [8, 9]) or for avoiding singular configurations (see for instance [7]). In the context of parallel manipulators, both Cha et al. [1] and Ebrahimi et al. [2] looked into selecting the best actuator displacements for a manipulator to remain as far as possible from singular configurations.

In this paper, an alternative actuation strategy based on minimising the actuator accelerations is proposed. Minimising the actuators' accelerations provides some bounds for the actuators forces without being required to develop a complete dynamic model of the manipulator which would be specific to predefined link geometries. The method consists in choosing a subset of actuator displacements as free variables for a numerical optimisation routine while the joint velocities, accelerations and remaining joint displacements are calculated using the kinematic equations for the manipulator. The numerical optimisation then seeks for the actuator displacements which will minimise the actuators' accelerations based on the displacement history and the desired current end effector pose. The method is explained through an example using the 3-PRPR redundant planar parallel manipulator for which the acceleration model is shown next (Section 2). Then, in Section 3, the motion planning strategy is described in detail. Numerical examples are provided in Section 4 while conclusions are presented in Section 5.

2 Acceleration Model Formulation

The 3-PRPR kinematically redundant planar parallel manipulator consists of 3 identical legs each with a couple of PR pairs in series (Fig. 1). The symmetrical version of the manipulator has the moving platform shaped as an equilateral triangle and the direction of the prismatic joints attached to the base frame at angles of $2\pi/3$ from each other. For simplicity, the first prismatic joint on the first leg is aligned with the inertial Y axis. The acceleration model of the manipulator was obtained by a vector mechanics approach [10, 11] and is briefly outlined here.

According to Fig. 1, the following vector-loop closure equation can be obtained for leg i:

$$\overline{OB}_i = \mathbf{p} + {}^{O}\mathbf{R}_P \mathbf{h}_i \tag{1}$$

where $\mathbf{p}$ is the vector from the origin of frame $\{O\}$ to the origin of frame $\{P\}$, ${}^{O}\mathbf{R}_P$ is the rotation matrix representing the orientation of frame $\{P\}$ relative to frame $\{O\}$ and $\mathbf{h}_i$ is a vector from the origin of frame $\{P\}$ to point B_i expressed in terms of frame $\{P\}$. Denoting by $\overline{OB}_i$ the position vector representing the position of point

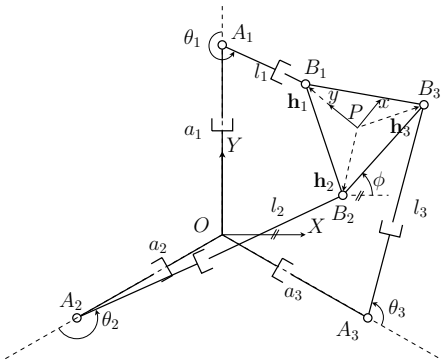


Fig. 1 The 3-PRPR redundant planar parallel manipulator

B_i relative to the origin of frame $\{O\}$, the following expression can be obtained

$$\overline{OB}_i = \mathbf{a}_i + \mathbf{l}_i = a_i \mathbf{e}_{a_i} + l_i \mathbf{e}_{l_i} \tag{2}$$

where $\|\mathbf{a}_i\| = a_i$ and $\|\mathbf{l}_i\| = l_i$ while vectors $\mathbf{e}_{a_i}$ and $\mathbf{e}_{l_i}$ are the unit vectors in the direction of $\mathbf{a}_i$ and $\mathbf{e}_i$, respectively.

Taking the time derivative of equations (1) and (2) yields the linear velocity of point B_i (i.e., $\mathbf{v}_{B_i}$):

$$\mathbf{v}_{B_i} = \mathbf{v}_P + \dot{\phi}\mathbf{z} \times {}^O\mathbf{R}_P \mathbf{h}_i \tag{3}$$

$$\mathbf{v}_{B_i} = \dot{a}_i \mathbf{e}_{a_i} + \dot{l}_i \mathbf{e}_{l_i} + (\omega_i \mathbf{z} \times l_i \mathbf{e}_{l_i}) \tag{4}$$

where $\mathbf{v}_P$ is the linear velocity of point P, $\dot{\phi}$ is the angular velocity of the moving platform relative to the base frame while ω_i is the angular velocity of the direction vector of the distal prismatic joint (i.e., the angular velocity of vector $\mathbf{e}_{l_i}$ which, according to Fig. 1, would correspond to $\dot{\theta}_i$). The angular velocity vector $\omega_i \mathbf{z}$ can be obtained by cross-multiplying both sides of equation (4) by $\mathbf{e}_i$:

$$\omega_i \mathbf{z} = \frac{1}{l_i}(\dot{a}_i \mathbf{e}_{a_i} \times \mathbf{e}_{l_i} - \mathbf{v}_{B_i} \times \mathbf{e}_{l_i}) \tag{5}$$

Equation (5) can now be substituted into equation (4). By developing the triple vector product, a simplified expression can be easily obtained:

$$\dot{l}_i \mathbf{e}_{l_i} + (\dot{a}_i \mathbf{e}_{l_i} \cdot \mathbf{e}_{a_i})\mathbf{e}_{l_i} - (\mathbf{e}_{l_i} \cdot \mathbf{v}_{B_i})\mathbf{e}_{l_i} = 0 \tag{6}$$

It can be noted that all three vector terms in equation (6) are along $\mathbf{e}_{l_i}$. Therefore, equation (6) provides 3 scalar equations for the manipulator (one for each leg).

Equation (3) is now differentiated with respect to time to obtain $\mathbf{a}_{B_i}$ (the linear acceleration of point B_i) in terms of $\mathbf{a}_P$ (the linear acceleration of point P). That is:

$$\mathbf{a}_{B_i} = \mathbf{a}_P + (\ddot{\phi}\mathbf{z} \times {}^O\mathbf{R}_P\mathbf{h}_i + \dot{\phi}\mathbf{z} \times (\dot{\phi}\mathbf{z} \times {}^O\mathbf{R}_P\mathbf{h}_i)) \tag{7}$$

Likewise, differentiating equation (4) with respect to time, $\mathbf{a}_{B_i}$ can be expressed as:

$$\mathbf{a}_{B_i} = \ddot{a}_i\mathbf{e}_{a_i} + (\dot{\omega}_i\mathbf{z} \times l_i\mathbf{e}_{l_i}) + 2(\omega_i\mathbf{z} \times \dot{l}_i\mathbf{e}_{l_i}) + \ddot{l}_i\mathbf{e}_{l_i} + \omega_i\mathbf{z} \times (\omega_i\mathbf{z} \times l_i\mathbf{e}_{l_i}) \tag{8}$$

The angular acceleration of the distal prismatic joint, i.e., $\dot{\omega}_i\mathbf{z}$, can be obtained by cross-multiplying equation (8) by $\mathbf{e}_{l_i}$ resulting in:

$$\dot{\omega}_i\mathbf{z} = \frac{1}{l_i}(\ddot{a}_i\mathbf{e}_{a_i} \times \mathbf{e}_{l_i} - 2\omega_i\dot{l}_i\mathbf{z} - \mathbf{a}_{B_i} \times \mathbf{e}_{l_i}) \tag{9}$$

This expression of $\dot{\omega}_i\mathbf{z}$ is now substituted into equation (8). By developing the triple vector product a simplified expression for the acceleration can be obtained, too:

$$(\ddot{a}_i\mathbf{e}_{l_i} \cdot \mathbf{e}_{a_i})\mathbf{e}_{l_i} + \ddot{l}_i\mathbf{e}_{l_i} - (\mathbf{e}_{l_i} \cdot \mathbf{a}_{B_i})\mathbf{e}_{l_i} - \omega_i^2 l_i\mathbf{e}_{l_i} = 0 \tag{10}$$

Similar to the linear velocity case, all the terms in equation (10) are vectors in the direction $\mathbf{e}_{l_i}$ thus providing 3 scalar equations for the manipulator (one for each leg).

3 Motion Planning

Here, a new motion planning strategy is proposed where actuator displacements are optimised to reduce the actuators' accelerations. The procedure can be briefly summarised as follows:

1. *Define search variables*: For a manipulator with n actuators (with $n > d$), select $n-d$ as the 'redundant actuators' where d is the dimension of the task space. The displacement of these redundant actuators will be used as the decision (design variables) in the trajectory optimisation.
2. *Define trajectory*: The desired trajectory of the end effector is selected. Using the inverse displacement solution [6], an arbitrary initial pose for the manipulator is selected.
3. *Define bounds*: Based on the current manipulator pose, some user-defined maximum joint rates and a predefined update rate for the simulation, determine the maximum and minimum displacement limits for the actuators which will act as bounds while optimising actuator displacements.
4. *Optimise next actuator move*: A non-linear constrained optimisation is performed using the 2-norm of the joint space acceleration vector as objective function.
   a. For every optimisation step, the d non-optimised actuator displacements are calculated from the trial values of the $n-d$ optimised actuators using the inverse displacement equations.

b. Using a portion of the time history of the actuator displacements, estimate the acceleration of each of the $n-d$ redundant actuators.
c. Using the estimated redundant actuator accelerations, the acceleration model for the manipulator and the end-effector velocity and acceleration, compute the acceleration of the remaining d actuators.
d. Determine the objective function value as the 2-norm of the n-dimensional actuator acceleration vector.

For the 3-PRPR manipulator described earlier, where $n = 6$ and $d = 3$, this procedure can be explained as follows. For a given end effector motion, the values of six variables (i.e., a_i, l_i, $i = 1,2,3$) are to be assigned amongst $\mathbb{R}^3$ possibilities. Therefore, three of them (i.e., $n-d = 6-3$) can be selected using an optimisation procedure (i.e., search variables) whilst the others are obtained from the inverse displacement problem (IDP) solution. The IDP can be solved combining equations (1) and (2) (see [6] for details). At this time, for any selected value of the search variables (i.e., the displacement of the $n-d$ redundant actuators), the joint rates (i.e., their time derivatives), are fixed, thus providing only one solution for the inverse velocity and acceleration problems. Accelerations of the non-optimised actuators are thus obtained from equation (10) with equation (5). Continuity of the accelerations profiles was guaranteed by a polynomial fitting of the joints rates.

The objective function $\mathscr{N}$ for the 3-PRPR is given as:

$$\mathscr{N} = \sqrt{\sum_{i=1}^{3} (\ddot{a}_i^2 + \ddot{l}_i^2)} \tag{11}$$

It may be observed that, for the the 3-PRPR manipulator, $\mathscr{N}$ is homogeneous in dimension because of the prismatic nature of all its actuators. This property would no longer be valid if a mixture of revolute and prismatic joints were used as variables since some arbitrary scaling factors would need to be added to the objective function. Also, in general the use of passive prismatic joints is ill-advised as they are often a source of error and failures due friction and even stick-slip phenomena.

4 Numerical Examples

The path planning strategy described in Section 3 was implemented in Matlab for the 3-PRPR manipulator. In the current numerical examples, the displacements of the proximal prismatic actuators of each leg, i.e., a_i for $i = 1,2,3$, were used as the redundant actuators while the distal actuators (i.e., l_i for $i = 1,2,3$) where considered the non-redundant actuators. All actuators, redundant or not, were allowed to vary in the range between 0.5 and 5$[l.u.]$[2] with the actuator joint rates limit in the range between -4.5 and $+4.5$ $[actuator\ l.u.]/[path\ l.u.]$. The path followed by the reference point P on the end effector is schematically shown in Fig. 2. More specifically

[2] The abbreviation $l.u.$ denotes length unit whereas $t.u.$ denotes time unit.

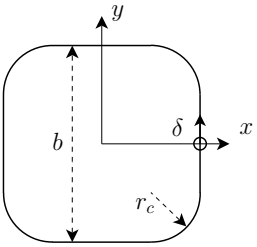


Fig. 2 Type of trajectories used in the numerical examples.

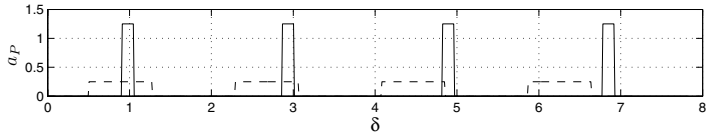


Fig. 3 End effector total Cartesian acceleration a_P in $[l.u.]/[t.u.]^2$ as a function of the path displacement δ in $[l.u.]$: trajectory 1 $(-)$ and trajectory 2 $(--)$.

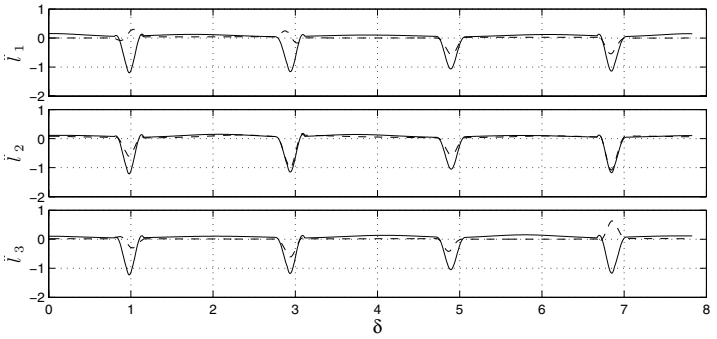


Fig. 4 Distal actuator acceleration $(\ddot{l}_i)$ in $[l.u.]/[t.u.]^2$ as a function of the path displacement δ in $[l.u.]$: non-redundant case $(-)$ and redundant case $(--)$ (trajectory 1).

and with reference to Fig. 2, the parameters for the two trajectories are: Trajectory 1: $b = 2[l.u.]$, $r_c = 0.1[l.u.]$; Trajectory 2: $b = 2[l.u.]$, $r_c = 0.5[l.u.]$. Both trajectories were followed at an arbitrary constant velocity $v_P = 11.2[l.u.]/[t.u.]$ while keeping the moving platform with constant orientation (i.e., $\phi = 0.2$ [rad]). Figure 3 shows the magnitude of the Cartesian acceleration vector, i.e., $a_P = \sqrt{\ddot{x}^2 + \ddot{y}^2}$, for the reference point of the end effector while performing trajectories 1 and 2 as function of the path displacement δ. When performing trajectory 1 the maximum value reached by a_P is $a_{P_{\max}} = 1.25[l.u.]/[t.u.]^2$, while $a_{P_{\max}} = 0.25[l.u.]/[t.u.]^2$ for trajectory 2.

Figure 4 shows the comparison between $\ddot{l}_i$ values obtained from an optimisation routine with a_i as the optimised variables and from a non-redundant case with a_i fixed at $a_1 = a_2 = a_3 = 1$. The comparison is made for trajectory 1. According to Fig. 4, the redundant manipulator can travel the trajectory with actuator acceleration

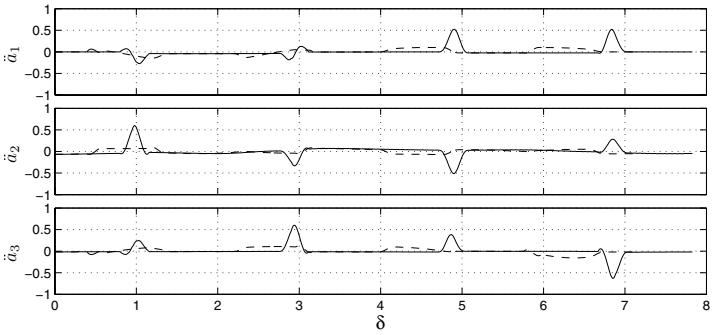


Fig. 5 Proximal actuator acceleration ($\ddot{a}_i$) in $[l.u.]/[t.u.]^2$ as a function of the path displacement δ in $[l.u.]$: trajectory 1 ($-$) and trajectory 2 ($--$) (redundant case).

values significantly lower than the non-redundant manipulator. In order to measure the difference the joints accelerations vector containing the maximum of the absolute value for each actuator is defined as:

$$\ddot{q}_{\max}|_{nr_j} = \left[\, |\ddot{l}_1|_{\max} \;\; |\ddot{l}_3|_{\max} \;\; |\ddot{l}_3|_{\max} \,\right] \tag{12}$$

$$\ddot{q}_{\max}|_{r_j} = \left[\, |\ddot{a}_1|_{\max} \;\; |\ddot{a}_2|_{\max} \;\; |\ddot{a}_3|_{\max} \;\; |\ddot{l}_1|_{\max} \;\; |\ddot{l}_2|_{\max} \;\; |\ddot{l}_3|_{\max} \,\right] \tag{13}$$

respectively for the non-redundant (sub-index *nr*) and redundant manipulators (sub-index *r*). Subscript *j* in equations (12) and (13) indicates the trajectory number. In the case of trajectory 1 the maximum values are:

$$\ddot{q}_{\max}|_{nr_1} = \left[\, 1.20 \quad 1.22 \quad 1.23 \,\right] [l.u.]/[t.u.]^2$$

$$\ddot{q}_{\max}|_{r_1} = \left[\, 0.52 \quad 0.60 \quad 0.63 \quad 0.58 \quad 1.08 \quad 0.63 \,\right] [l.u.]/[t.u.]^2$$

From comparison between $\ddot{q}_{\max}|_{nr_1}$ and $\ddot{q}_{\max}|_{r_1}$, it can be seen that the optimisation procedure can reduce the mean value of $|\ddot{l}_i|_{\max}$ of about 37%.

Figure 5 shows the comparison between the proximal actuators accelerations when the end effector is travelling either trajectory 1 or trajectory 2 both in the non-redundant and redundant cases. The ratio between the mean of the elements of the vectors in equation (13) (denoted by an overline) is defined for the redundant and non-redundant manipulators, such that: $G_r = \overline{\ddot{q}_{\max}|_{r_1}} / \overline{\ddot{q}_{\max}|_{r_2}}$ and $G_{nr} = \overline{\ddot{q}_{\max}|_{nr_1}} / \overline{\ddot{q}_{\max}|_{nr_2}}$. These ratios highlight the measure of the response of the optimisation for different trajectories. In the present comparison, greater numbers mean higher acceleration when smaller radii are used. For the current trajectories, these ratios result in:

$$G_{nr} = 6.3, \quad G_r = 4.6 \tag{14}$$

It can be deduced from these ratios that the optimisation procedure in trajectory 1 provides a significantly better joint trajectory than in trajectory 2 (even though the

latter already shows a significant improvement from the non-redundant case). Also important to note is that there is a reduction of 24% in average acceleration when using the redundant manipulator in trajectory 2 whereas this reduction is significantly higher (i.e., 44%) when the radius is smaller in trajectory 1.

5 Conclusions

As part of a research programme dedicate to propose and test different actuation strategies for kinematically redundant parallel manipulator, in this paper a new actuation strategy for kinematically redundant parallel manipulators is proposed and then applied to the 3-PRPR redundant planar parallel manipulator whose end effector travels at constant velocity two square trajectory with rounded corners. The actuation strategy is based on the minimisation of the actuated joints' acceleration. More specifically, the 2-norm of the joints accelerations vector was chosen as the objective function. Two important conclusions can be drawn from the work: a) The optimisation procedure for the redundant architecture considered significantly reduces the actuated joint accelerations with respect to those calculated for a similar non-redundant manipulator and, most importantly, b) The optimisation procedure exhibits much greater impact when it is most important, i.e., whenever the end effector Cartesian accelerations are greater.

References

1. Cha, S., Lasky, T.A. and Velinsky, S.A.: Kinematically-redundant variations of the 3-RRR mechanism and local optimization-based singularity avoidance. *Mechanics Based Design of Structures and Machines*, **35**(1), 15–38 (2007).
2. Ebrahimi, I., Carretero, J.A. and Boudreau, R.: Kinematic analysis and path planning of a new kinematically redundant planar parallel manipulator. *Robotica*, **26**(3), 405–413 (2008).
3. Ebrahimi, I., Carretero, J.A. and Boudreau, R.: 3-PRRR redundant planar parallel manipulator: Inverse displacement, workspace and singularity analyses. *Mechanism and Machine Theory*, **42**(8), 1007–1016 (2007).
4. Merlet, J.-P.: Redundant parallel manipulators. *Lab. Robotics and Aut.* **8**(1), 17–24 (1996).
5. Wang, J. and Gosselin, C.M.: Kinematic analysis and design of kinematically redundant parallel mechanisms. *Journal of Mechanical Design* **126**(1), 109–118 (2004).
6. Ruggiu, M. and Carretero, J.A.: Kinematic analysis of the 3-PRPR redundant planar parallel manipulator. In: *Proceedings of the CCToMM*, Quebec City (2009).
7. Nakamura, Y.: *Advanced Robotics: Redundancy and Optimization*. Addison Wesley (1991).
8. Maciejewski, A.A. and Klein, C.A.: Obstacle avoidance for kinematically redundant manipulators in dynamically variant environments. *Int. J. of Robotics Research* **3**(3), 109–117 (1985).
9. Nakamura, Y. and Yoshikawa, T.: Task-priority based redundancy control of robot manipulators. *International Journal of Robotics Research* **6**(2), 3–15 (1987).
10. Tsai, L.-W.: Solving the inverse dynamics of a Stewart–Gough manipulator by the principle of virtual work. *Journal of Mechanical Design*, **122**, 3–9 (2000).
11. Tsai, L.-W.: *Robot Analysis*. John Wiley and Sons, New York (1999).

Kinematics and Design of a 5-DOF Parallel Robot Used in Minimally Invasive Surgery

D. Pisla, N. Plitea, B.G. Gherman, C. Vaida, A. Pisla and M. Suciu

Technical University of Cluj Napoca, Romania; e-mail: Doina.Pisla@mep.utcluj.ro

Abstract. Robotic assisted surgery is a continuously developing field, as robots have proved their utility in the operating rooms. Some of their advantages could be summed as follows: increased precision, the possibility of reaching positions and places that could prove difficulty in reaching using classical instruments, the replacement of medical personnel in the operating room, allowing for more space and cost saving etc. Until now, the vast majority of robots used in surgery, had serial structures. A new parallel hybrid architecture is proposed in this paper, the kinematics of this new structure is determined, the singularities are discussed and the design of the robot is presented.

Key words: Parallel hybrid robot, minimally invasive surgery, kinematics, design.

1 Introduction

The latest advancements achieved in both medicine and science allowed the minimization or elimination of the surgical incisions. The surgical methodology for minimal incision or minimally invasive surgery (MIS) greatly enhances the outcome of surgery by shortening the recovery time and reducing the pain and trauma by minimizing the tissue and muscle damages incurred from incision.

One of the first robots created for medical purposes is AESOP, produced by Computer Motion. They have developed several versions of such a robot, until they finally developed ZEUS Robotic Surgical System with three robotic arms attached to the side of the operating table [1]. The most known commercial robot is the daVinci, a highly versatile, but also large and very expensive robot [2]. A highly articulated robot for minimally invasive surgery threading through tightly packed volumes without disturbing the surrounding organs or tissues is presented in [3]. It is actually a snake-like robot whose primarily use is for cardiac surgery. EndoAssist [4] is another camera manipulator having four degrees of freedom produced by Armstrong Healthcare Limited, UK and commercially available. The interesting fact about EndoAssist is the control: the robot is programmed to detect and follow the movements of the surgeon's head. LAPMAN [5] is a dynamic laparoscope holder with three degrees of freedom, guided by a joystick clipped onto the laparoscopic instruments.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_11, © Springer Science+Business Media B.V. 2010

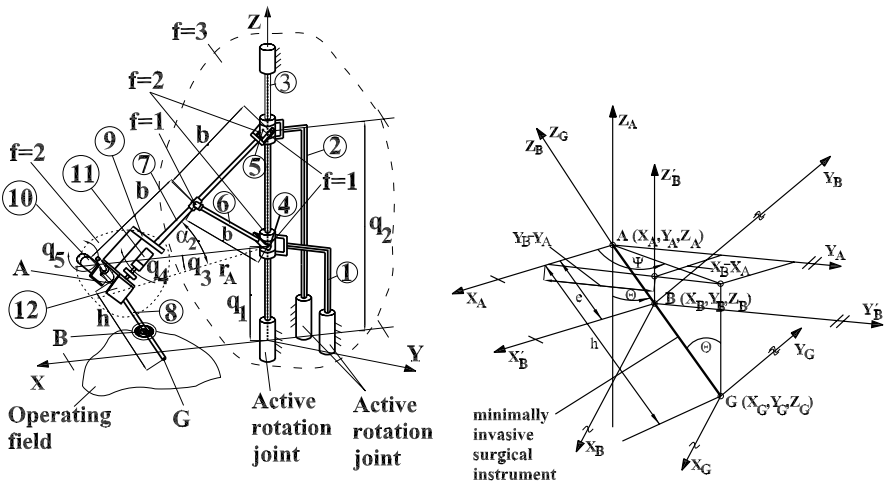
Just like other camera holders, it confers an optimal control over the visual field, the elimination of the human tremor, it helps to reduce the number of assistants in the operating room. A camera manipulating robot with 3 degrees of freedom is presented in [6]. The KaLAR robot needs an endoscope holder in order to place and hold the robot in a fixed place on the operating table. There are 2 degrees of freedom used to bend the end of the endoscope: one for the left/right motion and one for the up/down motion. The third degree of freedom is used to obtain the in/out motion of the camera.

A hybrid surgical robot model with a ball joint was studied in [7]. The forward kinematics was built using the Denavit-Hartenberg method combined with successive rotation transformation, while for the inverse kinematics the geometry method was used. In [8] the kinematics and concept designs of a reconfigurable 5 DOF hybrid robot is presented. The robot is composed of a 2-DOF parallel spherical mechanism that is serially connected with a 3-DOF open loop kinematic chain via a prismatic joint. The kinematics of a 3-DOF parallel robot for camera guiding was studied in [9]. The problems generated by this structure consists in the fact that the laparoscope will put some pressure on the abdominal muscles and in order to reduce the effects of this aspect, an orientation module with two degrees of freedom was introduced for a better guiding of the camera inside the patient, as it is presented in this paper.

2 The Surgical Parallel Robot

Among the main requirements surgeons impose for a surgical robot used for minimally invasive surgery are the following: the robot control has to be accurate, the robot has to be stable and rigid in the operating room, the robot should have low size, maximum speed 1 cm/s, robot precision 1 mm, maximum rotation (right/left): 90° etc. The robot presented in [9] had actually all of these characteristics, but from practice it has been proved that there are situations when the abdominal wall is under a lot of pressure in certain positions from the laparoscope, requiring a mechanism that will hold the camera and release some of that pressure. In the same time, the structure presented in this paper is a developing project, meaning that the camera will be replaced by an active instrument used, among other, for cutting, suturing, grasping.

The structure presented in this paper has five degrees of freedom: three degrees of freedom for general positioning of the camera and an orientation module, so that the camera will not have to rely on the abdominal wall. The robot structure is based on an already registered patent [10]. The robot has a hybrid parallel mechanisms assuring high rigidity due to its closed chains. There is a restriction the mechanism has to accomplish: the laparoscope will have to go through point B (X_B, Y_B, Z_B) representing the point of incision of the laparoscope in the abdomen. This restriction will have important consequences on the control of the mechanism, as it will be seen. The parallel mechanism has three rotational actuated joints. Through a ball-screw mech-



a) The robot structure. b) The rotation angles ψ and θ.

Fig. 1 PARASURG 5M structure and rotation angles ψ and θ.

anism ((1) and (2)), the rotation motions are transformed into translation motions. There are also the following passive joints: two cylindrical joints (between links (1) and (3), respectively between links (2) and (3)), two prismatic joints (between links (4) and (3), respectively (5) and (3)), three rotary joints (between links (4) and (6), (5) and (7) respectively (6) and (7)). At the end of element (7) the PARASURG 5M (Fig. 1a) structure the motor (10) which represents the q_4 active coordinate and is mounted on element (9). At the end of motor (10), on the output shaft there is mounted the element (12) on which there is mounted the motor (11), denoting the q_5 active coordinate. At the end of the element (12) there is mounted the element (8). The geometrical parameters of the parallel robot are represented by b, h, X_B, Y_B, Z_B. Angles ψ and θ are also presented in Fig. 1b. The laparoscope can be positioned in any point of the surgical field using 5-DOF of the robot. The advantage of such a structure as compared with the robots presented in [4] and [11] is a better guidance of the camera and the avoidance of certain pressure on the abdominal wall. Through its design, PARASURG 5M could be used either for laparoscope guidance or for positioning and orientation of an active instrument.

3 The Geometric Model

To obtain the direct geometric model, a second, mobile frame AX'Y'Z', attached to the endoscope in the point A is necessary. As one can see from Fig. 1, r_A has the following form:

$$r_a = \sqrt{(2 \cdot b)^2 - (q(2) - q(1))^2}, \tag{1}$$

where q_1, q_2, q_3, q_4, q_5 are the active coordinates of the robot. There can be defined a relation between some of these coordinates and the coordinates of point A(X_A, Y_A, Z_A):

$$X_A = r_A \cdot \cos(q_3), Y_A = r_A \cdot \sin(q_3), Z_A = q_1. \tag{2}$$

The direct geometric model

The displacements in motor coordinates are known and according to these the end-effector coordinates will be determined, meaning the coordinates of point G(X_G, Y_G, Z_G) - the tip of the endoscope, and the two rotation angles ψ and θ. One must take into consideration that the laparoscope must pass through the fixed point B. From equations (1) and (1) the coordinates of point A(X_A, Y_A, Z_A) can be obtained. By using Fig. 1, the angular rotation angles ψ and θ can be determined. Two cases are yielding:

Case 1. If $X_A = X_B$ and $Y_A = Y_B$, it yields:

$$X_G = X_A, Y_G = Y_A, Z_G = Z_A - h, \psi = 0, \theta = 0. \tag{3}$$

Case 2. If $X_A \neq X_B$ and $Y_A \neq Y_B$, it yields:

$$X_G = X_A + h \cdot \sin(\psi) \cdot \cos(\theta), Y_G = Y_A + h \cdot \cos(\psi) \cdot \cos(\theta), Z_G = Z_A - h \cdot \cos(\theta),$$

$$\psi = \arctan 2(Y_B - Y_A, X_B - X_A), \theta = \arctan 2\left(\sqrt{(X_B - X_A)^2 + (Y_B - Y_A)^2}, Z_A - Z_B\right). \tag{4}$$

The inverse geometric model

In this case we have as inputs the position of the tip of the laparoscope (point G(X_G,Y_G,Z_G)), having to find the active coordinates: q_1, q_2, q_3, q_4 and q_5. As with the direct geometric model, there are two cases to consider:

Case 1. If $X_A = X_B$ and $Y_A = Y_B$, it yields:

$$X_A = X_G, Y_A = Y_G, Z_A = Z_G - h, \psi = 0, \theta = 0. \tag{5}$$

Case 2. If $X_A \neq X_B$ and $Y_A \neq Y_B$, it yields:

$$X_A = X_G - h \cdot \sin(\psi) \cdot \cos(\theta), Y_A = Y_G + h \cdot \cos(\psi) \cdot \cos(\theta), Z_A = Z_G + h \cdot \cos(\theta),$$

$$\psi = \arctan 2(Y_B - Y_A, X_B - X_A), \theta = \arctan 2\left(\sqrt{(X_B - X_A)^2 + (Y_B - Y_A)^2}, Z_A - Z_B\right). \tag{6}$$

For both cases, from (1), and (2), the coordinates will be obtained:

$$q_1 = Z_A, q_2 = q_1 + \sqrt{(2 \cdot b)^2 - (X_A^2 + Y_A^2)}, q_3 = \arctan 2(Y_A, X_A). \tag{7}$$

For q_4 and q_5, we have:

$$\begin{cases} q_4 = \arctan 2(\cos(\theta), \cos(\psi - q_3) \cdot \sin(\theta)) - \alpha_2, \\ q_5 = \arctan 2(\sin(q_4 + \alpha_2) \cdot \sin(\psi - q_3) \cdot \sin(\theta), \cos(\theta)), \end{cases} \tag{8}$$

where

$$\alpha_2 = \arctan 2\left((q_2 - q_1), \sqrt{(2 \cdot b)^2 - (q_2 - q_1)^2}\right). \tag{9}$$

4 Kinematics

To obtain the kinematic equations, we have to start from the geometric model. For PARASURG 5M, we have the following implicit equations defining the tip of the endoscope:

$$\begin{cases} f_1 = Z_G + h \cdot \cos(\theta) - q_1, \\ f_2 = X_G - h \cdot \cos(\psi) \cdot \sin(\theta) - \sqrt{(2 \cdot b)^2 - (q_2 - q_1)^2} \cdot \cos(q_3), \\ f_3 = Y_G - h \cdot \sin(\psi) \cdot \sin(\theta) - \sqrt{(2 \cdot b)^2 - (q_2 - q_1)^2} \cdot \sin(q_3), \\ f_4 = \sin(\theta) \cdot \cos(\psi - q_3) - \cos(q_5 \cdot \cos(q_4 + \alpha_2), \\ f_5 = \sin(\theta) \cdot \sin(\psi - q_3) - \sin(q_5). \end{cases} \tag{10}$$

Using the matrix representation, the kinematic model is:

$$A \cdot \dot{X} + B \cdot \dot{q} = 0, \tag{11}$$

where $\dot{q} = [\dot{q_1}\ \dot{q_2}\ \dot{q_3}\ \dot{q_4}\ \dot{q_5}]^T$ are the driving velocities and $\dot{X} = [\dot{X_G}\ \dot{Y_G}\ \dot{Z_G}\ \dot{\psi}\ \dot{\theta}]^T$ are the end-effector velocities and angular velocities. From relation (11), both the *direct kinematic model (DKM)* and the *inverse kinematic model (IKM)* were determined. After the calculations were achieved, an analytical solution for both geometric and kinematic model resulted due to its relatively simple mathematical model.

5 Singularity Analysis

The algorithm used for the singularity analysis is based on deriving the determinants for the two Jacobian matrices obtained from the inverse and direct geometric model.

Type I singularities. First type of singularities occurs when the determinant of the Jacobian matrix B is 0, case in which the robot loses one or more degrees of freedom. The determinant of the B matrix is:

$$\det(B) = (q_2 - q_1) \cdot (\cos(q_5))^2 \cdot \sin(q_4 + \alpha_2). \tag{12}$$

We have the following situations:

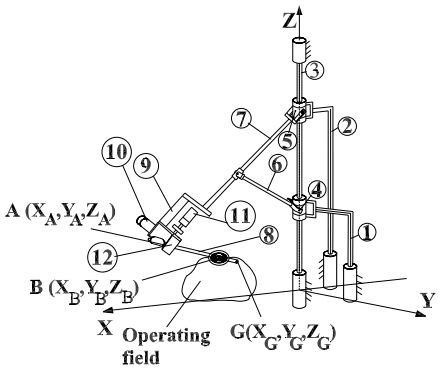


Fig. 2 Singularity positions for PARASURG 5M.

1. If $q_2 - q_1 = 0$, it yields $q_2 = q_1$, situation when the two elements having the lengths b and 2b are overlapped. Constructively, the mechanical structure of the robot will impose $q_2 > q_1$.
2. If $\cos(q_5) = 0$, we have: $\theta = \frac{\pi}{2}$, case in which the laparoscope is situated in a plane parallel with the XOY plane at point A. The robot loses one degree of freedom.
3. If $\sin(q_4 + \alpha_2) = 0$, we have also: $\theta = \frac{\pi}{2}$ and $\psi = q_3$, which is a particularization of the case discussed at point no. 2). The robot loses one degree of freedom.

Type II singularities. Second type of singularities occurs when the determinant of the Jacobian matrix A is 0. The determinant of the A matrix is:

$$\det(A) = -\sin(\theta) \cdot \cos(\theta). \tag{13}$$

1. The case when $\sin(\theta) = 0$ was taken as a separate case in kinematics.
2. It was observed that $\cos(\theta) = \cos(q_5) \cdot \sin(q_4 + \alpha_2)$, so when $\cos(\theta) = 0$, we obtain the same case discussed at type I singularities no. 2 and 3.

Type III singularities, so-called architectural singularities, take place when both Jacobian determinants are 0. One can observe that the matrices A and B are both 0 when $\sin(q_4 - \alpha_2) = 0$ and/or $\cos(q_5) = 0$, cases that should be avoided since the design stage. From Fig. 2 one can see that PARASURG 5M reaches the position of singularity at the top of the operating field, as well when the robot arm is 90° from the X axis.

6 Design of PARASURG 5M

All motors of the 5-DOF robot are rotation motors from MAXON [12], the translation motion being obtained by using two ball-screw mechanisms. In order to have a rotation and translation along the Z axis, a secondary shaft along the main guiding

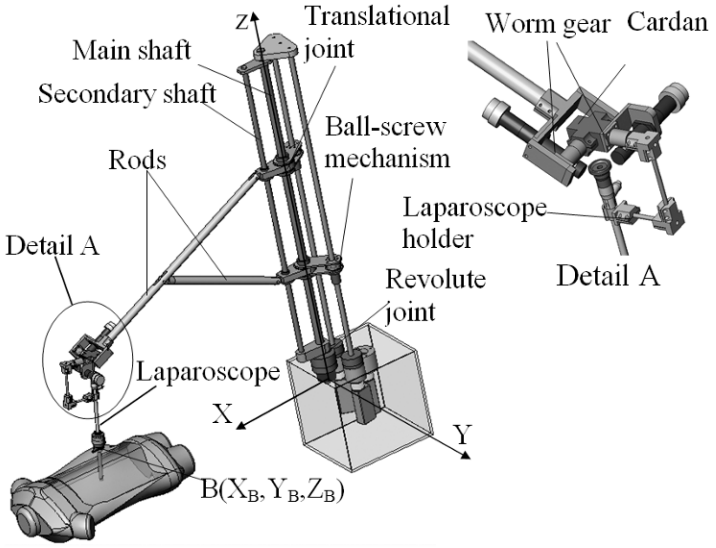


Fig. 3 PARASURG 5M as laparoscope holder.

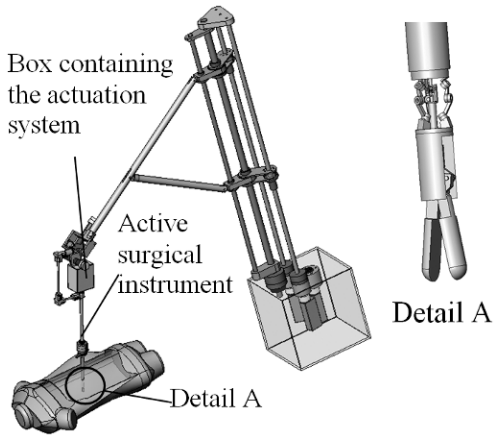


Fig. 4 PARASURG 5M as positioning module of an active surgical instrument.

shaft was added, as in Fig. 3. The three motors in the robot base have 60 W and the gear planetary transmission has a ratio of 1:26 for the motors that actuate the ball-screw mechanisms and 1:936 for the other. For the orientation mechanism, the motors have 6 W and a planetary gear ratio of 1:4592. A worm-gear transmission has been used in order to obtain a smoother motion. The overall dimensions of the robot are: 1000 mm $\times$ 1000 mm $\times$ 300 mm. Figure 4 presents the case when instead of the camera it is used an active surgical instrument.

7 Conclusions

The contribution of this paper consists in the development of a new, simple, parallel hybrid structure for minimally invasive surgery, aiming towards an active instrument manipulator. The advantage of this new parallel structure from the kinematic point of view is that its both direct and inverse kinematic models have been obtained through an analytical approach. Code sequences and sensors could be implemented in order to prevent the laparoscope to reach the positions of singularity. However, the results obtained from kinematics of the robot can be successfully implemented into a real-time control algorithm. Future work includes the study of the robot stiffness, evaluating the costs as well as designing and building an active surgical instrument.

Acknowledgements The research work reported here was financed by the PNCDI-2 P4 Grant, entitled "Multidisciplinary development of surgical robots using innovative parallel structures". It has been awarded by the Ministry of Education, Research, Youth and Sports of Romania.

References

1. Brown University, Division of Biology and Medicine http://biomed.brown.edu (2009).
2. Intuitive Surgical, http://www.intuitivesurgical.com/ (2009).
3. Degani, A., Wolf, A., Zenati, M.: Highly articulated robotic probe for minimally invasive surgery. *Robotics and Automation*, Orlando, pp. 4167–4172 (2006).
4. Halin, N., Loula, P., Aarnio, P.: Experiences of using the EndoAssist robot in surgery. In *Studies in Health Technologies and Informatics*, Tampere, Finland, pp. 125-161-3 (2007).
5. Pollet, R., Donnez, J.: Using a Laparoscope Manipulator (LAPMAN®) in Laparoscopic Gynecological Surgery. In *Surgical Technology International XVII – Gynecology*, San Francisco, pp. 187–191 (2008).
6. Saing, V., Sotthivirat, S., Vilasrussamee, R.N., Suthakorn, J.: Design of a new laparoscopic-holder assisting robot. In *Proceedings of the 3rd International Symposium on Biomedical Engineering*, Bangkok, Thailand, pp. 278–281 (2008).
7. Diao, Y., Yao, L., Chen, Z., Zang, J., Luo, H.: Configuration and kinematic analysis of a novel hybrid surgical robot with a ball joint. In *Intelligent Systems and Applications*, ISA, Wuhan, pp. 1-4 (2009).
8. Liu, H., Huang, T., Mei, J., Zhao, X., Derek, H.: Kinematic design of a 5-DOF hybrid robot with large workspace/limb-stroke ratio. *Journal of Mechanical Design* **129**, 530–537 (2009).
9. Pisla, D., Plitea, N., Gherman, B., Pisla, A., Vaida, C.: Kinematical analysis and design of a new surgical parallel robot. In *Computational Kinematics*, Duisburg, Germany, pp. 273–282 (2009).
10. Plitea, N., Pisla, D., Vaida, C., Gherman, B.: Robot chirurgical. Patent Pending No. a00525/7.07.2009, Romania (2009).
11. Pisla, D., Plitea, N., Vaida, C.: Kinematic modeling and workspace generation for a new parallel robot used in minimally invasive surgery. In *Advances in Robot Kinematics: Analysis and Design*, Batz-sur-Mer, France, pp. 459–468 (2008).
12. Maxon Motor AG, http://www.maxonmotor.com (2009).

Main Theorem on Schönflies-Singular Planar Stewart Gough Platforms

G. Nawratil

Institute of Discrete Mathematics and Geometry,
Vienna University of Technology, Austria;
e-mail: nawratil@geometrie.tuwien.ac.at

Abstract. Parallel manipulators which are singular with respect to the Schönflies motion group $\mathsf{X}(\mathsf{a})$ are called Schönflies-singular, or more precisely $\mathsf{X}(\mathsf{a})$-singular, where a denotes the direction of the rotary axis. A special class of such manipulators are architecturally singular ones because they are singular with respect to any Schönflies group. Another remarkable set of Schönflies-singular planar parallel manipulators of Stewart Gough type was already presented by the author. In this paper we give the main theorem on $\mathsf{X}(\mathsf{a})$-singular planar parallel manipulators.

Key words: Schönflies-singular, Schönflies motion group, Stewart Gough platform, singularity.

1 Introduction

The Schönflies motion group $\mathsf{X}(\mathsf{a})$ consists of three linearly independent translations and all rotations about the infinity of axes with direction a. This 4-dimensional group is of importance in practice because it is well adapted for pick-and-place operations.

The geometry of a planar parallel manipulator of Stewart Gough type (SG type) is given by the six base anchor points $\mathsf{M}_i \in \Sigma_0$ with coordinates $\mathbf{M}_i := (A_i, B_i, 0)^T$ and by the six platform anchor points $\mathsf{m}_i \in \Sigma$ with coordinates $\mathbf{m}_i := (a_i, b_i, 0)^T$. By using Euler Parameters (e_0, e_1, e_2, e_3) for the parametrization of the spherical motion group $\mathsf{SO}(3)$ the coordinates $\mathbf{m}_i'$ of the platform anchor points with respect to the fixed space can be written as $\mathbf{m}_i' = K^{-1}\mathbf{R}\,\mathbf{m}_i + \mathbf{t}$ with

$$\mathbf{R} := (r_{ij}) = \begin{pmatrix} e_0^2+e_1^2-e_2^2-e_3^2 & 2(e_1e_2-e_0e_3) & 2(e_1e_3+e_0e_2) \\ 2(e_1e_2+e_0e_3) & e_0^2-e_1^2+e_2^2-e_3^2 & 2(e_2e_3-e_0e_1) \\ 2(e_1e_3-e_0e_2) & 2(e_2e_3+e_0e_1) & e_0^2-e_1^2-e_2^2+e_3^2 \end{pmatrix}, \qquad (1)$$

the translation vector $\mathbf{t} := (t_1, t_2, t_3)^T$ and $K := e_0^2+e_1^2+e_2^2+e_3^2$.

It is well known that a SG platform is singular if and only if the carrier lines of the prismatic legs belong to a linear line complex, or analytically seen, if $Q := det(\mathbf{Q}) = 0$ holds, where the ith row of the 6×6 matrix $\mathbf{Q}$ equals the Plücker coordinates $\underline{\mathbf{l}}_i := (\mathbf{l}_i, \widehat{\mathbf{l}}_i) := (\mathbf{m}_i' - \mathbf{M}_i, \mathbf{M}_i \times \mathbf{l}_i)$ of the ith carrier line.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_12, © Springer Science+Business Media B.V. 2010

1.1 Notation

Definition 1. Parallel manipulators which are singular with respect to the Schönflies motion group $\mathsf{X}(\mathsf{a})$ are called Schönflies-singular, or more precisely $\mathsf{X}(\mathsf{a})$-singular.

For proving the so-called main theorem on Schönflies-singular planar Stewart Gough platforms we use the notation introduced in [1]. We denote the determinant of certain $j \times j$ matrices as follows:

$$|\mathbf{X},\mathbf{y},\ldots,\mathbf{Xy}|_{(i_1,i_2,\ldots,i_j)} := det(\mathbf{X}_{(i_1,i_2,\ldots,i_j)},\mathbf{y}_{(i_1,i_2,\ldots,i_j)},\ldots,\mathbf{Xy}_{(i_1,i_2,\ldots,i_j)}) \tag{2}$$

$$\text{with} \quad \mathbf{X}_{(i_1,i_2,\ldots,i_j)} = \begin{bmatrix} X_{i_1} \\ X_{i_2} \\ \vdots \\ X_{i_j} \end{bmatrix}, \ \mathbf{y}_{(i_1,i_2,\ldots,i_j)} = \begin{bmatrix} y_{i_1} \\ y_{i_2} \\ \vdots \\ y_{i_j} \end{bmatrix}, \ \mathbf{Xy}_{(i_1,i_2,\ldots,i_j)} = \begin{bmatrix} X_{i_1}y_{i_1} \\ X_{i_2}y_{i_2} \\ \vdots \\ X_{i_j}y_{i_j} \end{bmatrix} \tag{3}$$

and $(i_1,i_2,\ldots,i_j) \in \{1,\ldots,6\}$ and pairwise distinct. Moreover it should be noted that we write $|\mathbf{X},\mathbf{y},\ldots,\mathbf{Xy}|_{i_1}^{i_j}$ if $i_1 < i_2 < \ldots < i_j$ with $i_{k+1} = i_k + 1$ for $k = 1,\ldots,j-1$ hold. Moreover the algebraic condition that $\mathsf{M}_i, \mathsf{M}_j, \mathsf{M}_k$ or $\mathsf{m}_i, \mathsf{m}_j, \mathsf{m}_k$ are collinear is denoted by $C_{(i,j,k)} := |\mathbf{1},\mathbf{A},\mathbf{B}|_{(i,j,k)} = 0$ and $c_{(i,j,k)} := |\mathbf{1},\mathbf{a},\mathbf{b}|_{(i,j,k)} = 0$, respectively.

It should also be said that in the later done case study we always factor out the homogenizing factor K if possible. Moreover we give the number n of terms of not explicitly given polynomials F in square brackets, i.e. $F[n]$.

1.2 Related Work

Special Schönflies-singular manipulators are architecturally singular ones because they are singular with respect to any Schönflies group. As architecturally singular manipulators are already classified (see [2, 3, 4, 5, 6] for the planar case and [7, 8] for the non-planar one) we are only interested in Schönflies-singular manipulators which are not architecturally singular.

For the determination of $\mathsf{X}(\mathsf{a})$-singular planar parallel manipulators we distinguish the following cases depending on the angle $\alpha \in [0,\pi/2]$ enclosed by a and the carrier plane Φ of the base anchor points and the angle $\beta \in [0,\pi/2]$ between a and the carrier plane φ of the platform anchor points. Every $\mathsf{X}(\mathsf{a})$-singular manipulator belongs to one of the following 5 cases (after exchanging platform and base):

1. $\alpha \neq \beta$: (a) $\alpha = \pi/2, \beta \in [0,\pi/2[$ (b) $\alpha,\beta \in [0,\pi/2[$
2. $\alpha = \beta$: (a) $\alpha = \pi/2$ (b) $\alpha =]0,\pi/2[$ (c) $\alpha = 0$

According to [1] the solution set of case (1a) can be characterized as follows:

Theorem 1. *A non-architecturally singular planar manipulator is* $\mathsf{X}(\mathsf{a})$*-singular, where* a *is orthogonal to* Φ *and orthogonal to the x-axis of the moving frame if and only if* $rk(\mathbf{1},\mathbf{A},\mathbf{B},\mathbf{Bb},\mathbf{a},\mathbf{b},\mathbf{Ab})_1^6 = 4$ *holds.*

For more details on the self-motional behavior of the solution set of case (1a) as well as a geometric interpretation of the given rank condition we refer to [1].

In the following Sections 2 and 3 we prove that the manipulators of Theorem 1 are the only $\mathsf{X}(\mathsf{a})$-singular ones with $\alpha \neq \beta$ which are not architecturally singular.

2 Main Theorem for the General Case

Theorem 2. $\nexists$ *non-architecturally singular planar SG platforms with no 4 collinear anchor points which are* $\mathsf{X}(\mathsf{a})$*-singular if* $\alpha \neq \beta$ *and* a *not orthogonal to* Φ *or* φ*.*

Proof. Without loss of generality (w.l.o.g.) we can assume that $\alpha > \beta$ and therefore Φ cannot be parallel to a. Then we can choose coordinate systems such that $a_2A_2B_3B_4B_5c_{(3,4,5)}(a_3-a_4)(b_3-b_4) \neq 0$ hold (cf. [2, 1]). Moreover, due to $\alpha > \beta$ we can always rotate the platform about a such that the common line of Φ and φ is parallel to $[\mathsf{M}_1,\mathsf{M}_2]$. This yields the following coordinatization: $\mathbf{M}_i = (A_i,B_i,0)$ and $\mathbf{m}_i = (a_i,b_i\cos\delta,b_i\sin\delta)$ with $A_1 = B_1 = B_2 = a_1 = b_1 = 0$ and $\sin\delta \neq 0$.

As no four anchor points are collinear we can apply the elementary matrix manipulations given by Karger [2] to the Jacobian $\mathbf{Q}$. We end up with $\underline{\mathbf{l}}_6 := (v_1,v_2,v_3,0,-w_3,w_2)$ with

$$v_i := r_{i1}K_1 + (r_{i3}\sin\delta + r_{i2}\cos\delta)K_2, \quad w_j := r_{j1}K_3 + (r_{j3}\sin\delta + r_{j2}\cos\delta)K_4$$

and

$$\begin{aligned} K_1 &:= |\mathbf{A},\mathbf{B},\mathbf{Ba},\mathbf{Bb},\mathbf{a}|_2^6, & K_3 &:= |\mathbf{A},\mathbf{B},\mathbf{Ba},\mathbf{Bb},\mathbf{Aa}|_2^6, \\ K_2 &:= |\mathbf{A},\mathbf{B},\mathbf{Ba},\mathbf{Bb},\mathbf{b}|_2^6, & K_4 &:= |\mathbf{A},\mathbf{B},\mathbf{Ba},\mathbf{Bb},\mathbf{Ab}|_2^6. \end{aligned} \tag{4}$$

Due to Lemma 2.1 given by Mick and Röschel [9] this manipulator must also be $\mathsf{X}(\mathsf{s})$-singular where s denotes the direction of the common line of Φ and φ.

In the first step we will use this property to show that $K_1 = K_2 = 0$ must hold. Therefore we can set $e_2 = e_3 = \delta = 0$ and compute $Q[4224]$ in its general form. The necessity of $K_1 = K_2 = 0$ follows immediately from $Q_{101}^{42} + Q_{101}^{24} = K_2$ and $Q_{002}^{51} + Q_{002}^{33} + Q_{002}^{15} = K_1$, where Q_{ijk}^{uv} denotes the coefficient of $t_1^i t_2^j t_3^k e_0^u e_1^v$ of Q.

Now we go back to the general case. We replace the sixth line of the Jacobian $\mathbf{Q}$ by $(v_1,v_2,v_3,0,-w_3,w_2)$ under consideration of $K_1 = K_2 = 0$. In the following we prove by contradiction that also $K_3 = K_4 = 0$ must hold. This finishes the proof because $K_1 = K_2 = K_3 = K_4 = 0$ are the 4 necessary and sufficient conditions for a planar manipulators with no 4 points on a line to be architecturally singular (cf. [2]).

Part [A] $e_2 = 0$

We set $e_1 = e_4\cos\mu$ and $e_3 = e_4\sin\mu$, where e_4 is the homogenizing factor.

Moreover $\sin\mu\cos\mu \neq 0$ must hold. Then we compute $Q[35346]$ in dependency of K_3 and K_4 and denote the coefficients of $t_1^i t_2^j t_3^k e_0^u e_4^v$ of Q by Q_{ijk}^{uv}.

First we prove by contradiction that K_4 must also vanish. Assuming $K_4 \neq 0$ we get $b_2 = 0$ from $Q_{100}^{80} = 0$. Then the resultant of Q_{100}^{71} and Q_{200}^{51} with respect to B_3 can only vanish without contradiction (w.c.) for:

1. $b_i = 0$: Then $Q_{200}^{51} = 0$ implies $B_j = B_k$ (with $i,j,k \in \{3,4,5\}$ and pairwise distinct) and $Q_{200}^{33} = 0$ yields the contradiction.
2. $B_4 = B_5$, $b_3 b_4 b_5 \neq 0$: Then $Q_{200}^{51} = 0$ can only vanish w.c. for $B_3 = B_5$ or $b_4 = b_5$.
   a. $B_3 = B_5$: We get the contradiction from $Q_{200}^{33} = 0$.
   b. $b_4 = b_5$, $B_3 \neq B_5$: In this case $Q_{100}^{71} = 0$ yields the contradiction.

Now we can set $K_4 = 0$ and compute $Q = A_2 e_4 K_3 F[15090]$. We distinguish between the following two cases for proving that F cannot vanish w.c.:

1. $b_2 \neq 0$: W.l.o.g. we can compute a_5 from $F_{110}^{41} = 0$ and A_5 from $F_{101}^{50} = 0$.
   a. Assuming $b_3 \neq b_5 \neq b_4$ we can express A_4 from $F_{100}^{70} = 0$. Then $F_{100}^{61} = 0$ yields the contradiction.
   b. W.l.o.g. we set $b_4 = b_5$. Now F_{100}^{70} can only vanish w.c. for $b_5(b_2 - b_5) = 0$. In both cases $F_{100}^{61} = 0$ yields the contradiction.
2. $b_2 = 0$: Now F_{100}^{61} can only vanish w.c. for $b_3 b_4 b_5 C_{(3,4,5)} = 0$:
   a. $b_i = 0$: Then $F_{200}^{32} = 0$ implies $B_j = B_k$ and $F_{100}^{52} = 0$ yields $A_j = A_k$ (with $i,j,k \in \{3,4,5\}$ and pairwise distinct). Finally F_{100}^{43} cannot vanish w.c..
   b. $C_{(3,4,5)} = 0$, $b_3 b_4 b_5 \neq 0$: Assuming $B_3 \neq B_4$ we can compute A_5 from the collinearity condition and a_5 from $F_{020}^{41} = 0$. Now F_{200}^{32} can only vanish w.c. for $|\mathbf{B},\mathbf{b},\mathbf{Bb}|_3^5 = 0$. W.l.o.g. we can compute b_4 from this condition. Then $F_{100}^{52} = 0$ yields the contradiction. In the special case $B_3 = B_4 = B_5$ we can compute A_5 from $F_{100}^{52} = 0$ w.l.o.g.. Then $F_{200}^{14} = 0$ already yields the contradiction.

Part [B] $e_2 \neq 0$

We set $e_1 = e_4\cos\mu$, $e_3 = e_4\sin\mu$ and $e_2 = e_4 n$, where $n\sin\mu \neq 0$ holds. Moreover for $n\cos\delta + \sin\mu\sin\delta = 0$ we can assume $\cos\mu \neq 0$ because otherwise a is orthogonal to the platform. Again we prove by contradiction that K_4 must vanish.

Assuming $K_4 \neq 0$ we get $b_2 = 0$ from $Q_{100}^{80} = 0$. Then the resultant of Q_{110}^{60} and Q_{020}^{80} with respect to B_3 can only vanish w.c. in the following cases:

1. $A_2 = a_2$: In this case $Q_{110}^{60} = 0$ implies $|\mathbf{b},\mathbf{B},\mathbf{Bb}|_3^5 = 0$:
   a. For the special case $B_3 = B_4 = B_5$ we get $\mu = \zeta$ with $\zeta := -\arcsin(n\cot\delta)$ from $Q_{200}^{42} = 0$. Then $Q_{200}^{33} = 0$ yields the contradiction.
   b. W.l.o.g. we can solve $|\mathbf{b},\mathbf{B},\mathbf{Bb}|_3^5 = 0$ for b_5. Due to $Q_{200}^{42} = 0$ we must distinguish the following two cases:
      i. $b_4 = b_3 B_4/B_3$: W.l.o.g. we can express a_5 from the only non-contradicting factor of $Q_{020}^{60} = 0$. Then $Q_{020}^{51} = 0$ implies $a_4 = A_4 + B_4(a_3 - A_3)/B_3$.

Now we can solve $K_1 = K_2 = 0$ for A_6 and b_6 w.l.o.g.. Moreover, substitution of these expressions into K_4 shows that it is fulfilled identically.

ii. $\mu = \zeta$, $b_4 \neq b_3 B_4/B_3$: Then $Q_{200}^{33} = 0$ already implies the contradiction.

2. $b_3 b_4 b_5 = 0$, $A_2 \neq a_2$: W.l.o.g. we set $b_3 = 0$. Now $Q_{200}^{51} = 0$ implies two cases:
   a. $B_4 = B_5$: Then $Q_{200}^{42} = 0$ yields $\mu = \zeta$. $Q_{200}^{33} = 0$ yields the contradiction.
   b. $\mu = \zeta$, $B_4 \neq B_5$: $Q_{020}^{60} = 0$ yields $A_3 = a_3 A_2/a_2$ and $Q_{200}^{42} = 0$ the contradiction.
3. $B_4 = B_5$, $b_3 b_4 b_5 (A_2 - a_2) \neq 0$: Due to $Q_{200}^{51} = 0$ we must distinguish two cases:
   a. $B_3 = B_5$: Now $Q_{200}^{42} = 0$ implies $\mu = \zeta$. $Q_{200}^{33} = 0$ yields the contradiction.
   b. $b_4 = b_5$, $B_3 \neq B_5$: Then $Q_{010}^{80} = 0$ cannot vanish w.c..
   c. $\mu = \zeta$, $(b_4 - b_5)(B_3 - B_5) \neq 0$: $Q_{200}^{42} = 0$ yields the contradiction.

Now we can set $K_4 = 0$ and compute $Q = A_2 e_4 K_3 F[57528]$. We distinguish again between the following two cases for proving that F cannot vanish w.c.:

1. $b_2 \neq 0$: Now we can solve $F_{110}^{50} = 0$ for a_5. From $F_{200}^{32} = 0$ we can express a_4. From $F_{020}^{50} = 0$ we get A_5. $F_{020}^{41} = 0$ yields an expression for A_4. W.l.o.g. we can solve $K_1 = K_2 = 0$ for A_6 and b_6. Then $b_2 K_3 - a_2 K_4 = 0$ holds. This is a contradiction as $K_4 = 0$ implies $K_3 = 0$.
2. $b_2 = 0$: Now F_{200}^{50} implies $|\mathbf{b}, \mathbf{B}, \mathbf{Bb}|_3^5 = 0$. Again we start with the special case:
   a. $B_3 = B_4 = B_5$: $F_{110}^{41} = 0$ already yields the contradiction.
   b. W.l.o.g. we can compute b_5 from $|\mathbf{b}, \mathbf{B}, \mathbf{Bb}|_3^5 = 0$. Now F_{200}^{32} can only vanish w.c. in the following 2 cases:
      i. $b_4 = b_3 B_4/B_3$: An accurate case study shows that we only end up with contradictions. For the detailed discussion we refer to [10]. Moreover it should be noted that this case implies solutions for the the special case $\alpha = \beta \in]0, \pi/2[$.
      ii. $\mu = \zeta$, $b_4 \neq b_3 B_4/B_3$: Then $F_{200}^{23} = 0$ implies the contradiction. □

3 Main Theorem for the Special Case

Theorem 3. *$\nexists$ non-architecturally singular planar SG platforms with 4 collinear anchor points which are $\mathsf{X}(\mathsf{a})$-singular if $\alpha \neq \beta$ and a not orthogonal to Φ or φ.*

Proof. In order to prove this theorem efficiently we need a good choice for the coordinate systems in Σ and Σ_0. Based on some geometric considerations such a coordinatization can be done as follows: W.l.o.g. we can assume that the four collinear points are on the platform, i.e. $\mathsf{m}_1, \ldots, \mathsf{m}_4$ are situated on the line g. Now we must distinguish again two cases, depending on the property if $\gamma \geq \alpha$ or $\gamma < \alpha$ holds with $\gamma := \angle(\mathsf{g}, \mathsf{a}) \in [0, \pi/2]$.

3.1 $\gamma \geq \alpha$

In this case we translate φ and Φ such that $\mathsf{M}_1 = \mathsf{m}_1$ holds. As $\gamma \geq \alpha$ there exist at least one position by rotating of φ about a such that $\mathsf{g} \in \Phi$ holds. This is the starting configuration of the following coordinatization: $\mathbf{M}_i = (A_i, B_i, 0)$ and $\mathbf{m}_i = (a_i, b_i \cos\delta, b_i \sin\delta)$ with $A_1 = B_1 = a_1 = b_1 = b_2 = b_3 = b_4 = 0$ and $\sin\delta \neq 0$.

Moreover we set $e_1 = e_4 \cos\mu$, $e_3 = e_4 \sin\mu$ and $e_2 = e_4 n$, where $n = \cos\mu = 0$, $n = \sin\mu = 0$ or $\cos\mu = n\cos\delta + \sin\mu\sin\delta = 0$ yield contradictions.

Part [A] $\sin\mu \neq 0$

Firstly, we show that we can assume $\mathsf{M}_5 \neq \mathsf{M}_6$ and that no 5 platform anchor points are collinear because these two cases yield a contradiction:

1. $b_5 = 0$: We give those 5 coefficients which imply $rk(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{Aa}, \mathbf{Ba})_2^5 \leq 3$. This yields a contradiction due to [5]. We distinguish 3 cases:
   a. $n = 0$: Four conditions are given by $Q_{201}^{13} = Q_{200}^{15} = Q_{021}^{22} = Q_{020}^{24} = 0$. For $B_6 \neq 0$ we get the fifth condition from $Q_{001}^{62} = 0$. For $B_6 = 0$ and $A_6 \neq 0$ we get it from $Q_{001}^{53} = 0$. For the case $\mathsf{M}_1 = \mathsf{M}_6$ it is given by $Q_{101}^{33} = 0$.
   b. $n = \nu := -\sin\mu\tan\delta$: Four conditions are given by $Q_{201}^{13} = Q_{200}^{24} = Q_{021}^{31} = Q_{020}^{42} = 0$. For $B_6 \neq 0$ we get the fifth condition from $Q_{001}^{71} = 0$. For $B_6 = 0$ and $A_6 \neq 0$ we get it from $Q_{001}^{53} = 0$. For the case $\mathsf{M}_1 = \mathsf{M}_6$ it is given by $Q_{002}^{51} = 0$.
   c. $\nu \neq n \neq 0$: Four conditions are given by $Q_{201}^{22} = Q_{200}^{33} = Q_{021}^{31} = Q_{020}^{42} = 0$. For $B_6 \neq 0$ we get the fifth condition from $Q_{001}^{71} = 0$. For $B_6 = 0$ and $A_6 \neq 0$ we get it from $Q_{001}^{62} = 0$. For the case $\mathsf{M}_1 = \mathsf{M}_6$ and $\cos\delta \neq 0$ it is given by $Q_{002}^{51} = 0$. If additionally $\cos\delta = 0$ hold we get the last condition from $Q_{101}^{42} = 0$.
2. $\mathsf{M}_5 = \mathsf{M}_6$: We give the 4 necessary and sufficient conditions indicating the degenerated cases of architecturally singular planar parallel manipulators (cf. [4]):
   a. $n = 0$: $Q_{021}^{22} = Q_{020}^{24} = Q_{201}^{13} = Q_{200}^{15} = 0$.
   b. $n = \nu$: $Q_{021}^{31} = Q_{020}^{42} = Q_{201}^{13} = Q_{200}^{24} = 0$.
   c. $\nu \neq n \neq 0$: $Q_{021}^{31} = Q_{020}^{42} = Q_{201}^{22} = Q_{200}^{33} = 0$.

Moreover, w.l.o.g. we can assume that if 3 points of $\mathsf{M}_1, \ldots, \mathsf{M}_4$ are collinear and pairwise distinct they are $\mathsf{M}_1, \mathsf{M}_2, \mathsf{M}_3$. We can also assume that if 2 points of $\mathsf{M}_1, \ldots, \mathsf{M}_4$ coincide, they are M_2 and M_3.

Now $Q_{111}^{40} = 0$ and $Q_{021}^{40} = 0$ imply $|\mathbf{a}, \mathbf{A}, \mathbf{B}|_2^4 = 0$. W.l.o.g. we can express a_2 from this condition. In the next step we prove by contradiction that W must vanish with

$$W := a_3(A_2B_4 - A_4B_2)(B_2 - B_3) + a_4(A_3B_2 - A_2B_3)(B_2 - B_4).$$

From $Q_{101}^{60} = 0$ we get $B_5 = B_6$. Now Q_{200}^{42} can only vanish w.c. under consideration of $Q_{011}^{60} = 0$ for $n = 0$ or $n = \nu$. In both cases $Q_{200}^{33} = 0$ yields the contradiction.

Part [B] $(B_2 - B_3)\sin\mu \neq 0$

Under this assumption we can express a_3 from $W = 0$. Then $Q_{102}^{22} = 0$ together with $Q_{021}^{31} = 0$ imply an expression for a_5. Now Q_{100}^{71} can only vanish w.c. for:

1. $n = 0$: Now $Q^{62}_{100} = 0$ implies $B_5 = B_6$ or $B_2B_3B_4 = 0$.
   a. $B_5 = B_6$: Assuming $B_2B_3 \neq 0$ we can express A_4 from $Q^{42}_{101} = 0$. From $Q^{33}_{101} = 0$ we get A_6 and $Q^{53}_{100} = 0$ yields the contradiction. For the special case $B_2B_3 = 0$ we can set $B_2 = 0$ w.l.o.g.. Then $Q^{42}_{101} = 0$ implies $B_3 = B_4$. From $Q^{33}_{101} = 0$ we get $b_5 = b_6$ and $Q^{53}_{010} = 0$ yields the contradiction.
   b. $B_2B_3B_4 = 0$, $B_5 \neq B_6$: In all 3 cases we get the contradiction from $Q^{51}_{101} = 0$.

2. $B_5 \neq B_6$, $n \neq 0$: Now $Q^{42}_{020} = 0$ and $Q^{42}_{110} = 0$ can only hold if the common factor $G[48]$ vanishes or for $H_1[6] = H_2[6] = 0$. As the latter case yield easy contradictions we set $G = 0$ and introduce the following notation:

$$R := A_2B_3B_4(B_4 - B_3)(B_2 - B_6) - A_3B_2B_4(B_4 - B_2)(B_3 - B_6) + A_4B_2B_3(B_3 - B_2)(B_4 - B_6).$$

   a. $R \neq 0$: Now we can compute A_6 from $G = 0$. Then Q^{62}_{100} can only vanish w.c. for $n = \mu$, but in this case $Q^{53}_{100} = 0$ yields the contradiction.
   b. $R = 0$, $B_2B_3(B_6 - B_4) \neq 0$: Under this assumption we can compute A_4 from $R = 0$. Now $G = 0$ can only vanish w.c. for $b_5 = b_6$. Then $Q^{42}_{101} = 0$ implies $n = \nu$ and $Q^{44}_{100} = 0$ yields the contradiction.
   c. $R = 0$, $B_2B_3 = 0$: W.l.o.g. we set $B_2 = 0$. Then $R = 0$ can only vanish w.c. for:
      i. $B_6 = 0$: Due to $Q^{53}_{100} = 0$ we must distinguish two cases: For $B_3 = B_4$ we get $n = \nu$ from $Q^{62}_{010} = 0$ and $Q^{53}_{010} = 0$ yields the contradiction. For the second case $n = \nu$, $B_3 \neq B_4$ we get the contradiction from $Q^{35}_{100} = 0$.
      ii. $B_3 = B_4$, $B_6 \neq 0$: Due to $Q^{33}_{110} = 0$ we must distinguish 3 cases: For the cases $b_5 = b_6$ and $B_4 = B_6$ we get $n = \nu$ from $Q^{51}_{011} = 0$ and the contradiction from $Q^{42}_{011} = 0$. For the third case $n = \nu$, $(B_4 - B_6)(b_5 - b_6) \neq 0$ we get the contradiction from $Q^{24}_{110} = 0$.
   d. $R = 0$, $B_4 = B_6$, $B_2B_3 \neq 0$: Now R can only vanish w.c. for:
      i. $B_6 = 0$: $Q^{53}_{100} = 0$ implies $n = \nu$ and $Q^{35}_{100} = 0$ yields the contradiction.
      ii. $B_2 = B_6 \neq 0$: $Q^{42}_{101} = 0$ yields $n = \nu$ and $Q^{24}_{101} = 0$ the contradiction.

3. $B_4 = 0$, $n(B_5 - B_6) \neq 0$: We get the contradiction from $Q^{51}_{110} = 0$.
4. $B_2B_3 = 0$, $nB_4(B_5 - B_6) \neq 0$: W.l.o.g. we set $B_2 = 0$. Now $Q^{51}_{110} = 0$ implies $B_3 = B_4$ and then $Q^{71}_{010} = 0$ yields the contradiction.

Part [C] $B_2 = B_3$, $\sin\mu \neq 0$

Now W can only vanish w.c. in the following 2 cases:

1. $a_4 = 0$: Now $Q^{22}_{102} = 0$ and $Q^{31}_{021} = 0$ imply $|\mathbf{a}, \mathbf{b}, \mathbf{A}, \mathbf{B}|^6_3 = 0$. W.l.o.g. we can solve this condition for a_5. Due to $Q^{71}_{100} = 0$ we must distinguish four cases:
   a. $n = 0$: Then $Q^{62}_{100} = 0$ can only vanish w.c. in the following 2 cases: For $B_5 = B_6$ we get $B_4 = 0$ from $Q^{42}_{101} = 0$ and $Q^{53}_{010} = 0$ yields the contradiction. For the 2^{nd} case $B_i = 0$, $B_5 \neq B_6$ for $i = 3,4$ we get the contradiction from $Q^{51}_{101} = 0$.
   b. $B_5 = B_6$, $n \neq 0$: Now Q^{42}_{020} and Q^{42}_{110} can only vanish w.c. for:
      i. $B_3 = B_4$: Due to $Q^{51}_{011} = 0$ we must distinguish 2 cases: For $A_4 = B_4(A_3 - a_3)/B_6$ we get $n = \nu$ from $Q^{42}_{101} = 0$ and the contradiction from $Q^{24}_{101} = 0$. In the second case $n = \mu$ we get the contradiction from $Q^{42}_{011} = 0$.

ii. $b_6(A_4B_5 - B_4A_5) + b_5(A_6B_4 - A_4B_6) = 0, B_3 \neq B_4$: Assuming $B_4 \neq 0$ we can express A_6 from this condition. Then $Q_{100}^{62} = 0$ implies $n = \nu$ and $Q_{100}^{53} = 0$ yields the contradiction. For the special case $B_4 = 0$ the above condition can only vanish w.c. for $B_6(b_5 - b_6) = 0$. In both cases $Q_{011}^{51} = 0$ implies $n = \nu$ and $Q_{011}^{42} = 0$ yields the contradiction.

c. $B_3 = 0, n(B_5 - B_6) \neq 0$: We get immediately the contradiction from $Q_{110}^{51} = 0$.

d. $B_4 = 0, nB_3(B_5 - B_6) \neq 0$: In this case $Q_{110}^{42} = 0$ implies $n = \nu$ and finally $Q_{110}^{33} = 0$ yields the contradiction.

2. $B_3 = 0, a_4 \neq 0$: Now $Q_{102}^{22} = 0$ and $Q_{021}^{31} = 0$ imply again $|\mathbf{a}, \mathbf{b}, \mathbf{A}, \mathbf{B}|_3^6 = 0$. W.l.o.g. we can solve this condition for a_5. Then $Q_{110}^{51} = 0$ can only vanish w.c. for:

a. $n = 0$: $Q_{101}^{51} = 0$ implies $B_5 = B_6$ and $Q_{101}^{42} = 0$ yields the contradiction.

b. $B_5 = B_6, n \neq 0$: Now $Q_{110}^{42} = 0$ implies an expression for A_6. From $Q_{100}^{62} = 0$ we get $n = \nu$ and $Q_{100}^{53} = 0$ yields the contradiction.

Now only the discussion of the special case $\sin\mu = 0$ ($\Leftrightarrow \Phi \parallel \mathsf{a}$) is missing. This case study can exactly be done as the one for $\sin\mu \neq 0$. The only differences are that we always get $\cos\delta = 0$ instead of $n = \nu$ and that $n = 0$ yields a contradiction.

3.2 $\gamma < \alpha$

In this case we translate φ and Φ such that $\mathsf{M}_1 = \mathsf{m}_1$ holds. As $\gamma < \alpha$ there exist two positions by rotating of φ about a such that $[\mathsf{M}_1, \mathsf{M}_2] \in \varphi$ holds. This reasons the following coordinatization: $\mathbf{M}_i = (A_i, B_i, 0)$ and $\mathbf{m}_i = (a_i, b_i\cos\delta, b_i\sin\delta)$ with $A_1 = B_1 = B_2 = a_1 = b_1 = 0$, $a_i = b_i a_2/b_2$ for $i = 3,4$ and $b_2 \sin\delta \neq 0$.

We set $e_1 = e_4\cos\mu$, $e_3 = e_4\sin\mu$ and $e_2 = e_4 n$. As $\beta \leq \gamma < \alpha$ holds, $\sin\mu = 0$ yields a contradiction as well as $n = \cos\mu = 0$ or $\cos\mu = n\cos\delta + \sin\mu\sin\delta = 0$.

Moreover, due to the result of Sec. 3.1 we can stop the case study if 4 base anchor points are collinear or if $b_5 = b_6 = b_i = b_j$ holds with $i, j \in \{1, \ldots, 4\}$ and $i \neq j$.

In spite of all these assumptions the discussion of this case takes further 3 pages. Due to the limitation of pages we refer to the corresponding technical report [11]. □

4 Conclusion

Theorems 2 and 3 can be summed up to the following main theorem on Schönflies-singular planar SG platforms:

Main Theorem. $\mathsf{X}(\mathsf{a})$*-singular planar Stewart Gough platforms with* $\alpha \neq \beta$ *and where* a *is not orthogonal to* Φ *or* φ *are necessarily architecturally singular.*

Consequences of this main theorem are as follows:

- Mick and Röschel proved in Theorem 4.1 of [9] that a planar SG platform is architecturally singular if and only if it is singular with respect to a special 5-parametric set of displacements. Due to the given main theorem for $\mathsf{X}(\mathsf{a})$-singular manipulators we can improve this statement even to 4-parametric sets of displacements namely the Schönflies motion groups for which Theorem 2 and 3 hold.
  The question remains open, if this statement can further be improved to an even 3-dimensional Lie subgroup of $\mathsf{SE}(3)$, which are $\mathsf{SO}(3)$ and $\mathsf{H}(\mathsf{d}) \rtimes \mathbb{R}^2$. The latter is composed of translations on a plane and a helical motion (with pitch p) along the normal direction d of the plane. $\mathsf{H}(\mathsf{d}) \rtimes \mathbb{R}^2$ also includes the Cartesian motion group $\mathsf{T}(3)$ ($p = \infty$) and the planar motion group $\mathsf{SE}(2)$ ($p = 0$) as special cases. Due to the presented main theorem and the results given in [1, 10] we can restrict $\mathsf{H}(\mathsf{d}) \rtimes \mathbb{R}^2$ to $p \in [0, \infty[$ with $\angle(\boldsymbol{\Phi}, \mathsf{d}) \neq \angle(\boldsymbol{\varphi}, \mathsf{d})$ and d not orthogonal to $\boldsymbol{\Phi}$ or $\boldsymbol{\varphi}$.
- The manipulators given in Theorem 1 are the only non-architecturally singular planar SG platforms with $\alpha \neq \beta$ which are Schönflies-singular.
  Moreover it should be noted, that the missing special cases (i.e. $\alpha = \beta$) of Schönflies-singular planar Stewart Gough platforms are given in [10]. Therefore paper [10] also finishes the discussion of Schönflies-singular planar parallel manipulators which was started by Wohlhart [12] by giving an example for a $\mathsf{X}(\mathsf{a})$-singular planar SG platform of case (2a). The presented manipulator (*polygon platform*) even possesses a Schönflies self-motion because it is a special case of a parallel manipulator with Schönflies Borel-Bricard motions (cf. Husty and Zsombor-Murray [13]) listed by Borel [14]. Moreover Husty and Karger proved in [15] that Borel's list is complete.
  Therefore the only open problem in this context is the determination of all non-planar Schönflies-singular Stewart Gough platforms.

References

1. Nawratil, G.: A remarkable set of Schönflies-singular planar Stewart Gough platforms, Technical Report No. 198, Geometry Preprint Series, Vienna University of Technology (2009).
2. Karger, A.: Architecture singular planar parallel manipulators, Mechanism and Machine Theory **38** (11) 1149–1164 (2003).
3. Karger, A.: New Self-Motions of Parallel Manipulators, Advances in Robot Kinematics: Analysis and Design (J. Lenarcic, P. Wenger eds.), 275–282, Springer (2008).
4. Nawratil, G.: On the degenerated cases of architecturally singular planar parallel manipulators, Journal of Geometry and Graphics **12** (2) 141-149 (2008).
5. Röschel, O., Mick, S.: Characterisation of architecturally shaky platforms, Advances in Robot Kinematics: Analysis and Control (J. Lenarcic, M.L. Husty eds.), 465–474, Kluwer (1998).
6. Wohlhart, K.: From higher degrees of shakiness to mobility, Mechanism and Machine Theory **45** (3) 467–476 (2010).
7. Karger, A.: Architecturally singular non-planar parallel manipulators, Mechanism and Machine Theory **43** (3) 335–346 (2008).
8. Nawratil, G.: A new approach to the classification of architecturally singular parallel manipulators, Computational Kinematics (A. Kecskemethy, A. Müller eds.), 349–358, Springer (2008).
9. Mick, S., Röschel, O.: Geometry & architecturally shaky platforms, Advances in Robot Kinematics: Analysis and Control (J. Lenarcic, M.L. Husty eds.), 455-464, Kluwer (1998).

10. Nawratil, G.: Special cases of Schönflies-singular planar Stewart Gough platforms, Technical Report No. 202, Geometry Preprint Series, Vienna University of Technology (2009).
11. Nawratil, G.: Main Theorem on Schönflies-singular planar Stewart Gough platforms, Technical Report No. 199, Geometry Preprint Series, Vienna University of Technology (2009).
12. Wohlhart, K.: Architectural Shakiness or Architectural Mobility of Platforms, Advances in Robot Kinematics (J. Lenarcic, M.M. Stanisic eds.), 365-374, Kluwer (2000).
13. Husty, M.L., Zsombor-Murray, P.: A Special Type of Singular Stewart Gough platform, Advances in Robot Kinematics and Computational Geometry (J. Lenarcic, B. Ravani eds.), 439–449, Kluwer (1994).
14. Borel, E.: Mémoire sur les déplacements à trajectoires sphériques, Mém. présentés par divers savants, Paris **2** (33) 1–128 (1908).
15. Husty, M.L., Karger, A.: Self motions of Stewart-Gough platforms: an overview, In Proc. of the workshop on fundamental issues and future research directions for parallel mechanisms and manipulators (S.M. Gosselin, I. Ebert-Uphoff eds.), 131–141 (2002).

A Novel Actuation Module for Wearable Robots

M. Bergamasco, F. Salsedo, S. Marcheschi and N. Lucchesi

PERCRO Laboratory of Scuola Superiore Sant'Anna, Pisa, Italy;
e-mail: {massimo.bergamasco, fabio.salsedo, simone.marcheschi, nicola.lucchesi}@sssup.it

Abstract. Wearable robots, like prostheses, active orthoses and exoskeletons for human performance augmentations need a new breed of actuators, capable to exhibit a large number of desirable features: high power/torque density, high efficiency, zero backlash, low reflected mechanical impedance and high bandwidth. To fully meet these requirements, new basic actuation principles have to be investigated. Nevertheless, there is still scope now to innovate the field by combining mature components into new actuation schemes. This paper reports the development of an innovative actuator, expressly conceived as an actuation module for a fully powered whole body exoskeleton. The main design objective has been the enhancement of the torque density and the mechanical efficiency with respect to existing solutions. The final performances of the actuation module are: continuous torque of 500Nm, mechanical efficiency of 85%, zero backlash and total weight of just 6 Kg comprising the structural case.

Key words: Advanced actuators for robotics, high torque density and mechanical efficiency, advanced robotics, exoskeletons for human performance augmentation.

1 Introduction

Wearable robots, like exoskeletons for human performance augmentation or for the physical assistance/rehabilitation of the elderly/disabled, are emerging examples of advanced robots that operate in direct and tight interaction with humans. However, the less than ideal features/performances exhibited by commonly used actuation solutions make clear the need for a long term investigation and development of new basic actuation principles. On the other hand, innovative actuator schemes obtained by combining mature mechanical solutions have been recently proposed by the research community for improving features like force control accuracy, bandwidth of the force control, shock tolerance and the reduction of the energy transferred to humans in case of collision. All these improvements have been achieved through the purposely introduction of passive elastic elements in the mechanical transmission.

The first work about this solution has been carried at MIT [1]: the proposed actuator, dubbed Serial Elastic Actuator, demonstrates better stability and accuracy in force control, at the expense of a lower bandwidth. In an attempt to overcome this

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_13,

Fig. 1 Fully actuated body extender developed at PERCRO.

disadvantage, in [2] a Distributed Macro-Mini (DM2) approach has been proposed. In [3] an actuation solution able to store and release mechanical energy in accordance with the gait phases has been proposed. Furthermore, to keep under safety threshold the energy transferred during possible collisions with humans, Wolf and Hirzinger [4] have introduced stiffness-variable elastic elements in the mechanical transmission.

However, to the authors' knowledge, the problem of increasing the torque density, having at the meantime a relative intrinsic high back drivability and hence mechanical efficiency, has been poorly addressed. Indeed, higher actuator torque densities allow to achieve lighter robotic limbs and, hence, higher bandwidth as well as a more slender design of the robotic structure, with clear benefit for the aesthetic and user's acceptability of the device. On the other hand, a higher mechanical efficiency allows better estimation of the force exerted on the external environment also in case force sensors are not used, as well as to keep low the non linearity of the force transmission that can prevent to achieve high tracking performances due to the occurrence of instabilities. Due to the poor back drivability and force accuracy and costly maintenance of pneumatic and hydraulic solutions, the preferred technology for the actuation of the robotic joints is the electromagnetic motor used in combination with high reduction ratio speed reducers to increase the torque density. Nevertheless, the performances of this type of actuators are seriously deteriorated due to the friction, backlash, elasticity and transmission ripple introduced by the speed reducers. In the recent years, the Harmonic Drive has been selected as the preferred solution for speed reduction, by virtue of its compactness and lightness. A good example of this kind of implementation is the integrated joint developed at DLR [5] that uses last generation Harmonic Drive speed reducers with reduction ratio of the order of 100 in combination with high performance DC torque motors.

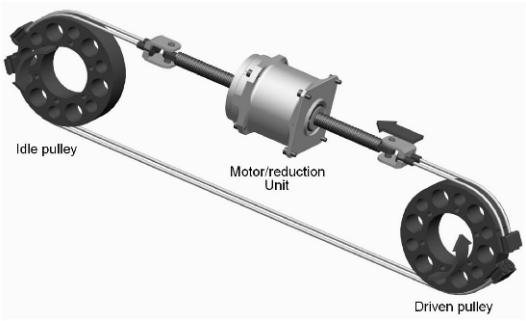


Fig. 2 Actuation scheme proposed in [6] (simplified).

In an attempt to go beyond the present actuation solutions, in [6] a lead screw is used as main speed reducer in combination with two metallic tendons and a driven and an idle pulleys to convert the linear motion into a rotation on the output axis (see the simplified scheme of Fig. 2). The scheme is potentially interesting if a ball lead screw is used, by virtue of its beneficial features in terms of high reduction ratio, lightness, zero backlash, smoothness and high mechanical efficiency that are not deteriorated by the use of the metallic tendons. Yet, it has the big drawback of producing large encumbrances in the case of large angular stroke and torque requirements. Starting from this embodiment, a novel actuation module has been developed at the Perceptual Robotics Laboratory, expressly conceived for being integrated in a fully actuated whole body exoskeleton intended for material handling (Fig. 1). Details about the design guidelines and the main features of the novel actuation module are reported in the next sections.

2 Design of the Actuation Scheme

The new actuator has been expressly conceived for driving the joints of a whole body exoskeleton. The main requirements for the mechanical design of the robotic limbs of the device are the following:

- kinematics isomorphic to that of the human limbs and encumbrance of the mechanics adherent to that of the human limbs;
- actuation achieved through electric motors;
- mechanics of the robotic limbs conceived as a series of self contained 1 DoF actuation modules connected together to form a serial chain.

Moreover, the new actuation scheme has been conceived for the implementation of the transversal joint (for example the elbow flexion-extension joint). For these joints, the ideal encumbrance distribution should have a major dimension in one of the directions orthogonal to the actuated axis, while the other two, and in particular

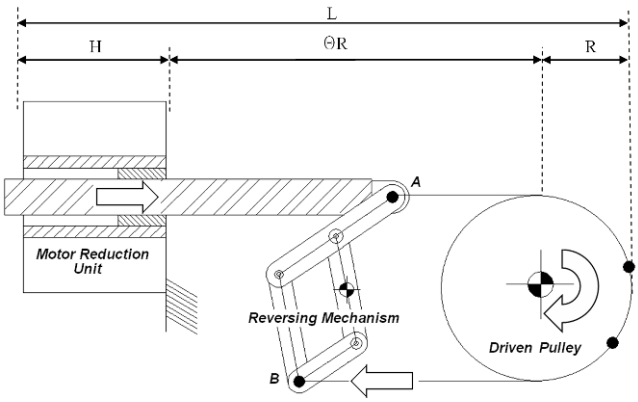


Fig. 3 First embodiment of the reversing mechanism.

the one aligned with the output axis, should be as small as possible, to keep the mechanics more adherent to the human limb.

The solution of using an electric motor in combination with a geared speed reducer mounted coaxially with the output axis, is poorly suitable because produces relatively large axial and radial encumbrance.

These considerations together with the general requirement of reducing the weight of the actuation module led us to select the ball lead screw as the main speed reducer. Furthermore it allows to position the axis of the motor orthogonally to the output axis and, hence, to have an encumbrance compatible to the requirement of the transversal joint. To have a rotational output, a mechanism is required to convert the linear motion of the screw. This can be simply achieved by directly connecting the screw to the output link, with an heavy reduction of the maximum generable torque near the angular limits, or by using a metallic tendon connected to the endpoint of the screw and wrapped to a driven pulley. Considering this case, to generate bidirectional torques a second metallic tendon can be driven as proposed in [6], but this solution produces very large encumbrance in the orthogonal direction. Indeed, indicating with Θ, R and H, respectively the angular stroke, the radius of the driven pulley and the axial dimension of the motor, the minimum theoretical encumbrance L is $L = 2R(\Theta + 1) + H$.

To drastically decrease this drawback, we propose a new actuation scheme (Fig. 3), that envisages the use of a mechanism, located between the motor reduction unit and the driven pulley, to drive the second tendon, in such a way that the total length of the tendon circuit (from A to B) remains constant for the different output angular positions of the driven pulley (also any given tendon preload remains constant). According to this scheme the theoretical minimum encumbrance is $L = R(\Theta + 1) + H$ that is quite half the encumbrance of the previous scheme. It is worth noting that, in general, the second line along which point B moves, can be inclined with respect the screw axis. This is useful for reducing the transversal encumbrance of the actuation module and increasing the angular stroke beyond $180°$.

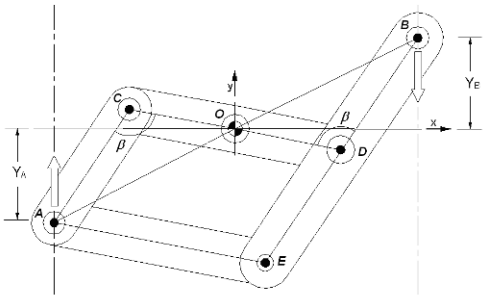


Fig. 4 Geometry of the pantograph mechanism.

3 Embodiments of the Reversing Mechanism

Three different embodiments have been identified for the implementation of the reversing mechanism envisaged in the new actuator scheme [7]. In this paper, only the preferred embodiment is described, using a pantograph mechanism for reversing the motion of the end point of the screw. Referring to Fig. 4, only the axis of rotational pair O is fixed, while the other axes are moving. To ensure the required kinematics property for the reversing mechanism, the following geometric conditions must be satisfied: $OC = OD$, $CA = DB$, $AE = CD$ and $ED = AC$.

The last two conditions ensure that the quadrilateral $ACDE$ is a parallelogram and hence BE is parallel to AC and the angles $\widehat{ACO}$ and $\widehat{ODB}$ are equal. Together with the first two conditions this ensures that the triangles OCA and ODB are equal and hence also Ya and Yb are equal. In conclusion if Ya varies of the quantity ΔY, Yb varies of the same quantity but in the opposite direction. More in general the points A, O and B lie on line and it is always $OA = OB$. Hence if point A (or B) follows a planar trajectory, B (A) follows the same trajectory but rotated of 180° around the fixed point O. In particular if point A moves along a straight line, point B moves along another straight line, resulting the rotation by 180° of the first line around the fixed point O.

By applying the virtual work principle it can be easily demonstrated that, in the case of negligible friction in the rotational pairs, if a force vector is applied in point A (or B), point B (A) generates to the external world a force vector that is equal to the force vector applied in A (B) and rotated of 180° around the fixed point O. In particular if A moves along a straight line, then also B moves along a straight line and if the force applied in B has null component in the direction normal to its straight line, then point A transfers to the external work a force with a null component in the direction normal to its straight line. Applied to the case of our actuation scheme, this property ensures that the screw is not loaded with force having component normal to its axis. Indeed, if the straight line followed by point B, when point A moves along the axis of the screw, is tangent to the primitive of the driven pulley, then, due to bending flexibility of the metallic tendon, the force applied by the tendon in B will

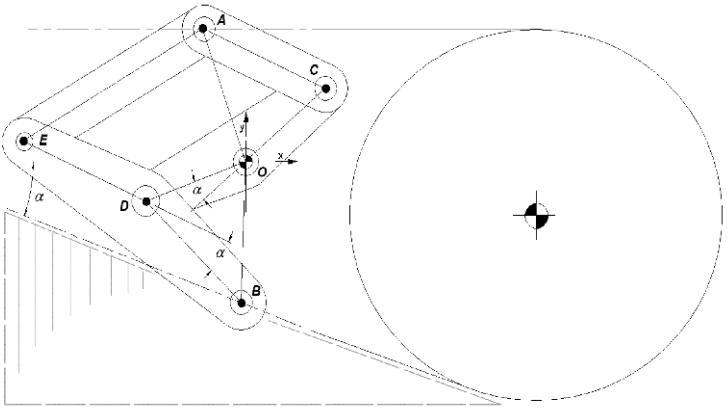


Fig. 5 Variant of the pantograph mechanism.

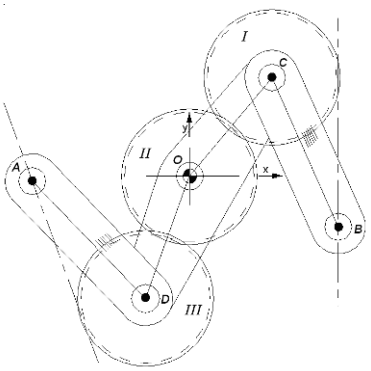


Fig. 6 Second embodiment of the reversing mechanism.

have a negligible component normal to the straight line and hence this will produce a negligible force component normal to the axis of the screw.

An interesting variant of this mechanism exhibits the same properties described above but rotating trajectories and forces by an angle different from 180° (Fig. 5). This can be easily achieved inserting an angle (α) between the 2 segments *OC* and *OD* of the lever rotating around *O*, while always satisfying the same above geometric conditions. As anticipated this is convenient for reducing the transversal encumbrance and increasing the angular stroke.

The second embodiment uses gears instead of the parallelogram to mirror the angles $\widehat{ACO}$ and $\widehat{ODA}$ (Fig. 6). Gear I and gear III have equal radius and are integral, respectively to lever *CB* and lever *DA*, while gear II is idle around fixed axis *O*. Also this embodiment can produce rotations of the trajectories and forces different from 180°. It is worth to note that any backlash existing in the engagement of the gears can be canceled by a suitable preload on the two tendons.

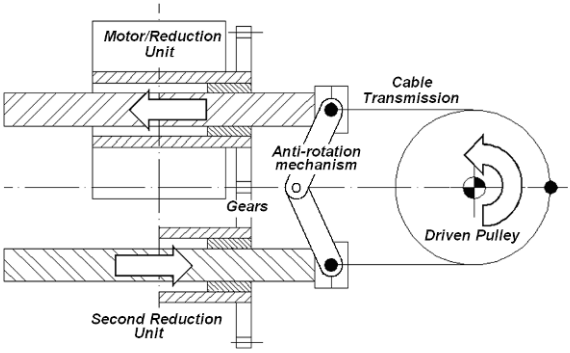


Fig. 7 Third embodiment of the reversing mechanism.

The third embodiment is somewhat different from the previous ones because the reversion of the motion is generated directly from the rotation of the motor, using a second lead screw whose nut is brought in rotation by a couple of gears with equal radius (Fig. 7). Also in this case any backlash existing in the gears can be cancelled by a suitable preload in the tendons. To prevent the rotation of the screw around their axes, the end points of the screws have been connected together with a two link mechanism, having three rotational pairs of which two in correspondence of the endpoints of the screws and one for the relative articulation of the links.

Even if in principle this embodiment could have a variant that allows the introduction of an angle between the two axes of the screws, in practice the implementation could be complex due to the need to use bevel gears and to possible interferences of the second screw with the actuator's body.

4 Features of the Actuation Module

The actuation module has been designed for delivering a maximum output continuous torque of about 500 Nm at a maximum output speed of 60°/s for an angular stroke of 110°. The mechanics of the motor/reduction unit has been expressly designed and optimized for the application.

The core components of the unit, a frameless brushed DC torque motor with rare earth permanent magnet (by AXYIS Technology Inc. USA) and a 16 mm diameter/4 mm lead precision ball screw (by THK CO. LTD, Japan) have been selected among the best commercially available. The torque motor is able to deliver up to 4.5–6 Nm continuous torque depending on the thermal resistance from the winding to the environment and more that 7 Nm peak torque at a maximum speed of 1050 rpm, weighting only 1.4 Kg (stator + rotor). The ball screw has been specified to withstand thrust forces up to 12000 N and weights only 300 g. A purposely de-

Fig. 8 CAD view of the actuation module (without case).

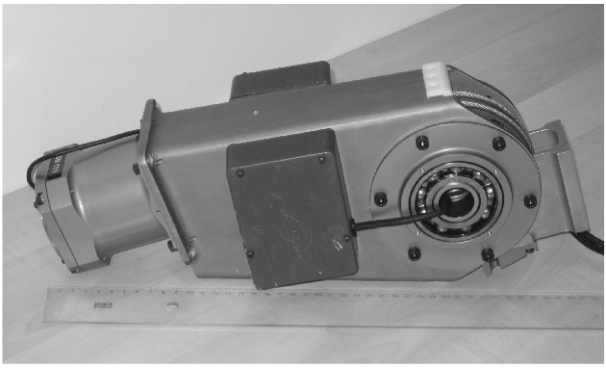


Fig. 9 The integrated actuation module.

veloped incremental encoder with 1000 pulses/revolution has been integrated in the mechanics.

The total mass of the motor reduction units is 2.4 Kg and is able to deliver more than 8000 N of continuous thrust force. The levers of the pantograph have been realized in high strength stainless steel (AISI 630 precipitation hardening stainless steel), while for the rotational pairs metallic plain bearings have been used. The 4 mm metallic tendons with clamped terminals have been supplied by Carl Stahl GmbH, Germany, according to the required length. The driven pulley, realized in hard anodized 7075-T6 aluminum with a set of slots to reduce the mass, has a primitive radius of 65 mm, setting the total reduction ratio at 102. It rotates on two ball bearings and is directly flanged to the actuator's output link. All the mechanics, apart the motor body, is contained in case made in casting aluminum, that provides structural strength for both the internal and external forces acting on the actuation module. The resulting CAD assembly model is depicted in Fig. 8. The total weight of the module is about 6 Kg and it is contained in an encumbrance envelope of about $370 \times 140 \times 70$ mm. A picture of the integrated module with the boxes containing the electronics is reported in Fig. 9. The theoretical mechanical efficiency is more than 85% and this value is confirmed by preliminary experimental tests.

5 Conclusions

A new actuation scheme using mature components has been proposed and a self contained actuation module has been developed according to it. The module has a very high torque density of the order of 80 Nm/Kg due to a combination of a state of art torque DC torque motor with a very lightweight speed reducer consisting of a ball screw, a reversing mechanism, a driven pulley and two metallic tendons. The mechanical efficiency is more than 85%, even if the total reduction ratio is more than 100. The backlash is zero and the functioning is smooth and with low noise. A patent application has been submitted to the Italian patent office [7].

References

1. Pratt G.A., Williamson M.M.: Series elastic actuators. In: *IEEE International Conference on Intelligent Robots and Systems*, 5–9 August, vol. 1, pp. 399–406, (1995).
2. Zinn M., Roth B., Khatib O., Salisbury J.K.: A new actuation approach for human friendly robot design. *The International Journal of Robotics Research*, **23**(4–5), 379–398 (2004).
3. Hollander K.W., Sugar T.G., Herring D.E.: Adjustable robotic tendon using a 'Jack Spring'. In: *IEEE International Conference on Rehabilitation Robotics, ICORR 2005*, 28 June–1 July, pp. 113–118 (2005).
4. Wolf S., Hirzinger G.: A new variable stiffness design: Matching requirements of the next robot generation. In: *IEEE International Conference on Robotics and Automation, ICRA 2008*, 19–23 May, pp. 1741–1746 (2008).
5. Hirzinger G., Sporer N., Schedl M., Butterfass J., Grebenstein M.: Torque-controlled lightweight arms and articulated hands: Do we reach technological limits now? *The International Journal of Robotics Research*, **23**(4–5), pp. 331–340 (2004).
6. Garrec P., Martins J.-P., Gravez F., Measson Y., Perrot Y.: A new force-feedback, morphologically inspired portable exoskeleton. In: *IEEE International Symposium on Robot and Human Interactive Communication, ROMAN 2006*, 6–8 September, pp. 674–679 (2006).
7. Bergamasco M., Salsedo F., Lucchesi N.: Attuatore compatto e leggero a escursione angolare limitata e elevata coppia. Italian Patent Application n. B100498, April 30 (2008).

Parallel Robot with Antagonistic Dielectric Elastomer Actuation for Human-Machine Interaction

R. Vertechy[1], G. Berselli[2], M. Bergamasco[1] and V. Parenti Castelli[2]

[1]*Gustavo Stefanini Research Center for Advanced Robotics, Scuola Superiore Sant'Anna, Pisa, Italy; e-mail: {r.vertechy, m.bergamasco}@sssup.it*
[2]*DIEM, Department of Mechanical Engineering, University of Bologna, Italy; e-mail: {giovanni.berselli, vincenzo.parenti}@unibo.it*

Abstract. The potentialities of a manipulator concept based on isotropic translational parallel mechanisms and agonistic-antagonistic dielectric elastomer actuation are investigated in the context of human-machine interfaces. Static analysis reveals that this manipulator concept is well suited to the development of novel generation of cost-effective interactive robots with low effective inertia and human-like performance and behaviour.

Key words: Interactive robot, dielectric elastomer, isotropic parallel manipulator.

1 Introduction

Several medical, training, social, working and entertainment activities or practices require robots capable of interacting with users and with unknown environments in a safe and human-like fashion. Potential benefits related to the availability of such interactive robots include: (1) improvement, speed-up and cost-reduction of rehabilitation and training protocols; (2) augmented interaction and communication capabilities; (3) improvement of work quality (which are all very important for the impaired). Desirable features of interactive robots comprise: (1) human matched performance (for instance workspace and force ranges); (2) stability robustness; (3) safety; (4) isotropy; (5) transparency (i.e. unitary ratio between transmitted and desired impedance). All these specifications call for devices with small moving masses, effective inertia and back-drive friction, and with zero-backlash and stiff mechanical transmissions. Several human-machine interaction devices (mainly haptic interfaces) have been proposed in the literature which differ in the number (1 to 6) and type (translational or rotational) of Degrees Of Freedom (DOF), mechanism architecture (serial or parallel), actuation (active, passive or hybrid) and control (impedance or admittance). Among them, isotropic devices based on parallel mechanisms usually provide: (1) fairly regular workspace and output forces; (2) higher and rather uniform rigidity; (3) lower and quite uniform effective inertia; (4) simple gravitational counterbalancing; and (5) better actuator exploitation. However, they

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_14, © Springer Science+Business Media B.V. 2010

often suffer from reduced workspace-to-encumbrance ratios. A rather comprehensive overview of parallel human-machine interface prototypes is provided in [1].

Despite the kinematic architecture, the major constraint to the development of safe interactive robots with human matched performance, sufficient transparency and stability robustness consists in the traditional actuation technology, which relies on either gear-head or direct-drive electric motors. Such actuators indeed feature [2] low power-to-mass (smaller than 40 W/kg) and torque/force-to-mass (smaller than 10 Nm/kg and 50 N/kg) ratios, which yield large effective inertia (larger for gear-head than for direct-drive motors) and masses (larger for direct-drive than for gear-head motors). Additionally, gear-head motors suffer from the series compliance, friction and backlash which are introduced by the geared transmission, while direct-drive motors suffer from force/torque ripples which are significant at the lower speed ranges that are typical in human-machine interaction applications. Electric motors are also rather expensive. To overcome this technological bottleneck, in recent years, new actuation approaches such as series-elastic actuation (based on the in-series connection of a motor with a linear spring [3]), distributed macro-mini actuation (based on the in-parallel connection of a high-torque low-frequency series-elastic motor and a low-torque high-frequency direct drive motor [4]), variable impedance actuation (based on the agonist-antagonist placement of two series-elastic actuators with non-linear spring stiffnesses [5]) and hybrid actuation (based on the in-parallel connection of a motor and a series-elastic brake [6]) have been proposed. Despite most of these attempts were successful, the developed solutions involve the use of electric motors, performant transmissions and additional mechanical components which make them rather complex, bulky and expensive.

Adequate but simpler, more compact and cheaper solutions may instead be achieved by exploiting Dielectric Elastomer (DE) technology. Dielectric elastomers are one of the best available multifunctional material for the development of solid-state, capacitive, easy-to-shape/-accommodate actuation systems featuring large force-to-weight (larger than 300 N/kg per unit of actuator length) and power-to-weight (larger than 400 W/kg) ratios, low costs (roughly 200 €/kg), large shock-insensitivity and silent operation [7]. Dielectric elastomers (like the pressure sensitive tape VHB-4905 by 3M) are elastic capacitors that can undergo finite deformations when subjected to large electric fields (about 100 MV/m). For actuation usage, DEs can be shaped in thin pre-tensioned films (about 50 μm, pre-tensioning is necessary to prevent wrinkling since DEs have negligible flexural rigidity), coated with compliant electrodes and connected to the links of parallel mechanisms. The electrical activation, obtained by placing the compliant electrodes to high voltage differences (1–10 kV), makes DEs expand in area. This yields forces on and motion of the links the DE is connected to, which can be used to produce useful mechanical work [8, 9]. Stacks of several DE films are required to obtain significant forces. DE actuated 1-DOF devices up to 200 N output force and multi-layered DE membranes with up to 400 films have been manufactured [10, 11]. Efficient and cheap driving electronics and effective closed-loop position regulators have also been demonstrated in practice [12].

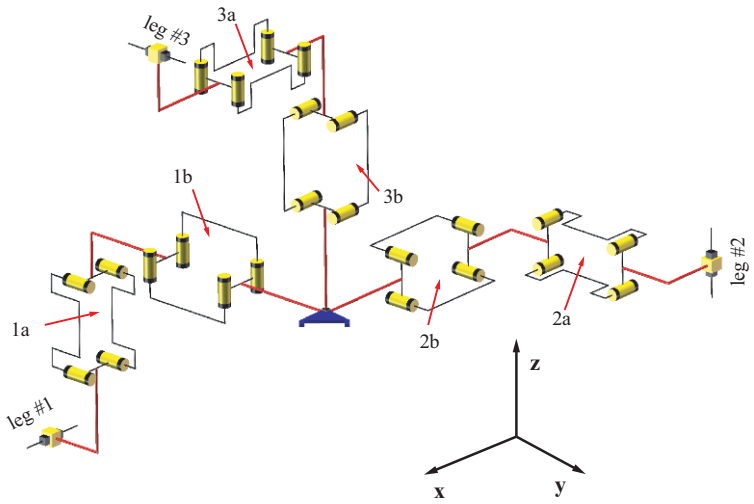


Fig. 1 Kinematic architecture of the proposed manipulator concept.

In this paper, an isotropic 3-DOF Translational Parallel Mechanism (TPM) actuated by Agonistic Antagonistic pairs of DE actuators with Parallelogram shape (AADEP) is presented as novel manipulator concept for safe interactive robots with human-like performance.

2 The Manipulator Concept

The kinematic architecture of the proposed manipulator concept is schematized in Fig. 1. A possible embodiment of the manipulator interacting with a human is drawn in Fig. 2.

It consists of a 3-PPaPa-type over-constrained TPM (P and Pa standing for prismatic pair and planar parallelogram loop respectively) with P$\perp$Pa$\perp^{\perp}$ Pa limb topology (i.e. the axis of the P pair and the normals to the Pa motion planes are mutually orthogonal) which is similar to that presented in [13, p. 497, fig. 5.9a]. They differ however since: (1) the TPM of Fig. 1 features grounded P pairs; (2) the common normal to the axes of the revolute pairs mounted on the common link of Pa and P is parallel (instead of orthogonal) to the direction of motion of P; (3) the two common normals to the axes of the revolute pairs of the two Pa loops which share the same intermediate link are parallel (instead of orthogonal). Such an arrangement provides a more compact device which is suited to be placed in the upper corner of a room with the three long rails of the P pairs fixed to the intersecting walls and ceiling (note that the first Pa loop of each leg is parallel and adjacent to the wall/ceiling it is connected to via the P pair). All Pa loops are identical, which yields a highly modular system that is simple to manufacture, assemble and maintain. The presence of

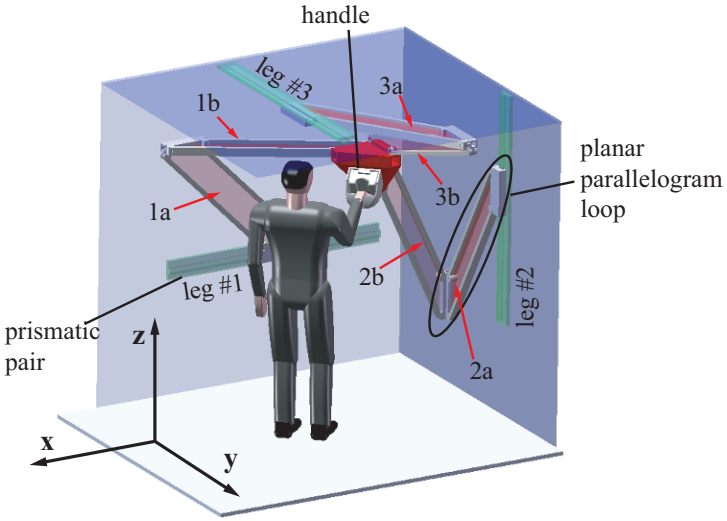


Fig. 2 CAD drawing of the proposed manipulator concept interacting with a human.

Pa loops also grants good stiffness-to-weight ratios and enables simple gravitational counterbalancing via linear springs [14]. The considered manipulator fits within a cube with 1.05 m long edges (roughly equalling the identical P rail lengths) and features a cubic workspace with 0.42 m long edges.

Each Pa loop accommodates within its interior a pre-tensioned multi-layered DE membrane with Parallelogram shape (DEP) whose perimeter edges are attached to the respective Pa links. This provides a rather simple means of actuation which does not require the interposition of any additional transmission (which may degrade mechanism stiffness and introduce sources of friction and backlash) and does not increase overall system encumbrance. Owing to the chosen limb arrangement, the six equal DEPs form three agonist-antagonist sets which actuate the TPM along three uncoupled orthogonal directions. That is, referring to Fig. 1, the DEP sets $\{2b,3a\}$, $\{1b,2a\}$ and $\{1a,3b\}$ act along the orthogonal directions of motion of the P pairs of limbs 1, 2 and 3 (hereafter called x, y and z directions) respectively, the first (second) DEP in each set providing a pushing force in the positive (negative) axis direction.

3 Static Analysis of the Manipulator Concept

Owing to the modularity of the proposed concept, the static analysis of the overall system founds on the solution of the basic DEP element shown in Fig. 3. In the literature, the statics of DE actuated mechanisms is usually solved via the d'Alembert principle by also accounting for the distributed forces (stresses) of electric nature

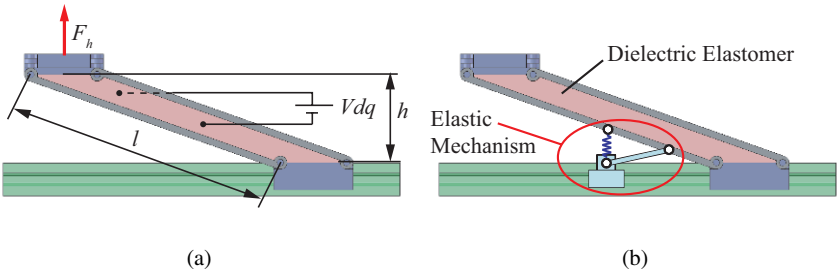


Fig. 3 DEP element (a), DEP with mechanism for stiffness adjustment (b).

that are induced in the elastomer as a result of electric field presence [8, 9]. Here, a more compact solution is obtained via energetic principles. The balance of electro-mechanical energy which flows through the DEP when a constant electric potential difference V (hereafter also called voltage) is applied by a power supply between its electrodes, is

$$d\Psi + dU = F_h dh + V dq \tag{1}$$

where $d\Psi$ is the variation of the isochoric hyper-elastic DEP strain-energy function Ψ [9], dU is the variation of the electrical energy U stored in the DEP ($U = 0.5CV^2$, C being the DEP capacitance), dh is the variation of the DEP height h (see Fig. 3(a)), F_h is the external force acting on the DEP in the direction of h (i.e. the only force which can generate mechanical work on the DEP) and dq is the variation of the electric charge q ($q = CV$) flowing between power supply and DEP electrodes (note that Vdq is the variation of electrical work provided by the power supply). Among the above quantities, Ψ and C depend on h. In fact, owing to the kinematic constraints on its boundary, the DEP can undergo uniform deformations only which yield

$$\Psi = \nu \sum_{i=1}^{2} \mu_i (\lambda_1^{\alpha_i} + \lambda_2^{\alpha_i} + \lambda_1^{-\alpha_i} \lambda_2^{-\alpha_i} - 3)/\alpha_i \text{ and } C = \varepsilon b^2 h^2 / \nu \tag{2}$$

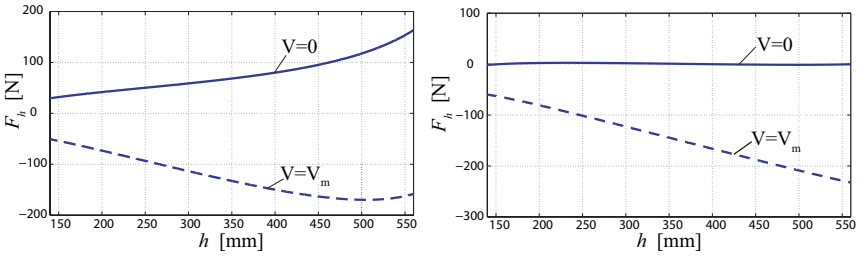
where $\mu_1 = 12.6$ Pa, $\alpha_1 = 6.06$, $\mu_2 = 26$ kPa, $\alpha_2 = 1.7$ and $\varepsilon = 4.5 \cdot 8.85e^{-12}$ F/m are material parameters of a typical DE (VHB-4905 by 3M); ν is the DEP volume ($\nu = blt_0/(\lambda_{1p}\lambda_{2p})$, b and l being the lengths of the Pa base/coupler and of its intermediate links, and t_0 being the DE film thickness in its undeformed state); λ_1 and λ_2 are the DEP principal stretches ($\lambda_1 = \lambda_{1p}[1 + [1 - (h/l)^2]^{0.5}]^{0.5}$ and $\lambda_2 = \lambda_{2p}[1 - [1 - (h/l)^2]^{0.5}]^{0.5}$, λ_{1p} and λ_{2p} being DEP pre-stretches in the configuration where the DEP is rectangle. Since $dq = VdC$ and $dU = 0.5V^2 dC$, then from Eq. (1)

$$F_h = {}^{el}F_h + {}^{em}F_h, \text{ where } {}^{el}F_h = d\Psi/dh \text{ and } {}^{em}F_h = -\varepsilon b^2 V^2 h/\nu \tag{3}$$

${}^{el}F_h$ and ${}^{em}F_h$ being the mechanical and electrical contributions to the external force F_h which are due to DE elasticity and dielectricity respectively. Generally, the mechanical term ${}^{el}F_h$ is non-linear in the DEP configuration h and can be either a pushing or a pulling force. For a given DE material, the shape of ${}^{el}F_h$ can be adjusted by properly choosing the DE pre-stretches λ_{1p} and λ_{2p}. The electric term ${}^{em}F_h$ is a pushing force (i.e. it attempts to increase parallelogram area) which depends linearly by the parallelogram height h and quadratically by the applied voltage V. That is, from a mechanical standpoint, DEP activation (via the regulation of the voltage V) is a means for actuating the Pa loop and is equivalent to the stiffness regulation of a linear negative-stiffness spring placed in parallel to the Pa loop and acting in the direction of the external force F_h. For a given DE material, such actuation (i.e. stiffness regulation) possibilities are limited by DEP electrical and mechanical failure. For instance, in every DEP working configuration: (1) the uniform electric field E ($E = \lambda_1\lambda_2 V/t_0$) acting across the DEP must not exceed the dielectric breakdown strength E_l in order to prevent DE electrical breakdown; (2) the principal elastic stresses ${}^{el}\sigma_1$ and ${}^{el}\sigma_2$ (${}^{el}\sigma_i = \lambda_i \partial\Psi/\partial\lambda_i$ for $i = 1,2$) should be greater than the "electric" stress ${}^{em}\sigma$ (${}^{em}\sigma = \varepsilon E^2$) in order to prevent DE wrinkling; (3) the principal stretches λ_1 and λ_2 must be lower than the ultimate stretch λ_l in order to prevent DE mechanical fracture ($E_l = 130$ MV/m and $\lambda_l = 5.5$ are characteristic DE material parameters). Examples of feasible Force-Length (FL) curves (i.e. $F_h - h$ curves) of the DEPs mounted on the TPM of Fig. 1 are depicted in Fig. 4 for different activation voltages ($V = 0$ in solid line and $V = V_m$ in dashed line, $V_m = 8$ kV and 7.5 kV for the cases in Figs. 4(a) and 4(b) respectively) and pre-stretch sets ($\{\lambda_{1p} = 3.2, \lambda_{2p} = 6.4\}$ in Fig. 4(a) and $\{\lambda_{1p} = 3.9, \lambda_{2p} = 4.3\}$ in Fig. 4(b)). The depicted solid and dashed lines represent the maximum and minimum forces which can be generated by the DEP; all the forces lying between these lines can be produced by regulating the voltage V according to Eq. (3). For both Figs. 4(a) and 4(b), the DEP consists of a stack of 45 DE films whose geometrical parameters are $l = 0.7$ m, $b = l/4$, $t_0 = 1$ mm. Figure 4 shows that despite the non-linear elasticity of the DE material (see Eq. (2)), different pre-stretch sets exist which make DEP FL curves quasi-linear in a rather broad range of its motion (each pre-stretch set yielding different DEP stiffness variation ranges).

Despite their quasi-linearity, FL curves like those of Fig. 4 are not ideal since they are not parallel (i.e. the available thrust ${}^{em}F_h$ is not constant) which is usually a desirable feature providing uniform system behaviour, enabling optimal use of the available output work and easing overall system control [9]. Adoption of two DEPs in agonist-antagonist configuration for each DOF of the TPM makes it possible to overcome this issue.

For instance, if the Agonist and Antagonist DEPs (AADEP) are connected so that their heights h and h' satisfy $h' = (l - h)$, AADEP reciprocal activation enables generating parallel FL curves (i.e. constant thrusts) over the full range of actuator motion (in this case a 420 mm stroke). This is shown in Figs. 5(a) and 5(b) which report the FL curves (i.e. $(F_h - F_{h'})$, where $F_{h'}$ is expressed by Eq. (3) with h replaced by h') of two different AADEP actuators based on the DEPs of Figs. 4(a) and 4(b) respectively. The parallel solid and dashed lines correspond to the recip-



(a) (b)

Fig. 4 FL ($F_h - h$) curves of two DEPs (a and b) with different pre-stretch sets.

rocal activation sets $\{V = 0, V' = V_m\}$ and $\{V = V_m, V' = 0\}$, V and V' being the activation voltages of the two DEPs in the same AADEP actuator. All the forces lying between these lines can be generated via proper selection of V and V'. Along with uniform graduation of force, AADEP actuators enable the simultaneous regulation of their intrinsic stiffness. This is shown in Fig. 5 by the intersecting dotted and dash-dotted lines which correspond to the AADEP co-activation sets $\{V = 0, V' = 0\}$ and $\{V = V_m, V' = V_m\}$. All the FL curves with varying slope lying within the pencil spanned by these lines can be generated via equal activation (i.e. $V = V'$) of the AADEPs. Owing to the quasi-linearity of the FL curves shown in Fig. 5, the overall AADEP actuator force ($F_h - F_{h'}$) and its stiffness ($dF_h/dh - dF_{h'}/dh$) can be approximated by F_h^* and K^*

$$F_h^* = K^* h + (V'^2 - V_m^2/2) l\varepsilon b^2/\nu,\ K^* = (V_m^2 - V^2 - V'^2)\varepsilon b^2/\nu, \tag{4a}$$

$$F_h^* = K^* h + V'^2 l\varepsilon b^2/\nu,\ K^* = (-V^2 + V'^2)\varepsilon b^2/\nu, \tag{4b}$$

where Eq. (4a) and (4b) hold for the cases of Figs. 5(a) and 5(b) respectively. The linear approximates F_h^* related to the four different activation sets $\{V = 0, V' = 0\}$, $\{V = V_m, V' = V_m\}$, $\{V = V_m, V = 0\}$ and $V = 0$, $V = V_m\}$ are reported in Fig. 5 with asterisks in order to show good agreement with the effective curves. The maximal deviations of Eqs. (4a) and (4.b) from the effective FL characteristics are ± 12 N and ± 3.8 N respectively (i.e. $\pm 6\%$ and $\pm 2.6\%$ of the maximum/minimum force, ± 200 N, and of the averaged maximum/minimum force, ± 146 N, which are generated by the AADEPs considered in Figs. 5(a) and 5(b) respectively).

Considering the AADEP actuator as a linear spring, Eq. (4) shows that open-loop control of V and V' enables mutual regulation of spring stiffness K^* and dead-length h^* (i.e. the zero-force equilibrium point)

$$h^* = l(V'^2 - V_m^2/2)/(V^2 + V'^2 - V_m^2), \tag{5a}$$

$$h^* = l(V'^2)/(V^2 + V'^2), \tag{5b}$$

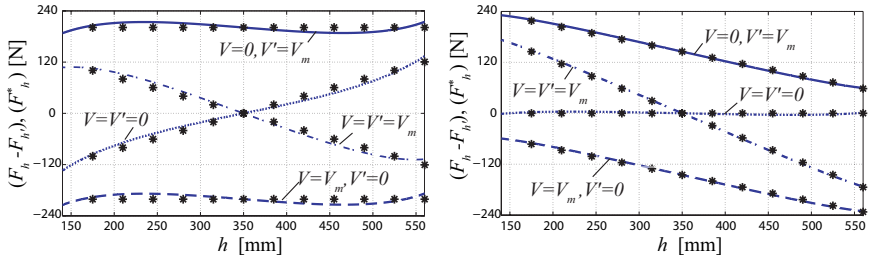


Fig. 5 FL curves of two AADEPs (a and b) with different pre-stretch sets.

where Eqs. (5a) and (5b) hold for the cases of Figs. 5(a) and 5(b) respectively, without requiring any feed-forward control law based on a position sensor measure (as impedance control schemes do) which often introduces vibrations due to quantization error, time-delays and low servo rates. The dead-length h^* can be set at any location h within the AADEP stroke. The open-loop regulation of K^* is within the range ± 0.57 N/mm (i.e. a 1.14 N/mm span) for the case of Fig. 5(a) and from 0 to 0.83 N/mm for the case of Fig. 5(b). Force-Length curves with controllable negative stiffness (such as those of Fig. 5) enable easy generation and regulation of unstable modes of interaction with the user, which are very important in rehabilitation procedures and skills training (where users have to learn how to compensate for instabilities [15]), but also in many haptic applications like the rendering of convex surfaces. If negative stiffness is not needed/desired, the FL curves of Fig. 5(b) (and similarly those of Fig. 5(a)) can be adjusted to positive stiffness values (i.e. stiffness variation from 0 to +0.83 N/mm) by adding in parallel to the AADEP an elastic mechanism providing a linear FL curve with positive stiffness and dead-length equalling 0.83 N/mm and $l/2$ respectively (an example of such a mechanism is depicted in Fig. 3(b)).

Force and stiffness magnitudes (200 N and 0.83 N/mm) produced by the considered AADEPs are in the range of those humans generate at their arm ends during either posture maintenance or movement execution in both stable and unstable environments [15, 16]. Owing to the chosen TPM architecture, the proposed manipulator concept can generate decoupled FL profiles alike those of Fig. 5 along the x, y and z directions. That is, the proposed manipulator concept enables easy generation and regulation of end-effector force fields (and in particular of stiffness ellipsoids) similar to those developed by humans for achieving stable, accurate, fast and robust control of multi-joint posture and motion [15, 16]. Such a human-like behaviour is expected to provide the proposed manipulator concept with improved safety and interaction performances especially in association with biologically-inspired control paradigms such as the “equilibrium point” hypothesis [17].

4 Dielectric Elastomer vs. Traditional Actuation

Along with human matched workspace, forces and adjustable springiness, actuators for safe interactive robots should also feature low mass and effective inertia, and possibly low cost.

With regards to mass and inertia, owing to the very low density of DE materials (less than 1000 kg/m^3), the AADEP with FL characteristics depicted in Fig. 5(a) (having $\pm$200N adjustable force) is expected to possess a mass M, $M \approx 0.5$ kg, and a variable effective inertia (reflected to the TPM platform) M_{eq}, $M_{eq} = [1/(1-(h/l)^2)+1/(1-(1-h/l)^2)M/3$ (which can be obtained via the kinetic energy equivalence), ranging between 0.24 kg and 0.35 kg. These values are less than one order of magnitude with respect to the masses and the reflected inertias of conventional gear-head electric motors which could be employed to actuate the same Pa loops in order to produce output forces similar to those of the considered AADEPs. For instance, a state-of-the-art gear-head brushless motor with 150 Nm nominal output torque, 2.5 kg weight, reduction ratio n, $n = 150$, and motor rotor inertia J, $J = 0.8$ kg/cm^2, yields on the TPM platform a variable reflected inertia M'_{eq}, $M'_{eq} = Jn^2/(l^2-h^2)$, ranging between 4 kg and 10 kg (only the best case in which the gear-head motor is fixed with respect to the manipulator base is considered here). Furthermore, AADEP mass and reflected inertia are also one order of magnitude lower than the stator and slider masses of direct-drive brushless linear motors that are often used for actuating the P pairs of isotropic TPMs. For instance, a state-of-the-art direct-drive brushless linear motor with 150 N nominal output force and 400 mm stroke possesses stator and slider masses roughly equalling 2 kg and 3 kg respectively.

With regards to costs, owing to the low price of DE materials, the expected cost of an AADEP is about 100 € which is far less than the price of both gear-head and linear electric motors with similar force performances (driving electronics are not considered here since their costs are going to be quite similar, probably cheaper for AADEPs [12]).

5 Conclusions

This paper has shown via static analysis that isotropic translational parallel mechanisms with agonist-antagonist dielectric elastomer actuators potentially provide an adequate and cost effective solution concept for the development of novel generation of interactive robots with low effective inertia and human matched workspace, forces and adjustable springiness. Future work will address the optimization and development of a real prototype along with its experimental validation and control.

References

1. Gogu, G.: Fully-isotropic parallel mechanisms: An innovative concept for haptic devices. In *Product Engineering: Tools and Methods Based on Virtual Teality*, Talaba and Amditis (Eds.), Springer, pp. 169–194 (2008).
2. Hollerbach, J.M., Hunter, I.W., Ballantyne, J.: A comparative analysis of actuator technologies for robotics. In *The Robotics Review 2*, O. Khatib, J.J. Craig and T. Lozano-Pérez (Eds.), MIT Press (1992).
3. Pratt, G., Williamson, M.: Series elastic actuators. In *Proc. of IEEE/RSJ Int. Conf. on Intell. Robots and Systems*, Pittsburgh, PA, Vol. 1, pp. 399–406 (1995).
4. Zinn, M., Roth, B., Khatib, O. and Salisbury, J.K.: New actuation approach for human-friendly robot design. *The International Journal of Robotics Research*, **23**, 379–398 (2004).
5. Bicchi, A., Tonietti, G.: Fast and soft arm tactics: Dealing with the safety-performance trade-off in robot arms design and control. *IEEE Robotics and Automation Magazine*, **11**, 321–354 (2004).
6. Conti, F., Khatib, O.: A new actuation approach for haptic interface design, *The International Journal of Robotics Research*, **28**(6), 834–848 (1966).
7. Madden, J.D.W., Vandesteeg, N.A., Anquetil, P.A., Madden, P.G.A., Takshi, A., Pytel, R.Z., Lafontaine, S.R., Wieringa, P.A., Hunter, I.W.: Artificial muscle technology: Physical principles and naval prospects, *IEEE Journal of Oceanic Engineering*, **29**(3) (2004).
8. Plante, J.S.: Dielectric elastomer actuators for binary robotics and mechatronics, Ph.D. Thesis, Massachusetts Institute of Technology (2006).
9. Vertechy, R., Berselli, G., Parenti Castelli, V., Vassura, G.: Optimal design of lozenge-shaped dielectric elastomer linear actuators: Mathematical procedure and experimental validation, *Journal of Intelligent Material Systems and Structures*, in press (2009).
10. Kovacs, G., Lochmatter, P., Wissler, M.: An arm wrestling robot driven by dielectric elastomer actuators, *Smart Mat. Struct.*, **16**, 306–317 (2007).
11. Kovacs, G., Dring, L., Michel S., Terrasi, G.: Stacked dielectric elastomer actuator for tensile force transmission, *Sensors and Actuators A: Physical*, **155**(2), 299–307 (2009).
12. Babic, M., Vertechy, R., Berselli, G., Lenarcic, J., Parenti Castelli, V., Vassura, G.: An electronic driver for improving the open and closed loop electro-mechanical response of Dielectric Elastomer actuators, *Mechatronics*, in press (2009).
13. Gogu, G.: *Structural Synthesis of Parallel Robots Part 2: Translational Topologies with Two and Three Degrees of Freedom*. Springer (2009).
14. Nathan, R.H.: A constant force generation mechanism, *ASME Journal of Mech, Transm., and Autom. in Design*, **107**, 508–512 (1985).
15. Franklin, D., Burdet, E., Rieko, O., Kawato, M., Milner, T.E.: Functional significance of stiffness in adaptation of multijoint movements to stable and unstable dynamics, *Exp. Brain Research*, **151**, 145–157 (2003).
16. Shadmer, R., Mussa-Ivaldi, F.A., Bizzi, E.: Postural force fields of the human arm and their role in generating multijoint movements, *The Journal of Neuroscience*, **13**(1), 45–62 (1993).
17. Feldman, A.G.: Functional tuning of the nervous system with control of movement or maintenace of posture, *Biophysics*, **11**, 565–578 (1966).

Using Redundancy in Serial Planar Mechanisms to Improve Output-Space Tracking Accuracy

S. Ambike[1], J.P. Schmiedeler[2] and M.M. Stanišić[2]

[1]*The Ohio State University, Columbus, Ohio, USA; e-mail: ambike.1@osu.edu*
[2]*University of Notre Dame, Notre Dame, Indiana, USA;*
e-mail: {schmiedeler.4,stanisic.1}@nd.edu

Abstract. Path tracking using planar mechanisms with one degree of redundancy can be enhanced by matching (when possible) or approximating second-order path properties with first-order joint coordination. Second-order tracking can reduce the frequency of feedback for the desired accuracy, and since this paper provides analytical expressions for the joint speed ratios, this advantage comes at no additional computational cost. Examples show that tracking solutions with this method are locally more accurate compared to unweighted pseudoinverse solutions. Therefore, the feedback frequency for a desired tracking accuracy can be reduced, potentially resulting in a reduced computational cost of path tracking.

Key words: Geometric tracking, speed-ratio control, redundancy resolution, planar mechanisms.

1 Introduction

Redundant manipulators are increasingly employed in useful practical tasks that are specified in terms of a geometric path to be followed by the end-effector. Redundant degrees of freedom make it possible to achieve objectives such as avoiding collisions, joint limits and/or singular configurations. However, objective criteria need to be specified to resolve the kinematic redundancy. Kinematic performance metrics, such as locally bounded joint-space velocities, involve computation of damped least-squares solutions [1], although such pseudoinverse-based control cannot avoid singular configurations [2]. Alternatively, time-optimal control uses the manipulator dynamics to minimize the performance time, which is a solved problem for non-redundant manipulators [3]. For kinematically redundant manipulators, numerical procedures have been proposed in [4] to achieve path-constrained time-optimal control. A computationally efficient feedback-control law is developed in [5] that provides joint forces/torques for a redundant manipulator while minimizing the output-space tracking error. These methods use information from the output-space path up to the first-order only, whereas the definition of the desired output-space path contains more geometric information in the form of higher order derivatives. Effective utilization of this path information can reduce the required feedback fre-

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_15,

quency for a desired tracking accuracy and potentially, the computational cost of path tracking.

In [6], a curvature-theory-based approach is developed that maps the first- and second-order information of the output-space path onto the first- and second-order geometric properties of the joint space path for non-redundant, planar mechanisms. This method is extended to using the third-order path properties of constant curvature output-space paths in [7]. Recently, this approach has been generalized to path tracking with spatial, non-redundant systems using the geometric properties of the output-space path up to any order [8]. The present work illustrates how this approach, termed speed-ratio (SR) control, can be effectively applied to redundant, planar manipulators. It resolves kinematic redundancy by including higher-order geometric information from the desired path into the problem formulation. The resulting system of polynomial equations has analytical solutions for planar, three-degree-of-freedom (DOF) mechanisms, making this method computationally efficient.

The remainder of this paper is organized as follows. Section 2 describes the SR paradigm. Section 3 develops the application of SR control to three-DOF planar manipulators. Section 4 presents numerical examples. Section 5 gives conclusions.

2 Speed-Ratio Control for Non-Redundant Manipulators

The SR approach to control non-redundant mechanisms, developed in [8], is briefly described here. A trailing subscript(s) for a quantity indicates derivative with respect to the subscript(s), and a zero after the subscript(s) indicates that the derivative has been evaluated in the zero position. For example,

$$\overline{r}_{\theta_1} := \frac{d\overline{r}}{d\theta_1}, \quad \overline{r}_{\theta_1\theta_2 0} := \left.\frac{d^2\overline{r}}{d\theta_1 d\theta_2}\right|_0 .$$

The zero position is defined by the initial increments in the joint variables all being zero.

The geometric path tracking approach involves a reparameterization of the forward kinematic map of a tracking mechanism using the displacement of one joint, called the *leading joint*, as the independent variable. Motion of the other joints are related to that of the leading joint via Taylor series, the constants of which are called *speed ratios*. The jth-order speed ratio relating the motion of joint i to that of the leading joint when evaluated in the zero position is denoted by $n_i^{(j)}$. For $j = 1$, the superscript is omitted. Therefore,

$$n_3^{(3)} := \left.\frac{d^3\theta_3}{d\theta_1^3}\right|_0 \quad \text{and} \quad n_2^{(1)} := n_2 = \left.\frac{d\theta_2}{d\theta_1}\right|_0 .$$

For a d-DOF, non-redundant mechanism ($d = 2$ for planar and 3 for spatial systems), let θ_i denote the joint variables. Without loss of generality, the joint motions are coordinated using θ_1 as the leading joint:

$$\theta_i = n_i\theta_1 + \frac{1}{2!}n_i^{(2)}\theta_1^2 + \frac{1}{3!}n_i^{(3)}\theta_1^3 + \cdots \qquad i = 2, \cdots d. \tag{1}$$

The path $\overline{r}$ generated by the controlled point on the end effector (EE) is expressed as a function of θ_1 as

$$\overline{r}(\theta_1) = \overline{r}_0 + \overline{r}_{\theta_1 0}\theta_1 + \frac{1}{2!}\overline{r}_{\theta_1\theta_1 0}\theta_1^2 + \frac{1}{3!}\overline{r}_{\theta_1\theta_1\theta_1 0}\theta_1^3 + \cdots, \tag{2}$$

where $\overline{r}_0$ is the initial EE position, and $\overline{r}_{\theta_1 0} = \overline{r}(n_i)$, $\overline{r}_{\theta_1\theta_1 0} = \overline{r}(n_i, n_i^{(2)})$, and so on for higher order derivatives. Equation (1) allows the forward kinematics of the manipulator to be expressed as a curve in the output space parameterized in terms of θ_1, given by Eq. (2). The kinematics can be similarly characterized in terms of the arc length. However, arc-length parameterization fails for singular poses of the mechanism, whereas the present parameterization allows accurate geometric tracking even for singular poses [6]. The generated path is described by a Frenet-Serret (FS) frame, the components of which are defined by the vector terms in Eq. (2). For example, the tangent of the frame is parallel to $\overline{r}_{\theta_1 0}$, and the normal is parallel to $\overline{r}_{\theta_1\theta_1 0}$. Note that this description involves the unknown speed ratios. The geometry of the desired path is also described using a FS frame, but in this case, all of the quantities defining the frame are known. Control of the EE path geometry is achieved by matching the geometric properties of the frames describing the generated and desired paths. For example, matching the first-order geometric property means forcing the tangent of $\overline{r}(\theta_1)$ to be parallel to the tangent of the desired path. The resulting *coordination equation* is solved for the first-order speed ratios. Each higher order of geometry matching yields coordination equations that can be solved for the speed ratios of the corresponding order. The coordination equations for the first two orders are [8]

$$\overline{r}_{\theta_1 0} \times \hat{T} = \overline{0}, \tag{3}$$

$$\overline{r}_{\theta_1\theta_1 0} \times \hat{T} + \kappa_d(\overline{r}_{\theta_1 0} \cdot \overline{r}_{\theta_1 0})\hat{B} = \overline{0}, \tag{4}$$

where $\hat{T}$ and $\hat{B}$ are the tangent and binormal of the FS frame and κ_d is the curvature. Note that as control of path geometry is achieved, the corresponding geometric properties of the FS frames for the desired and generated paths become identical, and it becomes unnecessary to explicitly distinguish between them. Equation (3) can be solved for n_2 and n_3, Eq. (4) gives $n_2^{(2)}$ and $n_3^{(2)}$, and so on. Matching pth-order geometric properties yields the pth-order speed ratios. Equations (1) and (2) then determine the motion of the EE.

3 Speed-Ratio Control for Three-DOF, Planar Manipulators

For redundant mechanisms, assuming that the pose is non-singular, there will exist a family of joint-velocity solutions that achieve first-order tracking. A unique solution from this set can be chosen based on higher-order geometric information of the desired path. The order of the geometric properties that can be used to make this choice is generally determined by the order of joint control and the degree of redundancy in the system. For example, with first-order joint coordination, or joint-velocity control as is assumed in this paper, a three-DOF system can use second-order path properties, and a four-DOF system can use second- and third-order path properties. By applying the coordination equations to the redundant system, a set of polynomial equations is obtained that provide insight into the tracking capabilities of the mechanism and, for a three-DOF mechanism, provide analytical solutions for the best possible second-order tracking performance. In contrast, a *non-redundant*, planar mechanism must use second-order joint coordination to track path curvature, and accurate curvature tracking is ensured for non-singular mechanism poses.

For a general, planar three-DOF system, let θ_1, θ_2, and θ_3 be the joint variables. The leading joint is θ_1, and n_2 and n_3 are the two first-order speed ratios. Equation (3) provides one linear equation, and Eq. (4) provides one quadratic equation in the two unknown speed ratios.

$$A_1 n_2 + A_2 n_3 + A_3 = 0, \tag{5}$$

$$B_1 n_2^2 + B_2 n_3^2 + B_3 n_2 n_3 + B_4 n_2 + B_5 n_3 + B_6 = 0, \tag{6}$$

where the coefficients A_i and B_i are functions of the desired path tangent and the mechanism's geometry and current pose. Since Eq. (4) is linear in the desired curvature κ_d, a univariate polynomial can be obtained from Eqs. (5) and (6).

$$\Omega(n_2) := (a_1\kappa_d + b_1)n_2^2 + (a_2\kappa_d + b_2)n_2 + (a_3\kappa_d + b_3) = 0, \tag{7}$$

where the coefficients a_i and b_i are derived from A_i and B_i, and so they are also functions of the desired tangent and the mechanism geometry and pose. The curvature κ_d can be accurately matched if Eq. (7) has real solutions, the condition being,

$$\Delta := \left(a_2^2 - 4a_1a_3\right)\kappa_d^2 + \left(2a_2b_2 - 4a_1b_3 - 4b_1a_3\right)\kappa_d + \left(b_2^2 - 4b_1b_3\right) \geq 0. \tag{8}$$

By explicit calculation, it can be shown that for three-DOF planar mechanisms of all possible morphologies, the function $a_2^2 - 4a_1a_3 = 0$. Therefore,

$$\Delta = \kappa_d C + D, \tag{9}$$

where $C = 2a_2b_2 - 4a_1b_3 - 4b_1a_3$, and $D = b_2^2 - 4b_1b_3$. Note that κ_d is positive by definition. Therefore, if C and D are both positive, Δ is also positive for any κ_d. In this case, two solutions are obtained from Eqs. (7) and (5). The rate of change in curvature of the desired path can be used to choose from these two solutions, as illustrated in the numerical example in the following section. If $\Delta < 0$, the curvature

cannot be matched exactly. In this case, the tracked curvature is treated as a variable, denoted by κ. The discriminant in Eq. (9) is now a function of κ, and a value $\kappa = \kappa_t$ can be obtained from the condition in Eq. (8) such that Eq. (7) yields real roots for n_2. If C and D are not both negative, the solution to the equation $\Delta(\kappa) = 0$ gives a positive generated curvature $\kappa_t = -\frac{D}{C}$, such that the error $|\kappa_t - \kappa_d|$ is minimum. The minimality condition is ensured by the continuity and monotonicity of $\Delta(\kappa)$.

When $C, D < 0$, the range of κ for which the condition in Eq. (8) is satisfied is given by $-\frac{D}{C} \geq \kappa$. Only negative values for κ are possible, indicating that the mechanism can only move along the desired tangent such that the normal vector of the generated path is in the opposite direction of the normal of the desired path. In this case, the generated path with the *smallest curvature magnitude* (the path with the greatest radius) will be the most accurate. Therefore, the smallest negative $\kappa_t = -\frac{D}{C}$ that satisfies the condition in Eq. (8) is the solution.

In conclusion, with first-order joint coordination, a three-DOF mechanism can track a desired path with a given curvature κ_d if the quantity Δ in Eq. (9) is greater than zero, and the speed ratios can be obtained from Eqs. (7) and (5). If $\Delta < 0$, the *best possible* generated path has curvature $\kappa_t := -\frac{D}{C}$. This solution can be substituted into Eq. (7) to obtain n_2, following which, Eq. (5) yields the value of n_3.

The three-prismatic manipulator is an exception to the above scheme. For this manipulator, C and D are zero, and the function Ω in Eq. (7) vanishes. This is a (perhaps intuitive) result indicating that the EE cannot move along curved paths with constant joint velocities for a three-DOF Cartesian robot, and therefore, path curvature cannot be tracked with first-order joint control.

The technique described here is applicable in principle to larger planar systems as well as spatial systems. However, the analyticity is quickly lost, since the size of the polynomial equations becomes large. For example, for a four-DOF, planar system employing first-order joint control, the first-, second-, and third-order coordination equations will be linear, quadratic, and sextic equations in three variables, respectively. A search algorithm must be employed to obtain real solutions for all three equations. This is a limitation of the approach. However, a useful result will be the characterization of the solution for the curvature-tracking problem before solving the full system. This is a subject of future work.

4 Examples

A 3-revolute (3R) mechanism has link lengths l_i, and θ_{i0} denote the initial values of the joint variables that define the zero position. Joint angle θ_{i0} is measured counterclockwise from link l_{i-1} to link l_i. The joint angle θ_{10} is measured with respect to a fixed reference axis. Two examples are provided, one with a positive value for the discriminant Δ, and the second with a negative value for Δ.

Example 1: Positive Δ
The link lengths are $l_1 = 1.15$, $l_2 = 1.5$, and $l_3 = 1.8$ in arbitrary length units. The initial pose, or zero position, is defined by $\theta_{10} = -14^\circ$, $\theta_{20} = -10^\circ$, and $\theta_{30} = -205^\circ$.

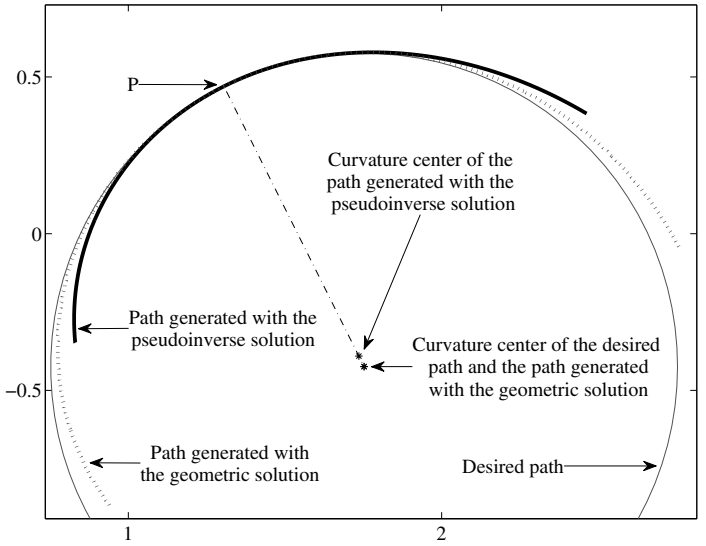


Fig. 1 The planar 3R mechanism is required to track the circular arc passing through point P. The dashed curve is the geometric tracking solution. The solid curve is the pseudoinverse tracking solution. Curvature centers for the three paths are shown. Only the curvature centers of the desired path and the path generated with the geometric solution match.

The desired path is a circular arc passing through the controlled point on the EE, point P. The desired tangent is $\hat{T} = [0.8944 \quad 0.4472 \quad 0]^T$, and κ_d is 1. Also, the rate of change of curvature is zero. The joint variable θ_1 is chosen as the leading joint variable. These system parameters yield constants A_i, a_i and b_i:

$$\begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix} = \begin{bmatrix} 0.2345 \\ 1.4247 \\ 0.0485 \end{bmatrix}, \quad \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} -6.4670 \\ -26.3920 \\ -26.9267 \end{bmatrix}, \quad \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 6.7015 \\ 27.8167 \\ 26.9752 \end{bmatrix}.$$

This gives $C = -48.6844$, $D = 50.6686$, and $\Delta = 1.9842$. Therefore, Eqs. (5) and (7) provide two solutions for the speed ratios: $\{n_2 = -0.0343, n_3 = 3.0319\}$, and $\{n_2 = -6.0402, n_3 = -3.7154\}$. The first solution has a rate of change of curvature of -0.8262, and the second solution has a rate of change of curvature of 0.4414. Since the latter value is closer to the desired value 0, the second solution is chosen.

Alternative first-order speed ratios obtained as ratios of the joint velocities from the Jacobian pseudoinverse are $\{n_2 = 2.9052, n_3 = 6.3343\}$. Note that the norm of the pseudoinverse solution will be lower than the norm of the geometric solution once the EE-velocity magnitudes for both solutions are made equal by proper choice of the leading joint velocity. In Fig. 1, the EE paths generated by implementing the chosen geometric solution and the pseudoinverse solution are plotted until the position error, defined as the minimum distance of the EE from the desired path, reaches 0.025. Clearly, superior tracking accuracy is achieved by implementing speed-ratio

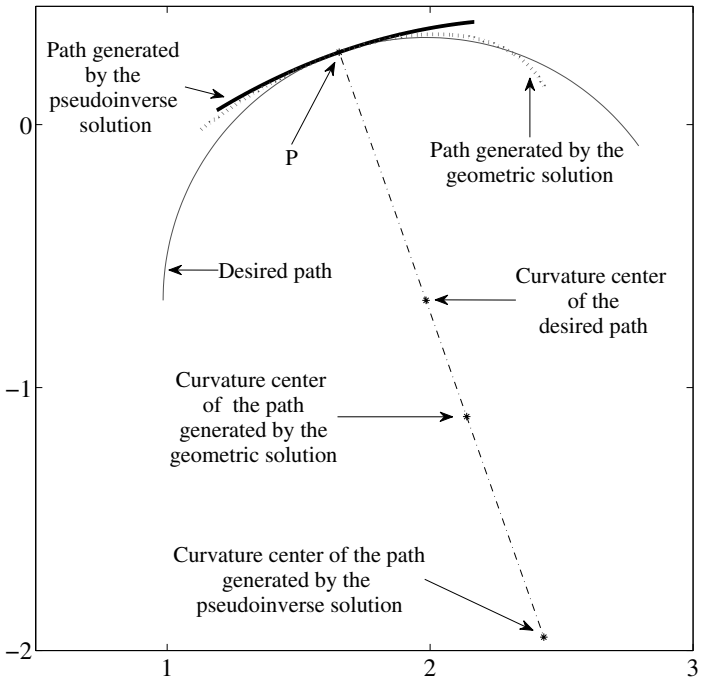


Fig. 2 The planar 3R mechanism is required to track the circular arc passing through point P. The dashed curve is the geometric tracking solution, and the solid curve is the pseudoinverse tracking solution. The curvature centers of the three paths are shown.

control. The EE path obtained from the geometric solution follows the desired path more closely and stays close to the desired path for a longer portion of the desired path compared to the pseudoinverse solution.

Example 2: Negative Δ

The zero position of the mechanism in the previous example is redefined by $\theta_{10} = -14^\circ$, $\theta_{20} = -40^\circ$, and $\theta_{30} = -205^\circ$. The desired path is a circular arc passing through point P with tangent $\hat{T} = [0.9439 \quad 0.3304 \quad 0]^T$ and κ_d is 1. The rate of change in curvature is zero. The joint variable θ_1 is chosen as the leading joint variable. The constants A_i, a_i and b_i are

$$\begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix} = \begin{bmatrix} -9.6866 \\ -40.7486 \\ -42.8541 \end{bmatrix}, \quad \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} -19.3305 \\ -102.2244 \\ -135.1469 \end{bmatrix}, \quad \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 3.4823 \\ 28.8917 \\ 49.2146 \end{bmatrix}.$$

This gives $C = -219.0375$, $D = 149.219$, and $\Delta = -69.8184$, indicating that the desired radius cannot be accurately matched. The tracked curvature is $\kappa_t = -\frac{D}{C} = 0.6812$, and the corresponding error is $|\kappa_d - \kappa_t| = 0.3188$. The geometric speed ratios are obtained by using the computed value of κ_t in Eqs. (7) and (5). The geomet-

ric and the pseudoinverse-based speed ratios, respectively, are $\{n_2 = -2.1033, n_3 = -0.7671\}$, and $\{n_2 = -0.7398, n_3 = -4.3965\}$. Figure 2 plots the paths generated by both solutions until the position error reaches 0.025. The curvature centers of neither solution match the desired curvature center. However, the curvature center obtained from the geometric solution is closer to the desired curvature center. Further, with first-order coordination, the curvature center of the generated path cannot be any closer to the desired curvature center given the tangent direction and the mechanism's pose. Therefore, the path obtained from the geometric solution is the most locally accurate path that can be achieved with first-order coordination.

5 Conclusions

Speed-ratio control is applied to resolve redundancy in path tracking with three-DOF planar mechanisms. Analytically obtained first-order joint motions minimize the difference between the curvatures of the generated and the desired output-space paths. Two examples are provided that compare the tracking performance of speed-ratio control with pseudoinverse solutions. The use of higher-order path information yields generated paths that track the desired path more closely. Less feedback will be required to achieve the desired tracking accuracy, and therefore, this method has the potential to reduce the computational cost of path tracking.

References

1. Chiaverini, S., Oriolo, G., Walker, I.A.D.: Kinematically redundant manipulators. In: Siciliano, B., Khatib, O. (Eds.), *Springer Handbook of Robotics*, Springer Verlag, pp. 245–268 (2008).
2. Baillieul, J., Hollerbach, J., Brockett, R.: Programming and control of kinematically redundant manipulators. In: *Proceedings of 23rd IEEE Conference on Decision and Control*, pp. 768–774 (1984).
3. Shin, K.G., McKay, N.D.: Minimum-time control of robotic manipulators with geometric path constraints, *IEEE Transaction on Automatic Control*, **AC 30**(6), 531–541 (1985).
4. Galicki, M.: Time-optimal control of kinematically redundant manipulators with geometric constraints. *IEEE Transactions on Robotics and Automation*, **16**(1), 89–93 (2000).
5. Galicki, M.: Adaptive control of kinematically redundant manipulator along a prescribed geometric path. In: Kozlowski, K. (Ed.), *Robot Motion and Control*, Springer, Berlin/Heidelberg, pp. 129–139 (2006).
6. Lorenc, S.J., Stanišić, M.M., Hall Jr. A.S.: Application of instantaneous invariants to the path tracking control problem of planar two degree-of-freedom systems: A singularity-free mapping of trajectory geometry, *Mech. Mach. Theory*, **30**(6), 883–896 (1995).
7. Lorenc, S.J., Stanišić, M.M.: Third-order control of a planar system tracking constant curvature paths, In: Lenarcic, J., Ravani, B. (Eds.), *Advances in Robot Kinematics and Computational Geometry*, Kluwer Academic Publishers, Dordrecht, pp. 229–238 (1994).
8. Ambike, S., Schmiedeler, J.P., Stanišić, M.M.: Geometric, spatial path tracking using non-redundant manipulators via speed-ratio control. In *Proceedings ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, IDETC/CIE, to appear (2010).

PART 3

Combining Structural and Kinematic Analysis Using Interval Analysis for a Wire-Driven Manipulator

S. Abdelaziz, P. Renaud, B. Bayle and M. de Mathelin

LSIIT-CNRS Strasbourg University, France;
e-mail: salih.abdelaziz@lsiit.u-strasbg.fr

Abstract. In this article, we consider wire-driven manipulators with instrumented truss structure for cable tension evaluation. For such mechanisms, we propose to perform the structural analysis using interval analysis. We introduce an algorithm to perform simultaneously kinematic and structural analysis. Examples are introduced to show the interest of the approach.

Key words: Wire-driven manipulator, interval analysis, structural analysis.

1 Introduction

Giving assistance to radiologists during percutaneous procedures [1] is currently a great challenge for roboticists: MRI scanners are used for their very interesting image details, however they introduce severe design constraints. The two principal ones are the limited space to introduce a robotic system in the scanner ring, and the presence of a strong permanent magnetic field. In that context, we propose to develop a robotic assistance by means of a wire-driven manipulator (WDM): actuators could then be removed from the MRI room to suppress artefacts and ensure MRI-compatibility. Furthermore, a WDM can allow us to get a very compact, lightweight robot for an easy integration in the MRI scanner, contrary to previously proposed systems [2, 3].

Placing the actuators at a distance can be performed by using pulleys to guide the cables connected to the end-effector (Fig. 1). The use of a truss for the structure seems a natural choice to minimize the size and the weight of the manipulator: such a structure has a very interesting stiffness/weight ratio. The cables are attached to the nodes of the structure, and these nodes are connected to the ground in a way that the structure is sensitive to the tension in the cables. The measurement of the truss deformation can allow the evaluation of the cable tensions, which is necessary for control purposes [4]. Even in presence of a magnetic field, deformation measurement is feasible, using optical sensors [5]. Our approach is thus to develop a WDM with instrumented truss.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_16,

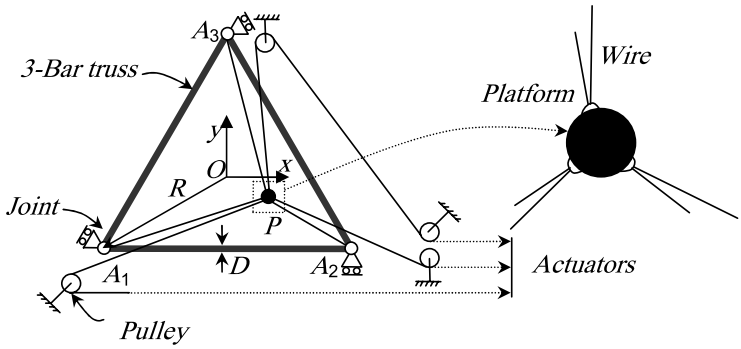


Fig. 1 Example of wire-driven manipulator with 3-bar truss structure and discarded actuators.

In this paper, we consider the first step of a design methodology, which is the development of an algorithm to characterize a given robot. We show that the structure and its instrumentation with strain sensors can be validated by using interval analysis, simultaneously to the validation of other constraints associated to WDMs. We illustrate the necessity to validate in the same time the parameters associated to the structure of the mechanism and those associated to the mechanism itself.

In the next section, approaches for the analysis of WDMs are first detailed. Foundations of structural analysis are then briefly recalled in Section 3. The proposed algorithm is introduced in Section 4. In Section 5, two examples are developed to outline the need of a simultaneous evaluation of structural and kinematic behaviour of the device. Finally, conclusions and future works are presented.

2 Related Work

Several constraints have been considered for the evaluation of the workspace of WDMs. Interferences between wires may limit the reachable workspace and has been analysed in [6–8] for parallel robots, with an approach that can be extended to WDMs. Influence of collisions between wires and cylindrical platforms was studied in [7]. Another workspace limitation comes from the unidirectional behaviour of wires, which has been investigated in [9–12], in particular using interval analysis. In this latter approach, the provided results are guaranteed, an asset that discretization methods can not provide [13]. In this paper, we thus also consider an interval analysis approach, and take into account the behaviour of the structure, a constraint generally not considered [14].

3 Structural Analysis Overview

A truss structure is composed of slight elements, called bars, connected by spherical joints, the nodes. The nodes ensure that the bars are only submitted to tension or compression. For many materials, an elastic deformation domain can be defined with the Hooke's law that links the deformation ε to the stress σ by the Young's modulus E: $\sigma = E\varepsilon$. The deformation of a bar is the ratio between the relative displacement of its ends and its length. The truss behaviour can be estimated using the direct stiffness method. This systematic approach allows the computation of the vector $\mathbf{u}$ composed of the nodes displacements from the external load vector $\mathbf{f}_e$ and a structural stiffness matrix $\mathbf{K}_s$: $\mathbf{u} = \mathbf{K}_s^{-1}\mathbf{f}_e$. The matrix $\mathbf{K}_s$ can be evaluated from the truss geometry and the bar properties. Deformations and stresses in the bars can then easily be derived.

The elastic deformation domain is generally characterized by two thresholds. During the truss analysis the following constraints have therefore to be fulfilled:

$$\sigma_c < \sigma_i < \sigma_t \, , \, i \in [1, b] \tag{1}$$

with σ_i the stress in the bar i of a b-bar truss, σ_c and σ_t respectively defining the limits in compression and tension. Buckling of the bars must also be avoided:

$$\sigma_i > \sigma_b \, , \, i \in [1, b] \tag{2}$$

The stress σ_b is expressed using the Euler's formula: $\sigma_b = \pi^2 EIA/L^2$, where I is the area moment of inertia, A is the cross sectional area of the bar and L its length.

4 The Proposed Algorithm

4.1 Criteria for Workspace Delineation

The workspace of a WDM is usually considered as the intersection between the workspace for which cables are able to counterbalance any wrench applied to the end-effector, called the Wrench-Feasible Workspace (WFW, [11]) and the (cable-cable, cable-platform) collision-free workspace. The use of an instrumented truss for cable tension evaluation induces novel constraints. Firstly, conditions (1) and (2) have to be verified. Secondly, the measurement range ε_0 of the strain sensors must be respected. This constraint can be expressed as a condition on the stress, using Hooke's law, so that a maximum value σ_0 must not be exceeded:

$$-\sigma_0 \leq \sigma_i \leq \sigma_0, \, i \in [1, b] \tag{3}$$

Finally, no interference must occur between the bars and the cables or the platform.

4.2 Structural Analysis Using Interval Analysis

In [11], the WFW is determined using interval analysis. The relationship between the vector τ describing the tensions in the cables and the wrench $\mathbf{f}$ applied on the end-effector is well-known:

$$\mathbf{W}\tau = \mathbf{f} \tag{4}$$

where $\mathbf{W}$ is the wrench matrix, which is a function of the end-effector pose $\mathbf{X}$. The components of $\mathbf{X}$, τ and $\mathbf{f}$ all belong to intervals, denoted in the following by the superscript I, so that the WFW can be evaluated using interval analysis, starting from the condition:

$$\forall \mathbf{W} \in \mathbf{W}^I, \forall \mathbf{f} \in \mathbf{f}^I, \exists \tau \in \mathscr{T} \text{ such that } \mathbf{W}\tau = \mathbf{f} \tag{5}$$

with, for m cables, $\mathscr{T} = \left\{ \tau \,|\, \tau_i \in [\tau_{\min}, \tau_{\max}], 1 \leq i \leq m \right\}$

In Section 3, it has been outlined that the displacements at the truss nodes can be expressed from the external load. In our context, this load is the tension of the cables. Furthermore, stresses are linearly linked to deformations, and deformations in the bars are a linear function of the node displacements. It then comes that a linear relationship can be derived between the stresses in the bars and the tensions of the cables:

$$\sigma = \mathbf{T}\tau \tag{6}$$

with σ a stress vector containing the stress in each bar: $\sigma = [\sigma_1, ..., \sigma_b]^T$ for a b-bar truss. The matrix $\mathbf{T}$, later called the stress matrix, depends on the pose $\mathbf{X}$. Conditions (1), (2) and (3) will create for a given structure an interval Σ. The components of τ are also described by intervals. The structural analysis, including the validation of the instrumentation, can as a consequence be treated with interval analysis tools. This latter approach allows us to conclude on the truss behaviour over the whole workspace.

It is important to note that we need to consider in the same time the problems described by expressions (4) and (6). Combining the two equations would allow to get a relationship between the applied wrench $\mathbf{f}$ and the stress vector σ. However, some situations may be considered as admissible in terms of stresses whereas the cable tensions exceed their maximum values.

4.3 Algorithm

We propose to estimate in a single algorithm the constraints associated to the mechanism, i.e. the limited range of admissible tensions in the cables, as well as the constraints associated to the intrumented truss structure: interferences with the structure and conditions (1), (2) and (3) defining an interval Σ. The global structure of the algorithm is obtained by combining algorithms proposed in [7, 11], adding the validation of stress constraints.

1. $i = 1, p = 1, \mathcal{B}_1 = \mathcal{W}$,
2. **While** $i \geq p$ **do**
3. $\mathbf{d}^I = [d_1{}^I, \ldots, d_b{}^I] = Compute\_DistanceVector\ (\mathcal{B}_i)$
4. **if** $\exists\, j \in [1, b] \mid \min\ (d_j^I) \leq d_s$ **go to** step 10
5. **else** $\mathbf{W}^I = Compute\_WrenchMatrix\ (\mathcal{B}_i)$
6. **if** Out $(\mathbf{W}^I, \mathbf{f}^I, \mathcal{T}) = 1$ then $i = i+1$, **go to** step 2
7. **else** **if** Feasibility $(\mathbf{W}^I, \mathbf{f}^I, \mathcal{T}) = 0$ **go to** step 10
8. **else if** $Compute\_Structure(\mathcal{B}_i) = 0$ **go to** step 10
9. **else** $i = i+1$, **go to** step 2
10. **if** the widths of all the components of $\mathcal{B}_i$ are lower than a threshold δ then **return** 0
11. **else** Bisect $(\mathcal{B}_i)$, $p = p+2$, $i = i+1$, **go to** step 2

Fig. 2 The proposed algorithm for workspace determination.

Let $\mathcal{W}$ be the prescribed workspace of dimension n to be analysed. The corresponding box is decomposed in boxes $\mathcal{B}_i$, $i \in [1, p]$ following the algorithm expressed in Fig 2.

The procedure *Compute_DistanceVector* performs the interval estimation of the distance of cables and end-effector with respect to the bars of the truss. A security distance d_s is taken into account to determine the existence of an interference.

The procedure *Compute_WrenchMatrix* determines the interval wrench matrix $\mathbf{W}^I$. In step 6, the function *Out* tests if $\mathbf{f}^I$ is fully inside $\mathbf{W}^I \mathcal{T}$. The procedure *Feasability* contains the implementation of the Rohn theorem: the system $\mathbf{W}^I \tau = \mathbf{f}^I$ is feasible if and only if the 2^n systems of linear equations $\mathbf{W}_Y \tau = \mathbf{f}_Y$ are feasible, where $\mathbf{W}_Y$ and $\mathbf{f}_Y$ are respectively the vertex matrix of $\mathbf{W}^I$ and the vertex vector of $\mathbf{f}^I$.

The procedure *Compute_Structure* is using the 2^n solutions of τ obtained during the feasability evaluation in step 7. These solutions are multiplied by the interval evaluation $\mathbf{T}^I$ of the stress matrix $\mathbf{T}$ to obtain 2^n interval evaluations σ^I of the stress vector. If at least one of the 2^n interval stress vectors is not included in the interval Σ, the procedure *Compute_Structure* returns a null value in step 8.

4.4 Remarks

In the procedure *Compute_Structure*, an overestimation of the stress matrix can occur due to interval evaluation, and as a consequence a bisection of a box can be needed. Such a problem is also encountered during for instance the computation of the WFW [11].

A variant of the algorithm could be considered to analyse the structure. Since Σ is known, the cable tensions that the structure is able to support can be computed by inverting Eq. (6): $\tau^I = \mathbf{V}^I \Sigma$ with $\mathbf{V} = \mathbf{T}^{-1}$. It seems then possible to check the cable

tensions τ needed to counterbalance the applied wrench by comparing the values of τ to this interval τ^I. This may induce into errors since τ^I is overestimated.

The proposed algorithm includes a procedure to evaluate the distances between the structure and other elements of the mechanism. It is well known that such distance evaluation is generally speaking a rather complex problem. In the following examples, only structure-end-effector interferences are evaluated in the context of planar mechanisms.

5 Examples

In the following, two examples are given to illustrate the method. The implementation is performed using MATLAB software with the INTLAB [15] interval arithmetic toolbox. On a Sony laptop (Intel Core^{TM} Duo Processor T8300, 2.4 GHz), the computation time of the algorithm for the given examples is about 15 minutes.

5.1 A 2-DOF Planar Robot with a 3-Bar Truss

Let us consider a 2-DOF planar robot with a 3-bar truss (Fig. 1). The position of the points A_i, $i \in [1,3]$ is defined by the radius R. The platform is a disk of center P. The bars are cylindrical elements of diameter D chosen identical to lighten the analysis. We consider as an input the desired workspace $\mathscr{W}$, the applied wrench to the platform $\mathbf{f}^I$, the admissible cable tensions $\mathscr{T}$ and the parameters defining the truss (Table 1).

With the first set of parameters, a portion of the workspace is not valid due to the buckling of the structure (Fig. 3).

With a higher diameter of the bars (Set #2, Table 1, Fig. 4), no failure of the structure occurs. We can only observe small limitations on the sides of the workspace due to the maximum tension of the cables. Further analysis shows however that the sensor measurement range $\varepsilon_0 = 6.10^{-3}$ is much larger than the maximum encountered deformation $\varepsilon = 3.10^{-3}$. In other words, the structure is not well adapted to the sensor range, that depends on manufacturer specifications. It is indeed logical to not be able to get simultaneously a satisfying deformation range and the absence of buckling: we have only one parameter to tune, and two constraints.

In the third set of parameters (Set #3, Table 1, Fig. 5), the bar diameter D and the mechanism radius R are modified. It allows to obtain a correct deformation range for the measurement, with a maximum deformation equal to $5.4.10^{-3}$. No buckling occurs, so the structure seems adequate for the application. However, the modification of the radius R induces higher tensions in the cables, and as can be seen on Fig. 5-A, the admissible workspace is reduced with respect to the second case. This shows that it is very important to consider in the same time the structural and kinematic analysis: without modification of the radius R, it is not possible to obtain

Table 1 First example – Mechanism parameters.

	$\mathscr{W}$ (mm)	$\mathscr{T}$ (N)	$\mathbf{f}^I$ (N)	E (GPa)	ε_0	D	R (mm)
Set #1	$\begin{bmatrix}[-12.5,12.5]\\[-12.5,12.5]\end{bmatrix}$	[1,100]	$\begin{bmatrix}[-10,10]\\[-10,10]\end{bmatrix}$	3	6.10^{-3}	3	27.5
Set #2	$\begin{bmatrix}[-12.5,12.5]\\[-12.5,12.5]\end{bmatrix}$	[1,100]	$\begin{bmatrix}[-10,10]\\[-10,10]\end{bmatrix}$	3	6.10^{-3}	3.8	27.5
Set #3	$\begin{bmatrix}[-12.5,12.5]\\[-12.5,12.5]\end{bmatrix}$	[1,100]	$\begin{bmatrix}[-10,10]\\[-10,10]\end{bmatrix}$	3	6.10^{-3}	4.3	26
Set #4	$\begin{bmatrix}[-12.5,12.5]\\[-12.5,12.5]\end{bmatrix}$	[1,230]	$\begin{bmatrix}[-10,10]\\[-10,10]\end{bmatrix}$	3	6.10^{-3}	4.3	26

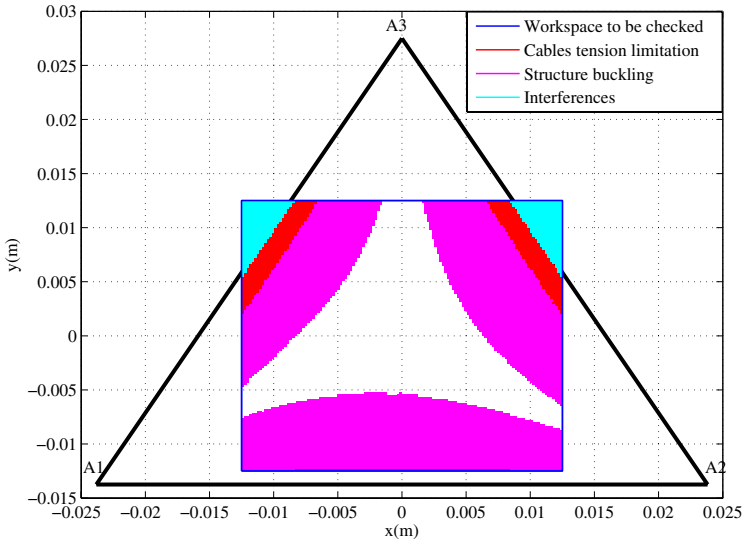


Fig. 3 First example – Workspace analysis with the first set of parameters.

an adequate deformation range, and modifying the value of R can lead to excessive tensions of the cables. Figure 5-B shows that if the admissible tensions are modified (Set #4 in Table 1), the workspace is only limited by the interferences with the structure.

5.2 A 2-DOF Planar Robot with a 9-Bar Truss

In this example, we still consider a 2-DOF planar robot but with a 9-bar truss as illustrated in Fig. 6(a). Such a structure can be of great interest because, contrary

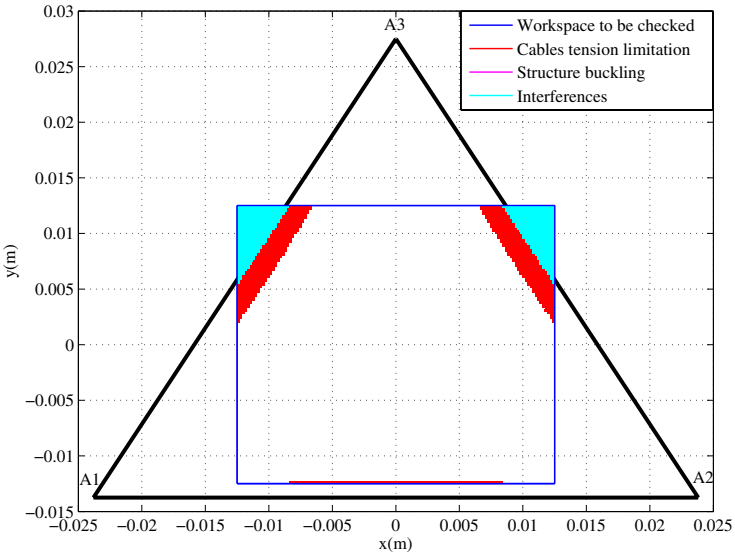


Fig. 4 First example – Workspace analysis with the second set of parameters.

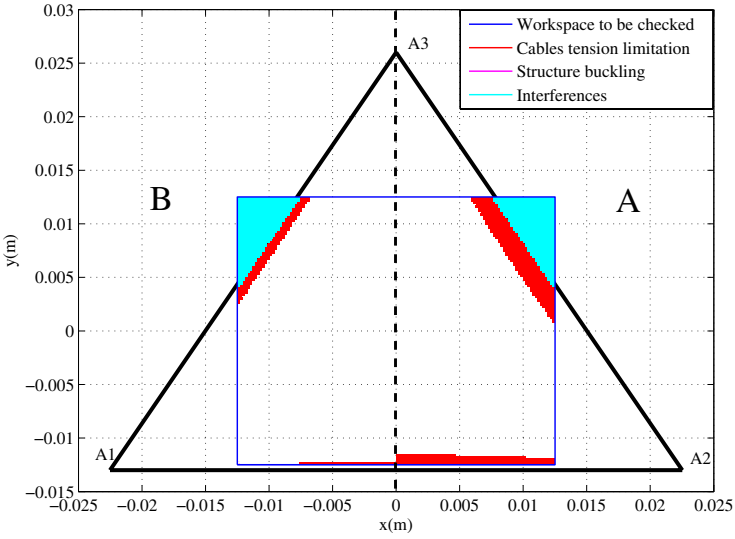
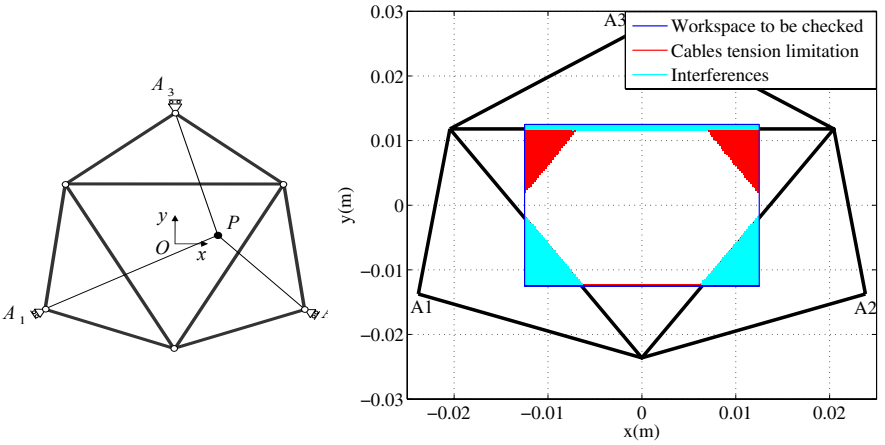


Fig. 5 First example – Workspace analysis with the third and fourth sets of parameters.

to the previous structure, some bars are submitted to tension and not compression: we can therefore use bars in tension to perform measurements, and reinforce bars in compression to avoid buckling. We here focus on the importance of the evaluation of interferences with the structure. The wires are considered to be in a plane above



(a) (b)

Fig. 6 (a) Second example – 2-DOF planar robot with 9-Bar truss, (b) Second example – Workspace analysis.

or under the structure, and interferences between the end-effector and the structure are determined.

The evaluation of the WFW only needs the position of the points A_i, $i \in [1,3]$, and is therefore independent from the number of bars in the truss. We can observe here that we need to take into account the exact geometry of the truss during the workspace evaluation, and thus to consider simultaneously structural and kinematic analysis: In Fig. 6 (b) the workspace is limited on one side by the cable tensions but on the other side only by the interferences between the truss and the end-effector. Obviously, the selection of the truss has an influence on the mechanism analysis.

6 Conclusions

In this article, we have proposed to consider simultaneously structural and kinematic analysis for WDMs, in the context of devices with instrumented truss structure. We have shown that constraints associated to structural analysis can be treated using interval analysis, and introduced an algorithm. Using two examples, we have shown that it is indeed important to perform in the same time structural and kinematic analysis. The next step will now to develop a design methodology using this algorithm to select and optimize a device for the considered application. The optimization procedure will visibly be time consuming, particulary if a large number of parameters are to be taken into account. A more efficient implementation, using for instance the BIAS/PROFIL C++ library [16], will therefore be considered.

References

1. J. Kaufman and M. Lee, *Vascular and Interventional Radiology – The Requesites*, Mosby Ed. (2004).
2. M. Muntener, A. Patriciu, D. Petrisor, M. Schär, D. Ursu, D. Y. Song and D. Stoianovici, Transperineal prostate intervention: Robot for fully automated MR imaging-system description and proof of principle in a canine model, *Radiology*, **247**(2) (2008).
3. A. Melzer, B. Gutmann, T. Remmele, R. Wolf, A. Lukoscheck, M. Bock, H. Bardenheuer and H. Fischer, Innomotion for percutaneous image-guided interventions, *IEEE, Eng. Med. Bio. Mag.*, **27**(3), 66–73 (2008).
4. P. Lafourcade, Etude des manipulateurs paralleles cables, conception d'une suspension active pour soufflerie, Thesis, National Higher School of Aeronautics and Space, France (2004).
5. www.fiso.com.
6. J. P. Merlet, Analysis of the influence of wires interference on the workspace of wire robots. In *ARK*, Sestri-Levante, June 28–July 1, pp. 211–218 (2004).
7. J. P. Merlet and D. Daney, Legs interference checking of parallel robots over a given workspace or trajectory. In *ARK*, Orlando, Florida, May 15–19, pp. 757–762 (2006).
8. D. Chablat and P. Wenger, Moveability and collision analysis for fully-parallel manipulators. In *12th RoManSy*, Paris, July 6–9, pp. 61–68, 1998.
9. P. Bosscher and I. Ebert-Uphoff, Wrench-based analysis of cable-driven robots. In *IEEE Int. Conf. on Robotics and Automation*, New Orleans, pp. 4950–4955 (2004).
10. G. Barrette and C. M. Gosselin, Determination of the dynamic workspace of cable-driven planar parallel mechanisms, *Journal of Mechanical Design*, **127**(2), 242–248 (2005).
11. M. Gouttefarde, J. P. Merlet and D. Daney, Wrench-feasible workspace of parallel cable-driven mechanism. In *IEEE Int. Conf. on Robotics and Automation*, Rome, pp. 1492–1497, (2007).
12. S. Bouchard, C. M. Gosselin and B. Moore, On the ability of a cable-driven robot to generate a prescribed set of wrenches. In *Proceedings of International Design Engineering Technical Conference and Computers and Information in Engineering Conference*, Brooklyn, New York, August 3–6 (2008).
13. D. Chablat, P. Wenger, F. Majou and J. P. Merlet, An interval analysis based study for the design and the comparison of three-degrees-of-freedom parallel kinematic machines, *the International Journal of Robotics Research*, **23**(6), 615–624 (2004).
14. R. Verhoeven, M. Hiller and S. Tadokoro, Workspace, stiffness, singularities and classification of tendon-driven Stewart platforms. In *ARK*, Strobl, June 29–July 4, pp. 105–114 (1998).
15. http://www.ti3.tu-harburg.de/ rump/intlab/
16. http://www.ti3.tu-harburg.de/Software/PROFILEnglisch.html

Multiple-Point Kinematic Control of a Humanoid Robot

Agostino De Santis[1], Giuseppe Di Gironimo[2], Luigi Pelliccia[2], Bruno Siciliano[1] and Andrea Tarallo[2]

[1]*PRISMA Lab, Dipartimento di Informatica e Sistemistica;* [2]*VRTEST Lab, Dipartimento di Meccanica ed Energetica, Università degli Studi di Napoli Federico II, Italy; e-mail: siciliano@unina.it*

Abstract. Robots designed to operate in everyday domains have to move in environments designed for the humans. Therefore, they will often have a humanoid kinematic structure. Simple and efficient kinematic models are needed for motion control of this class of robots. An algorithm is presented to solve the inverse kinematics problem in the presence of a number of control points arbitrarily located on the whole robot body, using an augmented Jacobian approach and including posture control. Simulation experiments are reported, showing the effectiveness of the proposed approach.

Key words: Multiple-point control, inverse kinematics, humanoid robot.

1 Introduction

Modelling and control issues for a humanoid robot with a high degree of redundancy are addressed in this paper, considering a multiple-control point inverse kinematics algorithm. The increasing attention of the robotics community towards *humanoid robotics* [1, 2, 3] is not simply related to the ancestral ambition of *building something that looks like a human*, but has also an immediate and objective reason. In particular, it arises from the apparently obvious fact that for all actions performed daily by humans, the objects that they manipulate and the environment where they live have been built or structured "on a human scale". For instance, all the objects that we manipulate have been conceived based on the shape of our hands.

If we really want to build machines able to cooperate with human beings, we need to design robots that not only can move through environments designed for humans, but can also handle objects particularly suited to our physical structure and our behavior. For instance, bipedal robots could potentially move in the same space where people work, such as an industrial plant with stairs and handrails specifically designed for human use. In this way, these robots could cooperate with humans and even collaborate with one another using already working ordinary tools or machinery. Further considerations can be made even under the aspect of human-robot and robot-robot communication channels [4]; for instance humanoid robots could

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_17,

even be used in the therapy of some forms of mental disorders [5]. It is therefore necessary to develop efficient models that allow us to accurately control the motion of humanoid robots. On the other hand, it is also necessary to develop simulation tools to study robots behavior in *unstructured* environments, considering the safety issues arising from the interaction with humans.

The model of humanoid robot that will be described in the following sections is also suitable for simulating the behavior of human beings in a virtual environment. This can be very useful for ergonomics analyses or even for manual processes simulations.

Manufacturing companies, indeed, have now taken the concept of "man adaptability" as a basic parameter of quality for their products, and thus they are giving an increasing attention to ergonomic analyses, even from the early stages of design [6, 7, 8, 9]. The so-called *virtual manikins*, provided by many process simulation software, are essentially virtual kinematic chains consisting of several segments and joints. The length of their segments are derived from anthropometric databases, which can be queried with respect to different percentiles in the population. However, these software tools are generally too much complicated to be handled and the so-called "process simulation" often becomes a very time-consuming task, mainly because of the difficulty in controlling the kinematic chain of the virtual humanoid.

Therefore, developing efficient algorithms aimed at controlling a high-articulated chain, such as the human mechanical structure, is very interesting even for fields apparently unrelated to robotics.

2 Kinematic Modelling

The basic idea in the described approach is to control the highly redundant kinematic structure of a humanoid robot, by means of only few control points which can move on the structure [10, 11, 12]. If one considers a serial manipulator and its direct kinematics equation, changing the value of its Denavit–Hartenberg (DH) parameters results in the kinematics equations of another manipulator, whose end-effector is located before the real one: that is equivalent to moving the control point of the structure. If the DH values are described in a symbolic form, they are such to identify an arbitrary point as a virtual end-effector of a smaller manipulator considered for the control. An arbitrary number of such control points can be considered. It is then possible to consider these control points as fixed or moving. Related to a humanoid, different kinematic chains will be considered. In addition, during the carrying of a certain task, the postures taken by our kinematic structure largely depend on balancing and mechanics issues, that are only partially related to the considered task. This implies the need for an additional posture control.

For this purpose, the goal has been to develop an inverse kinematics algorithm that allows concentrating only on a limited number of task-related control points, without the need of specifying the Degrees of Freedom (DoFs) of the chain for posture control. The position of the Center of Mass (CoM) of the humanoid has then

been taken into account: it is calculated on-line and always kept consistent with the balancing issues of the mechanical structure, by identifying the time-varying CoM as an additional moving control point.

It is worth noticing that the selected control points on the humanoid can be also selected automatically depending on the task and the environment, giving a very powerful tool for simulation.

2.1 Hierarchical Model of Humanoid Robot

Firstly, in order to take advantage of the systematic approaches typical of serial robots, the humanoid has been modeled as the combination of four kinematic chains, which share the same starting point, called *root*. The resulting model is the hierarchical structure shown in Fig. 1. Starting from this graph, it is possible to build up the DH model of the whole kinematic chain (Fig. 2).

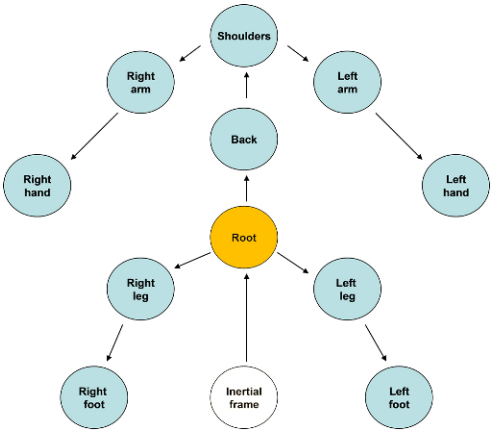


Fig. 1 Hierarchical model of humanoid robot.

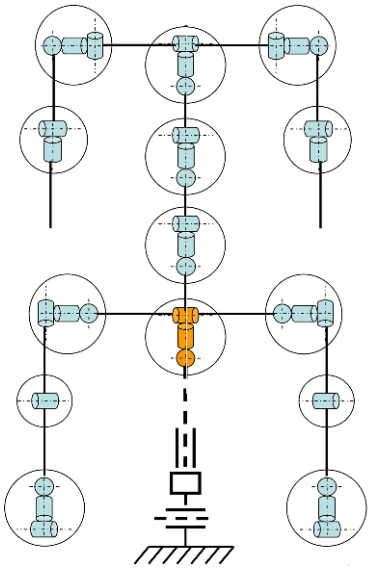


Fig. 2 DH model of humanoid robot.

In particular, we can define a number of direct kinematics equations, with respect to the *root* reference frame. One or more control points on the provided chains can be selected, by considering the proper set of DH parameters that specify such points.

As we can see, the position and orientation of the *root* node with respect to the reference frame is specified by introducing 6 *virtual joints* (see Section 2.2). Thus, the considered kinematic structure has 39 DoFs in all.

This kind of modeling has the advantage of simplicity, but generally it may cause a physical consistency problem, since some links (as well as all the virtual links) are shared among different kinematic chains. For instance the "back" of the virtual humanoid is shared between its right and left arms. This issue and its solution are discussed in Section 2.5.

2.2 Virtual Joints

Now, we must describe the position and orientation of the multi-legged kinematic chain with respect to an inertial frame. For industrial robots identifying such a frame is intuitive, because they have a fixed base. A humanoid robot, instead, is bound to the ground by a one-way constraint, that is the current support plane, for instance one foot.

However, this reference periodically changes during the walk, thus we apparently cannot identify a fixed base starting from which the DH method can be applied (Fig. 3).

Moreover, the presence of multiple end-effectors (two hands and two feet) implies the need to describe the position and orientation of many frames, differently from industrial robots, in which the kinematic chain has only one end-effector. This problem has been overcome using the *virtual joints* approach [13]. Namely, the humanoid robot has been conceived as connected to the ground plane through a virtual manipulator consisting of three prismatic and three revolute joints, which characterize its position and orientation. The attaching point has been called *root* (Fig. 4).

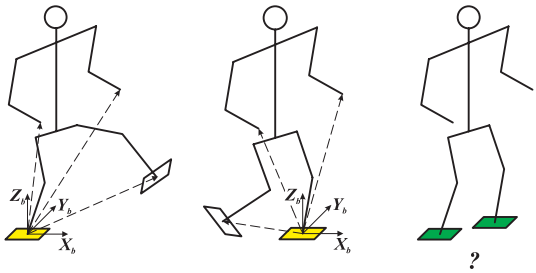


Fig. 3 Which reference frame?

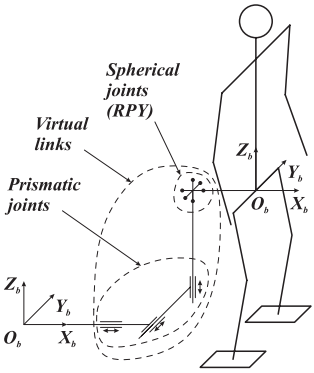


Fig. 4 Virtual joints approach.

With this approach, the hands and the feet (and even any other control point) are simply end-effectors that can be controlled with velocity references. In other words, the posture of the virtual humanoid is completely specified by the following parameters vector:

$$\mathbf{q} = \left[\mathbf{q_r}^T \;\; q_1 \; q_2 \; .. \; q_n\right]^T$$

where $\mathbf{q_r} = [\mathbf{p_r^{oT}} \; \boldsymbol{\omega}_\mathbf{r}^{\mathbf{oT}}]^\mathbf{T}$ identifies the *root* frame.

Moreover, virtual joints technique makes unnecessary the management of closed kinematic chains during the phase of double support. Indeed, this condition becomes merely equivalent, from a kinematic point of view, to imposing a null velocity reference to the feet.

2.3 Augmented Jacobian

Each chain has its own direct kinematic function, therefore a Jacobian matrix can be computed for a generic control point of the structure. Generally, considering n control points we can define the following set of equations:

$$\begin{aligned} \mathbf{v_1} &= \mathbf{J_1}\dot{\mathbf{q}} \\ \mathbf{v_2} &= \mathbf{J_2}\dot{\mathbf{q}} \\ &\vdots \\ \mathbf{v_n} &= \mathbf{J_n}\dot{\mathbf{q}} \end{aligned} \tag{1}$$

where the generic element $\mathbf{J_i}$ is the Jacobian matrix related to a specific control point.

It is understood that, if the generic joint variable q_j does not affect $\mathbf{v_n}$, it is $(J_n)_{ij} = 0$. This set of equations can be summarized as

$$\mathbf{v} \;=\; \mathbf{J_{AU}}\,\dot{\mathbf{q}} \tag{2}$$

where $\mathbf{J_{AU}}$ is the so-called *Augmented Jacobian*. On one hand, this approach allows us to solve the inverse kinematic problem with only one Closed-Loop Inverse Kinematics (CLIK) algorithm [14]. On the other hand, the trajectories defined for the control points will be all treated as *primary tasks*, unlike other solution methods do, such as null-space based approaches [15, 16].

In particular, in order to define the structure of $\mathbf{J_{AU}}$, the vector $\dot{\mathbf{q}}$ must be properly sorted. Since the humanoid structure is composed by four kinematic chains, we can write four different vectors of unknowns:

$$\begin{aligned} \dot{\mathbf{q}}_1 &= \left[\dot{\mathbf{q}}_r^T \;\; \dot{\mathbf{q}}_{rl}^T\right]^T && \textit{right leg} \\ \dot{\mathbf{q}}_2 &= \left[\dot{\mathbf{q}}_r^T \;\; \dot{\mathbf{q}}_{ll}^T\right]^T && \textit{left leg} \\ 4pt]\dot{\mathbf{q}}_3 &= \left[\dot{\mathbf{q}}_r^T \;\; \dot{\mathbf{q}}_b^T \;\; \dot{\mathbf{q}}_{ra}^T\right]^T && \textit{right arm} \\ 4pt]\dot{\mathbf{q}}_4 &= \left[\dot{\mathbf{q}}_r^T \;\; \dot{\mathbf{q}}_b^T \;\; \dot{\mathbf{q}}_{la}^T\right]^T && \textit{left arm} \end{aligned}$$

where $\dot{\mathbf{q}}_r$ are the velocities of the virtual joints that are shared among four kinematic chains. These vectors can be summarized in only one vector of unknowns

$$\begin{aligned}\dot{\mathbf{q}} &= \begin{bmatrix}\dot{\mathbf{q}}_r^T & \dot{\mathbf{q}}_{rl}^T & \dot{\mathbf{q}}_{ll}^T & \dot{\mathbf{q}}_b^T & \dot{\mathbf{q}}_{ra}^T & \dot{\mathbf{q}}_{la}^T\end{bmatrix}^T \\ &= \begin{bmatrix}\dot{q}_1 & \dot{q}_2 & .. & \dot{q}_{39}\end{bmatrix}^T.\end{aligned} \tag{3}$$

With this choice, the Augmented Jacobian takes on the following form

$$\mathbf{J_{AU}} = \begin{bmatrix} \mathbf{J_{r_i}} & \mathbf{J_{rl_i}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{J_{r_i}} & \mathbf{0} & \mathbf{J_{ll_i}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{J_{r_i}} & \mathbf{0} & \mathbf{0} & \mathbf{J_{b_i}} & \mathbf{0} & \mathbf{0} \\ \mathbf{J_{r_i}} & \mathbf{0} & \mathbf{0} & \mathbf{J_{b_i}} & \mathbf{J_{ra_i}} & \mathbf{0} \\ \mathbf{J_{r_i}} & \mathbf{0} & \mathbf{0} & \mathbf{J_{b_i}} & \mathbf{0} & \mathbf{J_{la_i}} \end{bmatrix}. \tag{4}$$

The matrix $\mathbf{J_{AU}}$, with the proposed humanoid model, has 39 columns, while the number of its rows depends on the number of control points considered.

2.4 Center-of-Mass Jacobian

Unlike industrial manipulators and, more generally, non-ambulatory robots, bipedal robots must concern about their balance while performing any task. If this does not happen, obviously, the robot would lean over and fall. Moreover, humanoid robots are inherently hyper-redundant, having a much higher number of joints than traditional industrial robots. Consequently, there are many postures that achieve the same position for its body terminals, corresponding to control points. Also, taking into account the balancing issues allows the humanoid to attain more natural posture, similar to those of human beings.

For this, the Virtual End-Effectors (VEEs) technique [10] has been implemented also with respect to the center of mass (CoM) of the digital humanoid, which becomes a further control point for the kinematic chain. In particular, the trajectory of the CoM can be defined in such a way that its vertical projection on the current support plane (namely, the Center of Pressure, CoP) belongs to the stability polygon formed by the feet (Fig. 5). It is worth noticing that the constraint about the CoP will be treated as a primary task, as well as the other tasks.

The basic idea is to obtain a differential relationship like

$$\mathbf{v_G} = \mathbf{J_G}\dot{\mathbf{q}}, \tag{5}$$

where $\mathbf{J_G}$ is a $3 \times n$ matrix, called Center-of-Mass Jacobian. Then, Eq. (5) will be inserted in Eq. (1) as a further control point.

For this purpose, we can define the CoM of a kinematic chain composed of n links as

$$\mathbf{p}_G = \frac{\sum_{i=1}^{n} m_i \mathbf{p}_{G_i}}{\sum_{i=1}^{n} m_i} = \frac{1}{m} \sum_{i=1}^{n} m_i \mathbf{p}_{G_i}. \tag{6}$$

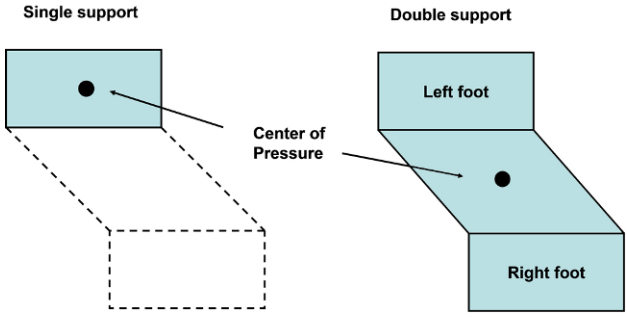


Fig. 5 Center of Pressure and support plane.

Equation (6) can be derived with respect to time

$$\mathbf{v_G} = \frac{1}{m} \sum_{i=1}^{n} m_i \mathbf{v}_{G_i}. \tag{7}$$

Since the center of mass of each link can be considered as a *Virtual End-Effector* (VEE), it is always possible to write the differential relationship

$$\mathbf{v_{G_i}} = \mathbf{J_{G_i}} \dot{\mathbf{q}}$$

where

$$\mathbf{J_{G_i}} = \begin{bmatrix} \gamma_{x,1} & \cdots & \gamma_{x,i} & 0 & \cdots & 0 \\ \gamma_{y,1} & \cdots & \gamma_{y,i} & 0 & \cdots & 0 \\ \gamma_{z,1} & \cdots & \gamma_{z,i} & 0 & \cdots & 0 \end{bmatrix}. \tag{8}$$

Indeed, if the vector $\dot{\mathbf{q}}$ has been properly sorted, $\mathbf{v_{G_i}}$ can be affected at most by the first i links of the chain. Now, Equation (7) can be written as

$$\mathbf{v_G} = \left[\frac{1}{m} \sum_{i=1}^{n} m_i \mathbf{J_{G_i}} \right] \dot{\mathbf{q}}. \tag{9}$$

By comparing Eqs. (5) and (9), we can finally assume

$$\mathbf{J_G} = \frac{1}{m} \sum_{i=1}^{n} m_i \mathbf{J_{G_i}}. \tag{10}$$

Given $\mathbf{J_G}$, the velocity of CoM $\mathbf{v_G}$ becomes a further control point for the kinematic chain. Thus, we can insert the kinematic relation (5) in the equations set (1). As a result, we will have an Augmented Jacobian matrix with two more rows, that are related to the components of $\mathbf{v_G}$ projected on the current support plane. As mentioned above, the implemented inversion algorithm assures that a constraint on CoM velocity becomes a high-priority task to be achieved, without using null-space projection.

Finally, it is worth emphasizing that the expression of $\mathbf{J_G}$ suggests also the possibility to use the *kinetostatic duality* [14] to compute the joint torques corresponding to forces applied to the structure.

2.5 Conflicting Tasks

As mentioned above, some tracts of the humanoid structure are shared among apparently different kinematics chains. For instance, the right and left arms of the humanoid share a common tract, namely the back. But, if actually the left and right arms were modelled as *independent* chains, they could perform different or even conflicting tasks.

For this, inverse kinematics algorithms for multi-legged robots generally provide two different solutions for the left and right arm. In particular, for the back it will be

$$\dot{\mathbf{q}}_{\mathbf{bl}} \neq \dot{\mathbf{q}}_{\mathbf{br}}, \tag{11}$$

where $\dot{\mathbf{q}}_{br}$ and $\dot{\mathbf{q}}_{bl}$ are different solutions obtained considering the back belonging respectively to the right and to the left arm. However, generally this issue is commonly solved with the following choice for the joints of the back:

$$\dot{\mathbf{q}}_{\mathbf{b}} = \frac{\mathbf{1}}{\mathbf{2}} (\dot{\mathbf{q}}_{\mathbf{br}} + \dot{\mathbf{q}}_{\mathbf{br}}). \tag{12}$$

This guarantees a physical consistent solution, but in general none of the conflicting tasks will be actually achieved.

The CLIK algorithm based on the Augmented Jacobian cleverly resolves also this issue. Indeed, the vector of solution $\dot{\mathbf{q}}$ has been sorted in such a way that its elements appear just once, thus the inversion algorithm provides only one solution that is consistent with all the physical constraints.

On the other hand, the main problem related to the application of the Augmented Jacobian method is the matrix inversion, due to its dimensions ($\mathbf{J_{AU}}$ has 39 columns) and consequently to the detection of its singularities.

3 Simulations in VR

In order to test the proposed inversion model, several VR simulations have been carried out. First of all, a geometric model for the virtual humanoid has been built up with a typical hierarchical approach. The result has been a VRML model used for the simulations.

After that, the kinematic model of the virtual humanoid and its CLIK algorithm have been implemented with *MathWorks MATLAB–Simulink* software. Since a weighted pseudo-inverse has been adopted to compute the inverse kinematics, a

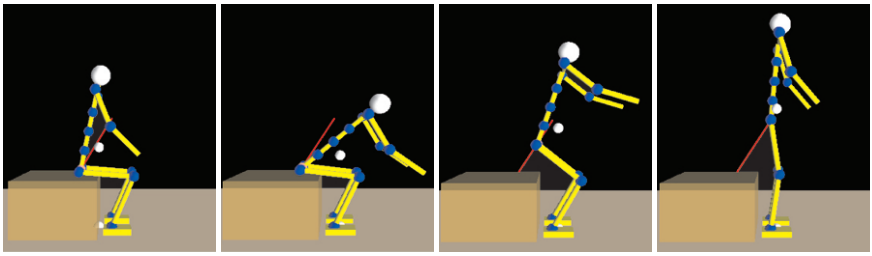


Fig. 6 Standing up from a sitting position.

proper choice of weights and of some optimization criteria have granted quite natural and fluid movements for the virtual humanoid. As a result, despite the ease of planning the movements of the digital humanoid, we can simulate quite complex tasks, by planning the trajectory for only a limited number of control points. For instance, the virtual humanoid can walk or even climb a ladder, as will be shown in the following sections.

3.1 Standing up from a Sitting Position

In Fig. 6 different frames of a standing-up simulation are shown. This task has been achieved just by imposing a null velocity to the feet of the virtual humanoid and by giving a vertical velocity reference to its pelvis. As a further constraint, the CoP must always belong to the support plane (*balance control*). As shown, the virtual humanoid performs the assigned movement always keeping itself in balance.

In a similar way, it is possible to simulate the virtual humanoid sitting down from a standing position.

3.2 Collision Avoidance

The approach can be used to take into account also possible obstacles in the humanoid workspace. Figure 7 shows again the simulation of a standing up, but this time there is a table. This task has been achieved by assigning to the control points velocity references coming from repulsive potential fields.

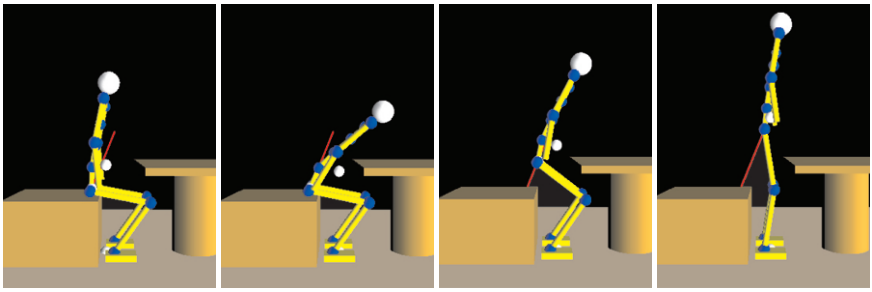


Fig. 7 Standing up from a sitting position near a table.

3.3 More Complex Tasks

In this section the results of the simulation of quite complex tasks are reported. The CLIK algorithm has always taken into account the constraints about the CoP, as mentioned above.

The results shown in figures can be quite interesting in the field of both computer graphics and robotics.

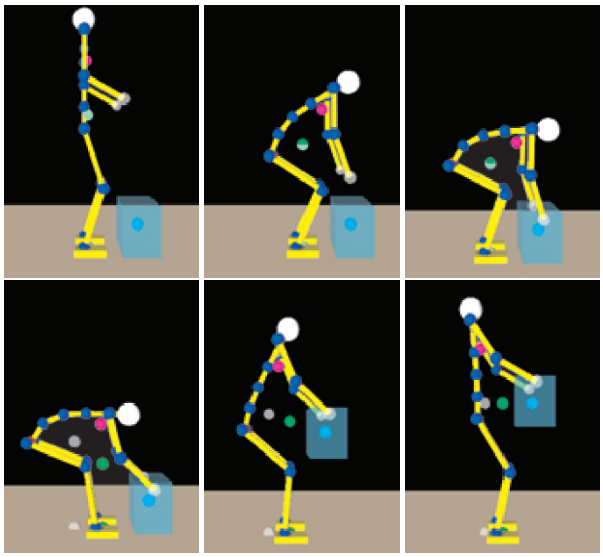


Fig. 8 The humanoid lifts a weight.

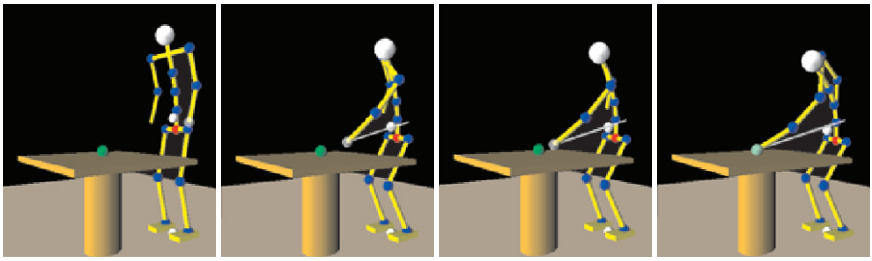


Fig. 9 The virtual humanoid grabbing an object on a table.

4 Conclusions and Future Work

The main contribution of the proposed approach is in the computation of an Augmented Jacobian matrix to specify trajectories for different control points, including the control of the center of mass of the kinematic structure. In fact, the definition of the movements of the center of pressure as a primary task has granted natural movements to the virtual humanoid, in spite of the limited number of considered control points. Notice that these points can move on specified sections of the humanoid, giving the possibility of controlling nominally every point on the kinematic structure. The symbolic implementation leads to a very fast response of the algorithm with respect to complex simulations in the humanoid configuration space.

Moreover, the developed model lends itself to a very different set of applications, even not strictly robotic. Firstly, it can be used for digital animation of virtual humanoids in the field of ergonomics and process analysis. In fact, despite complexity and cost of already existing software tools dedicated to this type of analysis, generally their simulation algorithms are still tied to "key-frame" animation techniques. The developed model instead makes it possible to observe the resulting joint motion simply by planning the trajectory of a limited number of control points.

Another field of application could be marker-based motion capture [17], where the algorithm can be used in order to limit the number of markers needed to capture human movements.

Finally, although the described model is advanced in terms of quality of analysis, it is also computationally efficient. Specifically, a symbolic representation for the kinematics of the digital humanoid has been derived. In this way, it is possible to change in real-time several characteristic parameters of the chain, such as the applied loads, without further computational overload.

Acknowledgements The research leading to these results has been partially supported by the DEXMART Large-scale integrating project, which has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement ICT-216239. The authors are solely responsible for its content. It does not represent the opinion of the European Community and the Community is not responsible for any use that might be made of the information contained therein.

References

1. Sakagami, Y., Watanabe, R., Aoyama, C., Matsunaga, S., Higaki, N., Fujimura, K.: The intelligent ASIMO: system overview and integration. In *Proceedings 2002 IEEE/RSJ International Conference on Intelligent Robots and System*, Lausanne, Switzerland (2002).
2. Kaneko, K., Kanehiro, F., Kajita, S., Yokoyama, K., Akachi, K., Kawasaki,T., Ota, S., Isozumi, T.: Design of prototype humanoid robotics platform for HRP. In *Proceedings 2002 IEEE/RSJ International Conference on Intelligent Robots and System*, Lausanne, Switzerland (2002).
3. Iwata, H., Sugano, S.: Design of human symbiotic robot TWENDY-ONE. In *Proceedings 2009 IEEE International Conference on Robotics and Automation*, Kobe, Japan (2009).
4. Siciliano, B., Khatib, O. (Eds.): *Springer Handbook of Robotics*. Springer, Heidelberg (2008).
5. Robins, B., Dautenhahn, K., Boekhorst, T., Billard, A.: Robotic assistants in therapy and education of children with autism: Can a small humanoid robot help encourage social interaction skills? *Univers. Access Inf. Soc.*, **4**(2), 105–120 (2005).
6. Di Gironimo, G., Lanzotti, A.: Designing in VR. *International Journal on Interactive Design and Manufacturing*, **3**(2), 51–53 (2009).
7. Caputo, F., Di Gironimo, G., Marzano, A.: Ergonomic optimization of a manufacturing system work cell in a virtual environment. *Acta Polytechnica*, **46**(5), 21–27 (2006).
8. Caputo, F., Di Gironimo, G., Papa, S.: A Virtual reality system for ergonomics and usability validation of equipment controls. *Anales de Ingegnería Gráfica*, **18**, 47–64 (2006).
9. Di Gironimo, G., Patalano, S.: Re-design of a railway locomotive in virtual environment for ergonomic requirements. *International Journal on Interactive Design and Manufacturing*, **2**(1), 47–57 (2008).
10. De Santis, A., Pierro, P., Siciliano B.: The virtual end-effectors approach for human-robot interaction. In *Proceedings 10th International Symposium on Advances in Robot Kinematics*, Ljubljana, Slovenia (2006).
11. De Santis, A., Albu-Schaeffer, A., Ott, C., Siciliano, B., Hirzinger, G.: The skeleton algorithm for self-collision avoidance of a humanoid manipulators. In *Proceedings 2007 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Zürich, Switzerland (2007).
12. De Santis, A., Siciliano B.: Inverse kinematics of robot manipulators with multiple moving control points. In *Proceedings 11th International Symposium on Advances in Robot Kinematics*, Batz-sur-Mer, France (2008).
13. Yamane, K.: *Simulating and Generating Motions of Human Figures*. Springer, Heidelberg (2004).
14. Siciliano, B., Sciavicco, L., Villani, L, Oriolo, G.: *Robotics: Modelling, Planning and Control*. Springer, London (2009).
15. Nakamura, Y.: *Advanced Robotics: Redundancy and Optimization*. Addison-Wesley, Reading, MA (1991).
16. Siciliano, B., Slotine, J.-J. E.: A general framework for managing multiple tasks in highly redundant robotic systems. In *Proceedings 5th International Conference on Advanced Robotics*, Pisa, Italy (1991).
17. Magnenat-Thalmann, N., Thalmann, D.: *Modelling and Motion Capture Techniques for Virtual Environments*. Springer Verlag (1998).

Optimum Design of a Pan-Tilt Drive for Parallel Robots

Danial Alizadeh[1], Jorge Angeles[1] and Scott Nokleby[2]

[1]*Department of Mechanical Engineering and Centre for Intelligent Machines, McGill University, Montreal, Canada; e-mail: {alizadeh, angeles}@cim.mcgill.ca*
[2]*Faculty of Engineering and Applied Science, University of Ontario Institute of Technology, Oshawa, Canada; e-mail: scott.nokleby@uoit.ca*

Abstract. The need to drive a serial array of two robotic axes by means of two motors fixedly mounted on a common base is the motivation behind the work reported here. An innovative two-degree-of-freedom drive is proposed, which is based on a serial array of an epicyclic gear train and a five-bar linkage. The ring and sun gears are driven by the two motors, its two outputs being the angular velocity of the planet-carrier and the relative angular velocity of the planet gears w.r.t. their carrier. The former is the pan rate, the latter the input to the five-bar linkage whose output is the tilt rate. Optimization is used to determine the proportions of the various dimensions involved so as to obtain a 2×2 Jacobian, mapping the motor rates into the pan and tilt rates, whose condition number, at a prescribed drive posture, is a minimum. Moreover, to improve the force-torque transmission characteristics, the transmission defect of the mechanism, as introduced elsewhere, is also minimized. To this end, a sequential-quadratic programming method using an orthogonal decomposition of the space of design variables is implemented; the dimension ratios thus resulting lead to a mechanism whose transmission defect is a minimum, while the condition number of its Jacobian matrix is a minimum as well, at the posture where the transmission angle is 90°.

Key words: Pan-tilt drive, condition number, transmission defect.

1 Introduction

Most commonly, parallel robots are designed with as many limbs as degrees-of-freedom (dof) [1]. While this layout eases the design, it does so at the expense of the robot footprint, which becomes correspondingly large because of the number of limbs, and of the manipulability. We understand the latter as the capability to accurately manipulate the end-link of a serial chain, the mobile platform (MP), from an actuator mounted on the base.

Parallel robots with a limited number of limbs may offer advantages over their many-limb counterparts. For example, a six-dof parallel robot with only three limbs,

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_18,

should offer a larger *dextrous workspace* – the region of the workspace on which its MP can attain a given range of orientations[1] – than its six-limb counterpart.

The motivation behind the work reported here is the ongoing work aimed at the development of a two-limb Schönflies-motion generator (SMG) [3] at McGill University, in collaboration with the University of Ontario Institute of Technology. In a nutshell, a SMG is a four-dof parallel robot capable of motions proper of SCARA (Selective Compliance Assembly Robot Arm) systems: three independent translations and one rotation about an axis of fixed orientation, usually vertical. Currently, only one parallel SMG, Adept's Quattro, is out in the market. Other parallel robots with four dof for Schönflies-motion generation feature a *hybrid architecture*, in that they entail a fully parallel array of three identical limbs for three-dof translations, in series with a fourth axis for rotations about a vertical axis, e.g., ABB Robotics' Flexpicker and Fanuc's M1-*i*A. The footprint of these robots could benefit from a number of limbs reduced to the lowest possible, namely, two. Each limb of the McGill SMG undergoes two rotations, one about a vertical axis, the other about a horizontal axis. The first rotation is referred to as the pan, the second being called the tilt. Pan and tilt motions are usually produced via a serial array of two orthogonal revolutes; however, higher accuracy and stiffness can be achieved by exploiting parallel pan-tilt motion generators, a number of them being reported in [4], and then ranked in terms of design complexity [5, 6]. The design complexity of a mechanism is understood as a measure of diversity of its components serving as a selection guide in the conceptual design phase, where no parametric model is available. In this paper, the winning mechanism in [4] with the least design complexity is optimally dimensioned.

The proposed mechanism is a two-dof drive for the production of pan and tilt by means of two motors mounted on a common base. The drive consists of the serial array of two mechanisms, one planar epicyclic gear train and one five-bar linkage, composed of the base, the screw, the slider, the intermediate and the output link, as shown in Fig. 1. One of the motors drives the sun gear, the other the ring gear of the epicyclic train. The two outputs of the train are the angular velocity of the planet-carrier and the relative angular velocity of the planets w.r.t. their carrier. The former provides directly the pan rate, the latter the tilt rate via the linkage.

The mapping of the motor rates into the pan and tilt rates is described by the 2×2 *drive Jacobian* $\mathbf{J}$. As it turns out, the latter is posture-dependent, in that it is a function of the tilt angle θ. The objectives of the optimization procedure described here are the minimization of both the condition number of $\mathbf{J}$ at a prescribed posture and the *transmission defect*, an index of the force-torque transmission characteristics [7]. To solve the optimization problem, a sequential-quadratic programming method based on an orthogonal decomposition of the space of design variables [8] is applied. The result is a set of optimum proportions of the various geometric parameters of the drives.

[1] Vijaykumar et al. [2] defined the dextrous workspace as that region of the positioning workspace where the end-effector of a serial robot can attain an arbitrary orientation. Arbitrary orientations being quite challenging with parallel robots, this definition is adapted here to the mobility limitations of the latter.

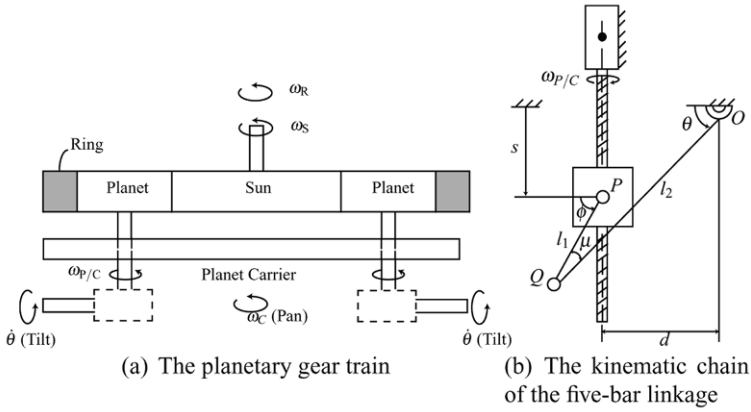


Fig. 1 The two components of the pan-tilt drive.

2 Kinematics of the Pan-Tilt Mechanism

From kinematics, the relation between the pan and the tilt rates, $\omega_1 = \omega_C$ and $\omega_2 = \dot{\theta}$, and the angular velocities of the sun and the ring gears, ω_S and ω_R, are found as:

$$\boldsymbol{\omega}_o = \mathbf{J}\boldsymbol{\omega}_i \tag{1a}$$

where

$$\boldsymbol{\omega}_o = \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix}, \quad \boldsymbol{\omega}_i = \begin{bmatrix} \omega_R \\ \omega_S \end{bmatrix}, \quad \mathbf{J} = \begin{bmatrix} \dfrac{1}{\nu+1} & \dfrac{\nu}{\nu+1} \\ \dfrac{k}{1-\nu} & \dfrac{-k\nu}{1-\nu} \end{bmatrix}, \quad k = \frac{\lambda \sin\theta - \sigma}{\delta \sin\theta - \sigma\cos\theta}\eta \tag{1b}$$

in which $\lambda = l_2/l_1$, $\sigma = s/l_1$, $\delta = d/l_1$ and $\eta = p/l_2$ are dimensionless parameters. Moreover, the pitch of the screw and the gear ratio between the sun and the ring gears are denoted by p and ν, respectively, while θ, l_1, l_2, s and d are shown in Fig. 1(b). The derivations leading to the above Jacobian matrix $\mathbf{J}$ are included in the Appendix.

To determine the required input rates for producing the desired pan and tilt rates, the above Jacobian should be inverted. However, to avoid round-off error amplifications, the mechanism should be dimensioned such that the condition number of the Jacobian attains a minimum. The square of this number based on the weighted Frobenius norm is obtained below [9]:

$$\kappa^2(\mathbf{J}) = \left[\frac{(1-\nu^4)}{4\nu k} \left(\frac{1}{(1+\nu)^2} + \frac{k^2}{(1-\nu)^2} \right) \right]^2 \tag{2}$$

3 Optimization of the Pan-Tilt Mechanism

The condition number of the Jacobian matrix, expressed in Eq. (2), is a function of two variables, namely, ν and k. However, unconstrained minimization [10] of the condition number over these two variables leads to λ, and hence l_2, unbounded. To cope with this problem, the radius of the sun gear is assumed to be equal to the diameter of the planet gear. The corresponding gear ratio $\nu = 0.5$ is then substituted into the expression for the condition number in Eq. (2), which leads to

$$\kappa^2(\mathbf{J}) = \frac{0.0434(1 + 9k^2)^2}{k^2} \tag{3}$$

Upon zeroing the derivative of the above expression w.r.t. k, its minimum is determined as $\kappa_{\min}(\mathbf{J}) = 1.25$, which occurs at $|k| = 1/3$; recalling the expression for k from Eq. (1b) reveals that more information is still required to synthesize the linkage.

In the realm of kinematic synthesis, a useful index called the *transmission quality* Q, or its complement, the *transmission defect* D, with $D = \sqrt{1 - Q^2}$, is used to evaluate the force-torque transmission characteristics of the synthesized mechanism. Denoting the transmission angle of the five-bar linkage with μ, the transmission defect D of the linkage is defined as:

$$D = \sqrt{\frac{1}{\Delta\theta} \int_{\theta_1}^{\theta_2} \cos^2 \mu d\theta}, \quad \Delta\theta = \theta_2 - \theta_1 \tag{4}$$

where θ_1 and θ_2 are the extreme values of the angle of rotation of the output link, which define the link mobility range.

For the five-bar linkage, the transmission angle is defined as the angle between the output link and the connecting rod, as shown in Fig. 1b. From geometry, we obtain

$$c_\phi = \lambda c_\theta - \delta, \quad s_\phi = \lambda s_\theta - \sigma \tag{5}$$

where c_x and s_x denote the cosine and the sine of angle x, respectively.

Multiplying the two expressions in Eq. (5) by c_θ and s_θ, respectively, and adding the results yield

$$c_\mu = \lambda - \delta c_\theta - \sigma s_\theta, \quad \mu = \phi - \theta \tag{6}$$

Moreover, s_μ can be obtained via the sine law applied to the triangle OPQ in Fig. 1b as:

$$s_\mu = \delta s_\theta - \sigma c_\theta \tag{7}$$

To eliminate the variable θ, Eqs. (6) and (7) are cast in vector form:

$$\begin{bmatrix} c_\theta & -s_\theta \\ s_\theta & c_\theta \end{bmatrix} \begin{bmatrix} \sigma \\ \delta \end{bmatrix} = \begin{bmatrix} -s_\mu \\ \lambda - c_\mu \end{bmatrix} \tag{8}$$

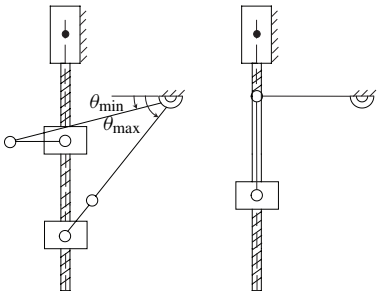


Fig. 2 The five-bar linkage: (a) at the extrema of the slider motion (b) at the optimum posture.

Since the matrix coefficient in Eq. (8) is a rotation, it preserves the Euclidean norm, whence,

$$\sigma^2 + \delta^2 = 1 + \lambda^2 - 2\lambda c_\mu \Rightarrow c_\mu = \frac{1 + \lambda^2 - \delta^2 - \sigma^2}{2\lambda} \tag{9}$$

Upon substituting c_μ from Eq. (9) into Eq. (4), the transmission defect D of the linkage is found as:

$$\begin{aligned} D^2 &= \frac{1}{4\lambda^2 \Delta\sigma}\left[\left(1 + \lambda^2 - \delta^2\right)^2 \Delta\sigma - \frac{2}{3}\left(1 + \lambda^2 - \delta^2\right)\Delta\sigma^3 + \frac{1}{5}\Delta\sigma^5\right], \\ \Delta\sigma^k &= \sigma_u^k - \sigma_l^k \end{aligned} \tag{10}$$

where σ_l and σ_u are the the lower and the upper bounds of $\sigma = s/l_1$. The right-hand side of Eq. (10) can be streamlined, to yield

$$\begin{aligned} \zeta &= \frac{1}{X_2}\left[(1 - X_1 + X_2)^2 - \frac{2}{3}c_1\,(1 - X_1 + X_2) + \frac{1}{5}c_2\right], \quad \zeta = 4D^2 \\ c_1 &= \sigma_u^2 + \sigma_u\sigma_l + \sigma_l^2 \\ c_2 &= \sigma_u^4 + \sigma_u^3\sigma_l + \sigma_u^2\sigma_l^2 + \sigma_u\sigma_l^3 + \sigma_l^4, \quad X_1 = \delta^2, \quad X_2 = \lambda^2 \end{aligned} \tag{11}$$

3.1 Minimization of the Transmission Defect

Our objective is now to find the dimensions of the mechanism that minimize its transmission defect. Since no feasible solution to the unconstrained optimization problem exists, constraints stemming from our engineering insight into the requirements of the task at hand should be brought-in. For instance, the mechanism must be capable of tilting in an interval which is larger than the prescribed $[\theta_{\min}, \theta_{\max}]$.

Considering Fig. 2(a), the constraints sought are derived as:

$$\delta - (1 + \lambda)\cos\theta_{\max} \leq 0, \quad \lambda\cos\theta_{\min} - \delta - 1 \leq 0 \tag{12}$$

Table 1 The local minima of the transmission defect.

	Initial guess:$[\lambda, \delta, y_1, y_2]^T$	λ	δ	y_1	y_2	ζ
1	$[1, 1, 1, 1]^T$	1	1	0	1	0.8
2	$[1, 1, 1, 0]^T$	1.9250	0.9250	0.7332	0	1.7779
3	$[1, 1, 0, 0]^T$	3	2	0	0	1.8667

It is noteworthy that the reflection of the mechanism w.r.t. the horizontal line does not alter the constraint relations expressed in Eq. (12). In our case, the tilt angle is desired to vary in the interval $[-60°, 60°]$; thus, $\theta_{\min}$ and $\theta_{\max}$ are assigned 0° and 60°, respectively.

The dimensionless bounds σ_l and σ_u corresponding to $\theta_{\min}$ and $\theta_{\max}$ are

$$\sigma_l = \lambda \sin\theta_{\min}, \quad \sigma_u = (1+\lambda)\sin\theta_{\max} \tag{13}$$

The optimization problem can now be stated as: minimize ζ in Eq. (11) over λ and δ subject to the two inequality constraints (12).

To solve the optimization problem, the inequality constraints are transformed into equalities by introducing two slack variables y_1 and y_2:

$$\delta - (1+\lambda)\cos\theta_{\max} + y_1^2 = 0, \quad \lambda\cos\theta_{\min} - \delta - 1 + y_2^2 = 0 \tag{14}$$

A numerical algorithm is utilized to solve the optimization problem; the algorithm used is based on sequential-quadratic programming using an orthogonal decomposition of the design-variable space with Gerschgorin stabilization [8]. Since the result of the algorithm depends highly on the choice of initial guess, the computer code is executed for several different initial guesses, mostly chosen on the boundaries of the feasible region as a result of not achieving acceptable solutions to the unconstrained minimization problem. Considering that both variables λ and δ are length ratios, and hence, positive, the results which violate this condition are simply left out. The feasible local minima of the problem are tabulated in Table 1.

Comparing the transmission defect values reported in Table 1, the first solution with $\lambda = \delta = 1$ is selected. The optimum five-bar linkage is displayed in Fig. 2(b) at the posture where $\theta = 0$, the transmission angle being 90°.

The last parameter to be determined is the pitch of the screw. For this matter, we require the five-bar linkage to produce the minimum condition number of the Jacobian at the posture with the maximum transmission angle, i.e., $\theta = 0$. As mentioned previously, the condition number of the Jacobian matrix attains a minimum at $|k| = 1/3$.

Substituting Eq. (7) into Eq. (1b), and letting $\mu = 90°$, we obtain

$$k = c_\theta \eta \tag{15}$$

Since θ vanishes at the posture where $\mu = 90°$, the optimum value of η is found as 1/3. The final optimum dimensions of the five-bar linkage satisfy the relations

Fig. 3 The CAD model of an optimum pan-tilt mechanism.

below:

$$l_1 = l_2 = d, \quad p = \frac{l_1}{3} \tag{16}$$

These dimensions lead to a mechanism which has the minimum transmission defect in its mobility range, while the condition number of its Jacobian matrix is a minimum at the posture with the maximum transmission angle, i.e., $\mu = 90°$. Finally, the pitch obtained from Eq. (16) should be available in ball-screw catalogs, which thus constrains l_1 to yield a feasible p. The CAD model of the pan-tilt mechanism with an optimum set of dimensions is illustrated in Fig. 3.

4 Conclusions

An innovative parallel pan-tilt drive was introduced, which consists of a two-dof planetary gear train whose sun and ring gears are actuated by means of two motors mounted on the same base platform, the outputs being the rotations of the planet-carrier and the planet gears. The first rotation produces the pan motion, while the rotation of each planet w.r.t. the planet-carrier is transmitted to a five-bar linkage so as to obtain the tilt motion.

The optimum dimensioning of the mechanism was sought such that the condition number of the Jacobian matrix becomes a minimum at an optimum posture – defined by an angle θ – where the transmission angle of the mechanism is 90°. Moreover, to enhance the force-torque transmission characteristics of the mechanism, its transmission defect was minimized. However, since the unconstrained optimization problem did not lead to a feasible solution, two geometric constraints guaranteeing a prescribed mobility range were introduced to constrain the optimization problem. The problem thus formulated was solved by resorting to a sequential-quadratic programming method based on an orthogonal decomposition of the design-variable space. The algorithm converged to three acceptable solutions, depending on various initial

guesses chosen. Upon comparing the three results, the optimum mechanism with the minimum transmission defect was found.

The downside of this mechanism is its size, as it requires, for the tilt axis sweeping an angle of 120°, a stroke Δs of the screw sliders, of $2\sqrt{3}l_1 = 3.464l_1$, which based on the assumption $\nu = 0.5$, yields $\Delta s = 2.598d_s$, with d_s denoting the pitch diameter of the sun gear. In the current design of the SMG, $d_s = 46$ mm, which leads to $\Delta s = 119.508$ mm. The stroke requirement is larger than the space available in the current SMG prototype, which calls for some extra design work.

Appendix

From the kinematics of planetary gear trains, we have

$$\omega_C = \frac{1}{\nu+1}\omega_R + \frac{\nu}{\nu+1}\omega_S, \quad \omega_{P/C} = \frac{1}{1-\nu}\omega_R - \frac{\nu}{1-\nu}\omega_S \tag{17}$$

Adding the squares of the two relations in Eq. (5) and calculating the time derivatives of the angles of interest from the equation thus resulting yields

$$\dot{\theta} = \frac{\lambda s\theta - \sigma}{\lambda(\delta s\theta - \sigma c\theta)}\dot{\sigma}, \quad \dot{\sigma} = \frac{p}{l_1}\omega_{P/C} \tag{18}$$

References

1. Kong, X., Gosselin, C.M.: *Algorithms for Computer Algebra*. Springer Tracts in Advanced Robotics, Vol. 33. Springer-Verlag, Berlin/Heidelberg (2007).
2. Vijaykumar, R., Tsai, M.J., Waldron, K.J.: Geometric optimization of serial chain manipulator structures for working volume and dexterity. *The International Journal of Robotics Research* **5**, 91–103 (1986).
3. Morozov, A., Angeles, J.: The mechanical design of a novel Schönflies motion generator. *Robotics and Computer-Integrated Manufacturing* **23**, 82–93 (2007).
4. Alizadeh, D., Angeles, J., Nokleby, S.: A novel pan-tilt motion generator. Technical Report TR CIM 10-01, Department of Mechanical Engineering and Centre for Intelligent Machines, McGill University (2010).
5. Khan, W.A.: The conceptual design of robotic architectures using complexity criteria. Ph.D. Thesis, Department of Mechanical Engineering, McGill University, Montreal, Canada (2007).
6. Khan, W.A., Caro, S., Pasini, D., Angeles, J.: Complexity analysis for the conceptual design of robotic architectures. In: *Advances in Robot Kinematics: Mechanisms and Motion*, J. Lenarčič and B. Roth (Eds.). Springer, Dordrecht, pp. 359–368 (2006).
7. Gosselin, C.M., Angeles, J.: Optimization of planar and spherical function generators as minimum-defect linkages. *Mechanism and Machine Theory* **24**, 293–307 (1989).
8. Teng, C.-P., Angeles, J.: A sequential-quadratic programming algorithm using orthogonal decomposition with Gerschgorin stabilization. *ASME Journal of Mechanical Design* **123**, 501–509 (2001).
9. Golub, J.H., Van Loan, C.F.: *Matrix Computations*. The Johns Hopkins University Press, Baltimore (1989).
10. Rao, S.S.: *Engineering Optimization. Theory and Practice*. Wiley (1996).

LQP-Based Controller Design for Humanoid Whole-Body Motion

Joseph Salini, Sébastien Barthélemy and Philippe Bidaud

Institut des Systèmes Intelligents et de Robotique, Université Pierre et Marie, CNRS UMR 7222 Pyramide T55, 4 Place Jussieu, 75252 Paris Cedex 05, France; e-mail: {joseph.salini, sebastien.barthelemy, philippe.bidaud} @isir.fr

Abstract. Motion synthesis is a central question in humanoid robotics research. The present paper proposes a generic approach exploiting a quadratic programming method for solving the problem of motion coordination under physical constraints. Details about the problem formulation for dealing with sequences of complex activities are given. The method is illustrated in several examples using the iCub robot. We will show how whole-body control under multiple contact constraints can be achieved. We will also discuss how it is possible to organize the priorities between the tasks in this framework to balance multiple objectives.

Key words: Whole-body motion, environment interaction, LQP, hierarchy.

1 Introduction

This paper considers the problem of task/posture coordination of humanoid robots interacting physically with their environment. It focuses on the design of an efficient framework for solving the underlying control problem in a generic way. Controlling the postural balance when the robot realizes a complex task is a very challenging problem which has received a limited number of contributions until now.

Task/posture coordination and whole-body motion are usually performed using linear algebra methods (pseudo-inverses, orthogonal projectors) on the linearized laws of motion [1–3]. These methods allow enforcing kinematic constraints and a strong hierarchy between tasks. Although they do not provide a natural way to take into account the inequalities that arise when modelling contacts and joint limits, authors [4] used them successfully to control a humanoid system in contact with its environment.

Keeping the robot in balance during dynamic motions is classically done using the zero moment point (ZMP) [5, 6] and the base of support, which are only defined if all the contacts are coplanar. Several authors extended them to handle three-dimensional contact configurations. For instance, Harada applied them to a robot grasping its environment [7]. However many of these extensions suffer from a lack of generality and physical meaning.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_19,

A notable exception is the work by Wieber [8, 9] where the problem, including spatial contacts, is modelled and solved using a linear quadratic program (LQP). The LQP formulation still requires a linearized model, but handles the linear inequalities directly. Later works [10–13] also use LQP in similar ways for a range of humanoid systems and motions. In this paper, we propose on validate a general framework for the synthesis of dynamic whole-body complex activities of humanoid robots while taking into account in passive or active way all the internal physical constraints and external contacts acting on the various bodies of the humanoid.

In the first section, we will recall the mechanical model for constrained motions of humanoid robot, as well as the task description with their related controllers. We The second section introduces the design of the multi-task LQP-based controller and the way we deal with task hierarchy for generating a whole-body motion satisfying postural stability and upper-limbs task constraints. The third section gives the implementation of this general control framework on a virtual iCub robot [14] for several scenarii including manipulation, rising up, walking, etc. Simulation results are presented and discussed. Finally, conclusions and perspectives for further works are given on the last section.

2 Modelling

2.1 Whole-Body Dynamics

The humanoid robot is a mechanical structure which can be viewed as an underactuated tree-structure of rigid bodies with a free-floating base (the trunk) on which are articulated the upper and lower limbs. The robot has active joints equipped with actuators that can generate bounded generalized forces (torques). The dynamic behavior of the system is described by its equation of motion, and the interaction forces with the environment are usually acting on several locations of the limbs, sometimes directly on the trunk.

The iCub robot is a humanoid robot sized as a three and half year-old child robot made of 53 degrees of freedom (dof) including those integrated in the hands. We do not model the hands in the present work, so the resulting virtual robot is left with only 38 dof: 32 joint parameters which are concerned by the main movements of the robot plus 6 additional parameters which are used to locate the trunk with a reference frame. For this kind of mechanical system, the Euler-Lagrange equations give the equation of motion of the robot Eq. (1), and the model is extended with its physical limitations (joint torques Eq. (2), joint position limits Eq. (3), etc.).

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} = \mathbf{g}(\mathbf{q}) + \mathbf{S}\tau + \mathbf{J_c}(\mathbf{q})^{\mathbf{t}}\mathbf{f_c} \tag{1}$$

$$\tau_{\mathbf{min}} \leq \tau \leq \tau_{\mathbf{max}} \tag{2}$$

$$\mathbf{q_{min}} \leq \mathbf{q} \leq \mathbf{q_{max}} \tag{3}$$

$\mathbf{M}, \mathbf{N}, \mathbf{g}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \mathbf{S}, \tau, \mathbf{J_c}, \mathbf{f_c}$ are respectively the generalized mass matrix, the Coriolis and non-linear effects matrix, the generalized gravity vector, the generalized position, velocity and acceleration vectors, the actuation matrix, the robot torque vector, the contact points Jacobian matrix and finally the contact forces vector. Actuation matrix $\mathbf{S}$ allows to actuate or not each degree of freedom. All the mechanical properties of the iCub robot has been carefully identified for being integrated in the simulation environment described in section 4.

2.2 Contact Model

For all the humanoid bodies interacting with the environment, the contacts are described from a set of punctual contact with friction. Each contact point i has a velocity $\mathbf{v_{ci}} = \mathbf{J_{ci}}(\mathbf{q})\dot{\mathbf{q}} \in \mathbb{R}^3$, and each point develops a force denoted $\mathbf{f_{ci}} \in \mathbb{R}^3$. Four cases may happen

- the contact is persistent, $\mathbf{v_{ci}} = \mathbf{0}$ and $\mathbf{f_{ci}}$ lies inside the Coulomb cone,
- the contact is lifting, $\mathbf{v_{ci}}.\mathbf{n} > 0$ and $\mathbf{f_{ci}} = \mathbf{0}$,
- there is no contact, $\mathbf{v_{ci}} \in \mathbb{R}^3$ and $\mathbf{f_{ci}} = \mathbf{0}$.
- the contact is sliding $\mathbf{v_{ci}} \times \mathbf{n} \neq \mathbf{0}$ and $\mathbf{f_{ci}} \neq \mathbf{0}$

where $\mathbf{n}$ is the normal vector of contact. In order to be integrated in the LQP-based dynamic controller, we express the contact point constraints in terms of joint accelerations and forces constraints under linearized cones, which gives for the two first cases for instance:

$$case\ 1\ (\mathbf{v_{ci}} = \mathbf{0})\ :\quad \mathbf{J_{ci}}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}_{\mathbf{ci}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} = \mathbf{0} \tag{4}$$

$$\mathbf{C}\mathbf{f_{ci}} \leq \mathbf{0} \tag{5}$$

$$case\ 2\ (\mathbf{v_{ci}}.\mathbf{n} > \mathbf{0})\ :\ \left(\mathbf{J_{ci}}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}_{\mathbf{ci}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}\right).\mathbf{n} > \mathbf{0} \tag{6}$$

2.3 Task Description and Control

Basically, a task is described as an acceleration imposed to an element related to the robot (joints, frame, free-floating base, center of mass, etc.). These tasks can be transformed into a generalized acceleration through the Jacobian matrix of the point, the frame, the joint, etc. involved in the task. The acceleration may result from the action of a controller. Several kinds of controllers have been implemented, as Proportional Derivative (PD) controller to reach a pose or a position, impedance controller to monitor the contact force, and predictive controller in the particular case of locomotion.

If we consider for instance the action of the PD controller, the desired acceleration of the element related to the task T will be computed through the proportional and derivative errors. If the related element of task T is a frame, the proportional

error (here, the pose error) $\mathbf{x_T^{err}}$ between the controlled frame and the goal is the concatenation of the position error $\delta\mathbf{p} = \mathbf{p^{des}} - \mathbf{p}$ and the attitude error $\delta\mathbf{Q}$. We use quaternion $\mathbf{Q^{des}} = (\eta, \theta) \in \mathbb{R} \times \mathbb{R}^3$ to describe the attitude [15], and the error is given by $\delta\mathbf{Q} = (\eta.\eta^{des} + \theta.\theta^{\mathbf{des}}, \eta.\theta^{\mathbf{des}} - \eta^{des}.\theta - \theta \times \theta^{\mathbf{des}})$. Finally, the desired acceleration of the frame related to the task T is obtained in the following equation $\dot{\mathbf{v}}_\mathbf{T}^{\mathbf{des}} = \mathbf{K_p}\mathbf{x_T^{err}} - \mathbf{K_d}\mathbf{v_T}$ where $\mathbf{v_T} = \mathbf{J_T}\dot{\mathbf{q}} \in SE(3)$ is the velocity of the related element with $\mathbf{J_T}$ the Jacobian matrix of task T, $\mathbf{K_p}$ the stiffness and $\mathbf{K_d}$ the damping.

To perform locomotion, a predictive controller described on Kajita work [6] is used. The controller predicts the future position of an approximate Zero Moment Point (ZMP) on a horizontal plane along a predefined horizon, and the prediction is updated periodically to correct the trajectory during time. To compute this approximation, the humanoid robot is modelled as an inverted pendulum, $z\tilde{m}p_{x,y} = com_{x,y} - co\ddot{m}_{x,y}(com_z/g)$ where g represents the gravity acceleration. It generates a ZMP trajectory close to a reference, and applies the computed input. This control is used for instance in example 4.3.

3 LQP-Based Controller Design

3.1 Designing LQP

The approach we have used for the synthesis of dynamic whole-body motions is based on constrained optimization algorithm (quadratic program QP) exploiting a multi-criteria weighted quadratic function reflecting the tasks to perform and their relative importance. A QP has the following general form where $Fopti(\mathbf{x})$ is a linear quadratic function of $\mathbf{x}$ and represents the weighted criteria to minimize.

$$\begin{aligned} \textit{solve: } \mathbf{x}^* &= \text{argmin}_{(\mathbf{x})}(Fopti(\mathbf{x})) \\ \textit{subject to: } \mathbf{Ax} &= \mathbf{b} \\ \mathbf{Cx} &\leq \mathbf{d} \end{aligned}$$

The problem here is to cast physical laws and desired motions into a suitable LQP. Physical laws are defined by the equations Eqs. (1)–(6). Since they must be respected, we set them as equality and inequality constraints in LQP. Eq. (1) is generally non-linear, so the linearized equations around $(\mathbf{q}, \dot{\mathbf{q}})$ are used, and the problem is solved at each time step. $\ddot{\mathbf{q}}, \tau, \mathbf{f_c}$ are the LQP variables, and when a solution exists, the optimal torque τ^* found by LQP will finally be applied to the robot.

$$\mathbf{q_{min}} \leq \mathbf{q} + \dot{\mathbf{q}}h + \ddot{\mathbf{q}}h^2/2 \leq \mathbf{q_{max}} \tag{7}$$

A basic LQP consists in minimizing τ^2 to generate smooth motion, under the constraints Eqs. (1)–(6). All these equations are functions of LQP variables, except Eq. (3). This one will be replaced in the LQP formulation by an inequation which constrains the predicted position of $\mathbf{q}$. Let us consider that the acceleration $\ddot{\mathbf{q}}$ is

constant during a prediction horizon h. At time t, we can compute the future position of $\mathbf{q}$ at time $t+h$, and inequation Eq. (3) becomes Eq. (7).

3.2 Performing Tasks

A task has to be converted into a form suitable for the LQP. The effective acceleration of the element related to the task T is computed as $\dot{\mathbf{v}}_\mathbf{T} = \mathbf{J_T}\ddot{\mathbf{q}} + \dot{\mathbf{J}}_\mathbf{T}\dot{\mathbf{q}}$ where $\mathbf{J_T}, \dot{\mathbf{J}}_\mathbf{T}$ are the Jacobian and derivative Jacobian matrices of task T. The control is possible because the equation is function of $\ddot{\mathbf{q}}$. Section 2.3 gives means to compute desired acceleration $\dot{\mathbf{v}}_\mathbf{T}^\mathbf{des}$ at each time step. LQP has to find $\ddot{\mathbf{q}}$ such as $\mathbf{J_T}\ddot{\mathbf{q}} + \dot{\mathbf{J}}_\mathbf{T}\dot{\mathbf{q}} = \dot{\mathbf{v}}_\mathbf{T}^\mathbf{des}$, so the task is defined as $\min_{(\ddot{\mathbf{q}})}(\Delta_T = (\mathbf{J_T}.\ddot{\mathbf{q}} + \dot{\mathbf{J}}_\mathbf{T}.\dot{\mathbf{q}} - \dot{\mathbf{v}}_\mathbf{T}^\mathbf{des})^2)$.

To deal with several tasks simultaneously, the quadratic cost function is set as the sum of the weighted tasks functions, $\min_{(\ddot{q})}(\alpha_I.\Delta_I + \alpha_J.\Delta_J + \cdots)$ where α_I, α_J represent respectively the importance coefficients of tasks I, J. Indeed, the tasks with the highest coefficient cost more, and will be minimized better than the following ones. Although this approach is not equivalent to a hierarchy, it may offer more flexibility because tasks are not constant during the whole simulation: objectives can move, the relative importance can change, and tasks may be active or not.

In order to avoid discontinuity in controlled task accelerations and consequently in applied torques when a goal is replaced by another one, a smooth transition is required. A solution to this problem is to create a new task T' which controls the same element of the robot and to give it a new objective. If α_T decreases to 0 during a short period and $\alpha_{T'}$ increases from 0 to the old value of α_T during the same period, a continuous change from an objective to another is insured. This will be illustrated in example 4.2.

4 Application to the Virtual iCub

4.1 Managing Multi-Tasks

The method has been validated through various scenarii where numerous tasks are performed. The simulator used to perform the simulations is Arboris [16], a robotic simulator designed for the Matlab software, and LQP is solved at each time step thanks to the Yalmip software [17]. The simulation time step used in the following simulations is $\delta t = 2.5$ ms and the prediction horizon used in Eq. (7) is $h = 0.125s$. About task control, a proportional derivative control law is generally used. For all tasks, the stiffness and the damping are set as $\mathbf{K_p} = 30\mathbb{I}\, s^{-2}$ and $\mathbf{K_d} = 2(\mathbf{K_p})^{1/2}\, s^{-1}$, where $\mathbb{I}$ represents the identity matrix.

In this simulation, several tasks are performed simultaneously. Figure 1 shows what tasks are performed. The robot moves the two hands simultaneously at two different locations, and after one second, the right hand goal changes and reaches the

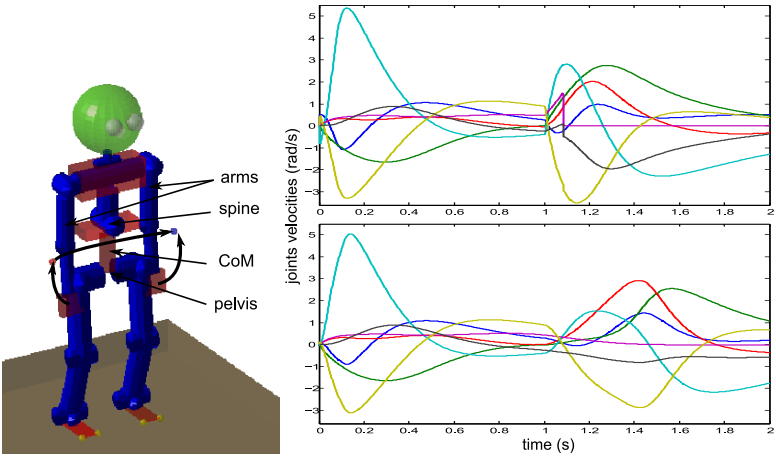


Fig. 1 Left: Controlled part of the robot; Right: Right arm joints velocities with (down) and without (up) joint stop anticipation and smooth α change.

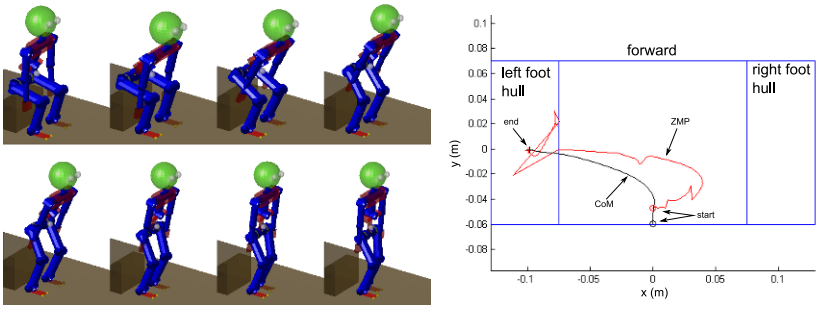


Fig. 2 Left: Sit-to-stand sequence; Right: ZMP and CoM trajectories while standing up.

left hand goal. the center of mass ($\alpha_{CoM} = 1$) is controlled in position, the pelvis and the hands ($\alpha_{pelvis} = \alpha_{hands} = 10^{-2}$) in position and attitude, and the spine ($\alpha_{spine} = 10^{-2}$) and the general posture ($\alpha_{posture} = 10^{-8}$) in angular position.

In Fig. 1, the graph shows the right arm joint trajectories. On the upper graph, a sharp change occurs, due to goal changes without transition. Furthermore, at 1.2 seconds, one joint hits the stop, so its velocity falls down to 0. But joint limits anticipation and smooth transitions avoid any big evolution in joint velocities, as shown on the lower graph.

4.2 Sit-to-Stand on One Foot

In this simulation, the robot stands up from a sitting position. The CoM position is controlled in the horizontal plane to reach the center of the base of support. When

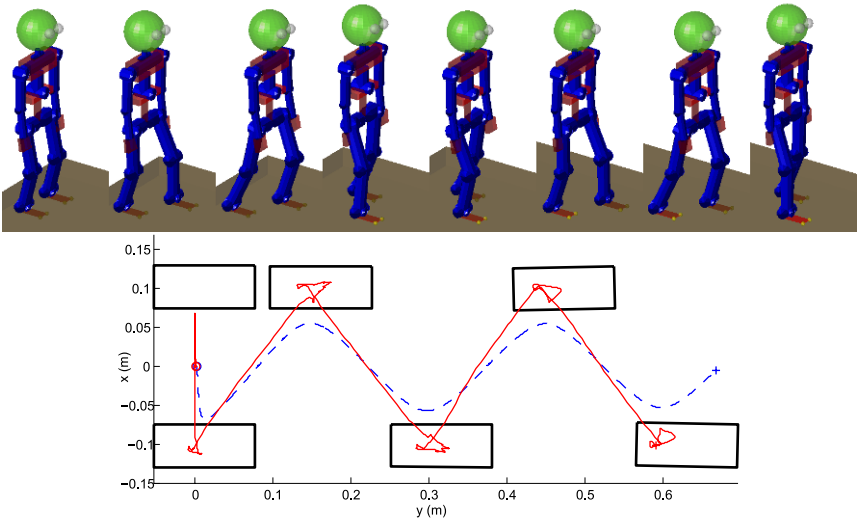


Fig. 3 Up: Walking sequence; Down: CoM (dotted) and filtered ZMP (solid) trajectories.

the CoM enters into the support polygon, its goal moves softly to the left foot, and is also controlled in vertical position. Notice that the pelvis attitude is not controlled in the sagittal plane while the robot rises. In Fig. 2, this sequence is shown: the robot stands up, then moves its CoM toward left foot, and lifts right foot when ZMP enters into left foot base of support. The CoM and ZMP trajectories are drawn in Fig. 2 from when CoM enters in the base of support to the end of the sequence.

4.3 Walking

In this experiment, the CoM is controlled trough the ZMP trajectory defined by a predictive controller (see Section 2.3). Besides, we control pelvis attitude, spine and general posture as before, and we add one task at each foot, to produce the foot trajectories computed *a priori* ($\alpha_{trajectory} = 10^2$). The CoM and approximated ZMP trajectories are shown in Fig. 3.

5 Conclusion

This paper presents a new generic dynamic LQP-based controller which solves at each time step a complex multi-tasks system, represented by weighted quadratic functions, under linear constraints. A kind of hierarchy is established with importance coefficients which allows to chaining up tasks. Several scenarii have been developed in simulation on a virtual robot inspired from a real iCub. From this paper,

the genericity, robustness and efficiency of the framework designed in this paper will be checked. Other simulations with more complex tasks and events will be carried out, followed by applications on the real robot.

References

1. O. Khatib. Dynamic control of manipulators in operational space. In *Sixth CISM-IFToMM Congress on Theory of Machines and Mechanisms*, pp. 1128–1131 (1983).
2. P. Baerlocher and R. Boulic. An inverse kinematics architecture enforcing an arbitrary number of strict priority levels. *Visual Computer*, **20**(6), 402–417 (2004).
3. L. Sentis. Synthesis and control of whole-body behaviors in humanoid systems. PhD thesis, Stanford University (2007).
4. Jaeheung Park. Control strategies for robots in contact. PhD thesis, Stanford University (2006).
5. Ph. Sardain and G. Bessonnet. Forces acting on a biped robot. Center of pressure – zero moment point. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, **34**(5), 630–637 (2004).
6. S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, Taipei, Taiwan (2003).
7. K. Harada, H. Hirukawa, F. Kanehiro, K. Fujiwara, K. Kaneko, S. Kajita, and M. Nakamura. Dynamical balance of a humanoid robot grasping an environment. In *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, pp. 1167–1173 (2004).
8. P.-B. Wieber. Constrained dynamics and parametrized control in biped walking. In *Proceedings International Symposium on Mathematical Theory of Networks and Systems*, Perpignan (2000).
9. P.-B. Wieber. On the stability of walking systems. In *Proceedings International Workshop on Humanoid and Human Friendly Robotics* (2002).
10. F.-T. Cheng, T.H. Chen, and Y.-Y. Sun. Efficient algorithm for resolving manipulator redundancy – The compact QP method. In *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, Nice, France (1992).
11. Y. Abe, M. da Silva, and J. Popovic. Multiobjective control with frictional contacts. In *Proceedings Symposium on Computer Animation (SCA)* (2007).
12. C. Collette, A. Micaelli, C. Andriot, and P. Lemerle. Dynamic balance control of humanoids for multiple grasps and non coplanar frictional contacts. In *Proceedings of Humanoids'07* (2007).
13. S. Barthélemy, P. Bidaud, and A. Micaeli. *Contrôle de l'équilibre des humains virtuels*, chapter 2. École des Mines de Paris (2009).
14. G. Metta, G. Sandini, D. Vernon, L. Natale, and F. Nori. The iCub humanoid robot: an open platform for research in embodied cognition. In *Permis: Performance Metrics for Intelligent Systems Workshop*, Washington DC, USA, August (2008).
15. J. Yuan. Closed-loop manipulator control using quaternion feedback. *IEEE Journal of Robotics and Automation*, **4**(4), 434–440 (1988).
16. S. Barthélemy and A. Micaelli. Arboris for Matlab.
17. J. Löfberg. Yalmip: A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan (2004).

Persistent Screw Systems

M. Carricato and J.M. Rico Martínez

[1]*DIEM – Department of Mechanical Engineering, University of Bologna, Italy; e-mail: marco.carricato@mail.ing.unibo.it*
[2]*Deparment of Mechanical Engineering, University of Guanajuato, Mexico; e-mail: jrico@salamanca.ugto.mx*

Abstract. In 1978, Hunt found a set of vector subspaces of screws that guarantee 'full-cycle mobility' of linkages and exhibit remarkable properties. They are subalgebras of the Lie algebra $se(3)$ of the Euclidean group and they are at the basis of most families of mechanisms with special motion capabilities. This paper proves the existence of screw systems that, though not being subalgebras of $se(3)$, still exhibit important properties for full-cycle motions, namely the invariance of both the space dimension and the pitch of the principal screws. Such systems are named *persistent* and they are believed to play an important role in both mobility analysis and mechanism synthesis.

Key words: Screw theory, mobility analysis, mechanism synthesis.

1 Introduction

Screw systems, which are vector subspaces of the Lie algebra $se(3)$ of the Euclidean group $SE(3)$, are fundamentals tools in the kinematics and statics of rigid-body mechanisms and structures. In 1978, Hunt [1] found a set of screw systems that guarantee 'full-cycle mobility' of linkages. These systems were proven to be subalgebras of $se(3)$ and labeled *invariant* in [2, 3]. They exhibit remarkable invariance properties and they are at the basis of most families of mechanisms with special motion capabilities (planar, spherical, translational, Schoenflies, etc.). Indeed, mechanisms of these sorts may be assembled from components picked more or less at random from 'kits' of parts that meet simple and broad geometrical criteria, e.g. parallelism of axes, equality of pitches, and the like.

This paper shows the existence of *persistent screw systems* (PSSs), which, though not being subalgebras of $se(3)$, still exhibit remarkable properties for full-cycle motions, namely the invariance of both the space dimension and the pitch of the principal screws. A kinematic chain whose freedoms belong to a PSS in a reference configuration generates, in *any* other nonsingular pose, a screw subspace that is congruent to the reference one under a proper isometry. The concept of PSSs generalizes that of invariant systems, the latter emerging from the former when the mentioned isometry is the identity mapping. PSSs are believed to play an important

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_20, © Springer Science+Business Media B.V. 2010

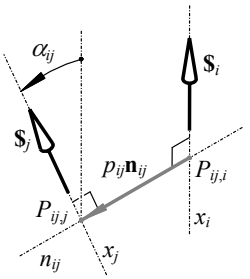


Fig. 1 Relative posture between two screws.

role in both mobility analysis and the synthesis of mechanisms, particularly spatial parallel manipulators.

In the following, n-dimensional subspaces of screws are referred to as n-systems and they are denoted by capital italic letters. When required, their dimension is specified by a lowercase subscript, i.e. S_n. Hunt's classification of screw spaces is adopted [1]. Accordingly, the locutions nG system and nR system, with R being a Roman numeral, denote respectively a *general* n-system and a special n-system of the Rth type. The subalgebras of $se(3)$ are denoted by capital calligraphic letters, followed by their characteristic geometric quantities in parentheses, e.g. $\mathscr{Y}(\mathbf{u},h)$. A screw representing a relative twist between two bodies is designated by **\$**. The axis of the screw is denoted by x, with $\mathbf{s}$ being a unit vector along x. Given $\$_i$ and $\$_j$, n_{ij} is the common perpendicular between the axes of the screws; $P_{ij,i}$ and $P_{ij,j}$ are the feet of n_{ij} on x_i and x_j; $\mathbf{n}_{ij}$ is a unit vector parallel to n_{ij} and directed from $P_{ij,i}$ to $P_{ij,j}$; p_{ij} and α_{ij} are the shortest distance and the relative angle between x_i and x_j, with α_{ij} being evaluated according to the right-hand rule about $\mathbf{n}_{ij}$ and being comprised in the interval $[-\pi/2,\pi/2]$ (Fig. 1). Once the relative posture between the axes of two screws is assigned (by the parameters p_{ij} and α_{ij}), a *rigid connection* (or a *link*) is said to be laid between them and the screws are said to be adjacent.

2 Definition

Consider an n-system S and let a set of n linearly independent screws $\$_i$, $i=1\ldots n$, belong to it, so that $S=\text{span}\{\$_1,\ldots,\$_n\}$. If the vectors of S represent instantaneous relative freedoms of motion, the screws $\$_i$ may be serially connected by an appropriate number of rigid connections, the link 0 being attached to the first kinematic element of $\$_1$, the link n being attached to the second kinematic element of n and the link i, $i=1\ldots n-1$, being laid between $\$_i$ and $\$_{i+1}$. This operation is called a *serial composition* of screws. The vectors of S represent all the admissible instantaneous motions of link n with respect to link 0, chosen as the reference frame. The so-obtained chain represents an *instantaneous kinematic generator* of S.

After the serial composition has been accomplished, each $\$_i$ may be thought of as a kinematic pair, thus governing the *finite* relative motion between the axes of $\$_{i-1}$ and $\$_{i+1}$, with the corresponding Euclidean displacement $\mathbf{D}_i(\theta_i)$ depending on both

the geometric parameters of $\$_i$ (the axis x_i and the pitch h_i) and the amplitude θ_i of the relative motion (the latter evaluated with respect to a reference configuration and according to the right-hand rule about $\mathbf{s}_i$). If S' is the n-system spawned by the kinematic generator in its reference pose, after a generic movement of the constituting screws (and out of singular configurations), a different n-system is generally produced, namely $S'' = \text{span}\{\$''_1, \ldots, \$''_n\}$, with the generic displaced screw being $\$''_i = \prod_{j=0}^{i-1} \mathbf{D}_j(\theta_j)\$'_i = \prod_{j=1}^{i} \mathbf{D}_j(\theta_j)\$'_i$, where $\mathbf{D}_0$ is equal to the identity mapping $\mathbf{I}$ and $\$'_i = \mathbf{D}_i\$'_i = \mathbf{D}_i^{-1}\$'_i$. It is evident that S'' depends on the amplitudes of all relative displacements with the exception of the last one, i.e. $S'' = S''\left(\theta_1, \ldots, \theta_{n-1}\right)$.

S is said to be a *persistent screw system* (PSS) if it admits a kinematic generator such that, for *arbitrary finite* motions $\{\mathbf{D}_1(\theta_1), \ldots, \mathbf{D}_{n-1}(\theta_{n-1})\}$, a Euclidean displacement $\mathbf{D}(\theta_1, \ldots, \theta_{n-1})$ exists so that

$$S'' = \mathbf{D}\left(\theta_1, \ldots, \theta_{n-1}\right) S', \tag{1}$$

namely, if S'' is congruent to S' under a *proper isometry*. This is equivalent to requiring that, for any $\$'' \in S''$, a screw $\$' \in S'$ exists such that $\$'' = \sum_{i=1}^{n} \alpha''_i \$''_i = \mathbf{D}\$' = \sum_{i=1}^{n} \alpha'_i \mathbf{D}\$'_i$, with α''_i and α'_i, $i = 1 \ldots n$, being real coefficients. Of course, if the Euclidean displacement $\mathbf{D}$ transforms S' into S'', the same transformation super-imposes the reciprocal system R' of S' to the reciprocal system R'' of S''. In order to lighten the notation, we will often not distinguish between a PSS and its kinematic generator. Conversely, we will use primes to specify, when necessary, the vector subspace generated by the kinematic generator in a specific configuration.

A PSS has persistent geometric properties, as the vector subspaces generated by the associated kinematic generator in all its possible nonsingular configurations are mutually congruent. It follows that a PSS is not only *type-invariant* (in the sense that it preserves the 'qualitative' geometric pattern that gives reason for its classification as an nG or nR system), but it also preserves its relevant 'quantitative' features, such as, in particular, the pitch of its principal screws (the *principal pitches*). Indeed, any vector subspace of screws is characterized by a distinctive pattern of principal screws that constitute a basis of it,[1] so that a rigid transformation of the system amounts to the rigid displacement of its principal screws. Accordingly, PSSs may be also defined as screw systems that preserve their principal pitches for arbitrary displacements of their associated kinematic generator. According to current terminology [4], any mechanical system embodying the kinematic generator of a PSS is a mechanical generator of it.[2]

It is worth observing that, since $\$''$ may be expressed as

$$\$'' = \mathbf{D}_1 \cdot \left(\alpha''_1\$'_1 + \alpha''_2\$'_2 + \alpha''_3\mathbf{D}_2\$'_3 + \cdots + \alpha''_n\mathbf{D}_2 \ldots \mathbf{D}_{n-1}\$'_n\right), \tag{2}$$

[1] To be more precise, any screw system is characterized by a distinctive set of principal screws if $n = 1$, 2 or 3 and by a distinctive set of principal reciprocal screws if $n = 4$ or 5, with the latter being the principal screws of the reciprocal system [1].

[2] Usually, there is not a one-to-one correspondence between kinematic and mechanical generators. Indeed, it will be seen that PSSs may be generated by composing *invariant screw systems* and these may be ordinarily produced by different arrangements of kinematic pairs.

$\$'' = \mathbf{D}\$'$ if and only if

$$\left(\alpha_1''\$_1' + \alpha_2''\$_2' + \alpha_3''\mathbf{D}_2\$_3' + \cdots + \alpha_n''\mathbf{D}_2 \ldots \mathbf{D}_{n-1}\$_n'\right) = \mathbf{D}_1^{-1}\mathbf{D}\$'. \tag{3}$$

In other words, S is persistent if and only if a proper isometry $\mathbf{D}\left(\theta_2, \ldots, \theta_{n-1}\right)$ exists such that

$$\text{span}\left\{\$_1, \$_2, \mathbf{D}_2\$_3, \ldots, \mathbf{D}_2 \ldots \mathbf{D}_{n-1}\$_n\right\} = \mathbf{D}\left(\theta_2, \ldots, \theta_{n-1}\right) S'. \tag{4}$$

Hence, the fulfillment of condition (1) only depends on the relative posture between the screws of the kinematic generator and, thus, on the relative displacements associated with its intermediate screws. As an immediate corollary, the persistent properties of a system do not depend on the choice of the reference frame. In fact, if an arbitrary displacement $\mathbf{D}_0 \neq \mathbf{I}$ is assigned to link 0, the screw subspace that is obtained, namely $S''' = \mathbf{D}_0 S'' = \mathbf{D}_0 \mathbf{D} S'$, still satisfies Eq. (1). In particular, by choosing $\mathbf{D}_0 = \mathbf{D}_i^{-1} \ldots \mathbf{D}_2^{-1}\mathbf{D}_1^{-1}$, the reference frame may be conveniently set on whichever link of the kinematic generator, with S''' being in this case $\text{span}\left\{\mathbf{D}_i^{-1} \ldots \mathbf{D}_2^{-1}\$_1, \ldots, \mathbf{D}_i^{-1}\$_{i-1}, \$_i, \$_{i+1}, \mathbf{D}_{i+1}\$_{i+2}, \ldots, \mathbf{D}_{i+1} \ldots \mathbf{D}_{n-1}\$_n\right\}$.

If, in Eq. (1), $\mathbf{D}\left(\theta_1, \ldots, \theta_{n-1}\right)$ coincides with the identity mapping, S is called an *invariant* or *full-cycle screw system* (ISS) [1–3]. Thus, ISSs are particular cases of PSSs. They do not only preserve the value of the principal pitches for arbitrary motions, but also the orientation and the position of the principal screws with respect to any possible reference frame chosen on the kinematic generator. In this paper, ISSs are denoted as follows: $\mathscr{H}(x,h)$ is the helicoid ISS of pitch h and axis x (if $h = 0$ or ∞, $\mathscr{H}$ becomes, respectively, the revolute ISS $\mathscr{R}(x)$ or the linear-translation ISS $\mathscr{P}(\mathbf{s})$); $\mathscr{C}(x)$ is the cylindrical ISS along x; $\mathscr{T}_2(\mathbf{u})$ is the planar-translation ISS perpendicular to $\mathbf{u}$; $\mathscr{T}_3$ is the spatial-translation ISS; $\mathscr{S}(O)$ is the spherical ISS about O; $\mathscr{Y}(\mathbf{u},h)$ is the helicoid-planar ISS of pitch h and direction $\mathbf{u}$ (the planar-motion ISS is obtained from $\mathscr{Y}(\mathbf{u},h)$ by letting $h = 0$, but it will not be distinguished from it in this paper); $\mathscr{X}(\mathbf{u})$ is the Schoenflies ISS parallel to $\mathbf{u}$.

It is well known that ISSs are subalgebras of $se(3)$ and they are in one-to-one correspondence with the subgroups of $SE(3)$ [3,4]. ISSs are symmetric under all finite motions taking place about any screw belonging to the system itself. In other words, if S is an ISS, S remains the image of itself under any displacement $\mathbf{D}$ belonging to the associated subgroup of $SE(3)$. Such a symmetry guarantees ISSs remarkable properties. As mentioned in Section 1, kinematic generators of any ISS may be assembled by choosing, more or less at random, from a 'kit' of screws meeting simple and broad geometrical criteria (e.g. parallelism of axes, equality of pitches, etc.), the number of these screws (equal to or greater than n) and the sequence by which they are serially composed being irrelevant[3] [1]. Such a latitude of choice is not shared by PSSs that are not ISSs. Indeed, a generic PSS preserves its persistent character-

[3] Relevance, in this case, is evaluated only with respect to the ability of the kinematic generator to originate a specific screw system. The type, the number and the order of the screws do affect other kinematic properties of the generator, such as, for instance, its singular configurations.

istics (i.e. the pitch of its principal screws) only if a generally more *restricted* set of requisites is met when building the proper kinematic generators (cf. Section 4).

3 Forms of Persistent Screw Systems

The kinematic generators of two screw systems, e.g. $A_n = \text{span}\{\$_1, \ldots, \$_n\}$ and $B_m = \text{span}\{\$_{n+1}, \ldots, \$_{n+m}\}$, may be serially composed by laying a rigid connection between $\$_n$ and $\$_{n+1}$. Such an operation generates, algebraically, a vector space C_l that is the sum of A_n and B_m, namely a space whose dimension is $m+n-\dim(A_n \cap B_m)$ and whose elements represent the admissible instantaneous motions of link $n+m$ with respect to link 0, chosen as the reference frame.

If A_n and B_m are ISSs, the architecture according to which they are built is irrelevant and C_l only depends on the 'shape' of the connection between $\$_n$ and $\$_{n+1}$. The concept of serial composition may be thus extended to ISSs and denoted by the symbol $\circ$ (a single screw is, indeed, a one-dimensional ISS). As there is a one-to-one correspondence between any ISS and a subgroup of $SE(3)$, there is also a one-to-one correspondence between the serial composition of ISSs and the product of the associated subgroups. In other words, if A_n and B_m are ISSs, the set of the admissible displacements between link $n+m$ and link 0 is the kinematic bond stemming from the product of the displacement subgroups associated with A_n and B_m [4].

Any PSS may be obtained as the composition of a convenient number of ISSs (it suffices to observe that any screw system may be decomposed in a sequence of 1-systems and any 1-system is an ISS). A particular sequence of ISSs generating a PSS S, namely $S = A^1 \circ \ldots \circ A^h$, and such that no two adjacent A^j, $j = 1 \ldots h$, may be merged to originate an ISS, is referred to as a *form* of S, with h being named the *cardinality* of the form. Of course, a PSS of dimension n cannot comprise ISSs of dimensions greater than n (and if it contains one of dimension n, it coincides with it). In general, any PSS admits several forms (with the 1-systems being an obvious exception). A form is said to be *unary* if $h = 1$, *binary* if $h = 2$, and so on. The *cardinality* c of a PSS is the smallest cardinality among all its possible forms, the PSS being accordingly called *unary*, *binary*, etc. ISSs are the only unary PSSs. A form $A^1 \circ \ldots \circ A^h$ is respectively said to be *conjoint* or *disjoint* depending on whether $A^s \cap A^t \neq \mathbf{0}$ or $A^s \cap A^t = \mathbf{0}$ for *all* pairs of adjacent A^s and A^t therein; it is *partially-joint* otherwise. As the vector space comprising the admissible instantaneous twists between two rigid bodies does not depend on which one of them is chosen as the reference frame, it is obvious that, if a PSS includes the form $A^1 \circ \ldots \circ A^h$, it also encompasses its reverse one, i.e. $A^h \circ \ldots \circ A^1$. Sufficient conditions for generating persistent screw systems are presented in the following.

The composition of two ISSs has been widely studied in the literature [4, 5], as a means to generate kinematically meaningful manifolds of $SE(3)$. It is proven here that the *composition between two ISSs always yields a PSS* (accordingly, any PSS obtained as the composition of two ISSs is binary). Let A_n and B_m be kinematic generators of ISSs and let C_l be their serial composition, i.e. $C_l = A_n \circ B_m$. In partic-

ular, let $A_n = \text{span}\{\$_1, \ldots, \$_n\}$ and $B_m = \text{span}\{\$_{n+1}, \ldots, \$_{n+m}\}$. For the properties of the ISSs, for any $\alpha_i'' \in \mathbb{R}$ and $\theta_i \in \mathbb{R}$, $i = 1 \ldots n+m$, suitable coefficients $\alpha_i' \in \mathbb{R}$ exist so that, out of singularities,

$$\alpha_1''\$_1' + \alpha_2''\mathbf{D}_1\$_2' + \cdots + \alpha_n''\mathbf{D}_1 \ldots \mathbf{D}_{n-1}\$_n' = \sum_{i=1}^{n} \alpha_i'\$_i', \tag{5}$$

$$\alpha_{n+1}''\$_{n+1}' + \alpha_{n+2}''\mathbf{D}_{n+1}\$_{n+2}' + \cdots + \alpha_{n+m}''\mathbf{D}_{n+1} \ldots \mathbf{D}_{n+m-1}\$_{n+m}' = \sum_{i=n+1}^{n+m} \alpha_i'\$_i'. \tag{6}$$

Hence, after an arbitrary displacement, $C_l'' = A_n' + \mathbf{D}_{1n}B_m'$, with $\mathbf{D}_{1n} = \prod_{i=1}^{n} \mathbf{D}_i(\theta_i)$. Since $\mathbf{D}_{1n}$ is a displacement of the subgroup associated with A_n, it is also $A_n' = \mathbf{D}_{1n}A_n'$ and hence $C_l'' = \mathbf{D}_{1n}(A_n' + B_m') = \mathbf{D}_{1n}C_l'$. Consequently, C_l is persistent.

Since A_n and B_m are ISSs, their intersection is also an ISS (the intersection of two subalgebras is also a subalgebra). It follows that, if $I_h = A_n \cap B_m \neq \mathbf{0}$, then h adjacent screws in both A_n and B_m may be suitably chosen in order to form persistent kinematic generators of I_h. In this case, if A_{n-h}^* and B_{m-h}^* are the portions of A_n and B_m that do not include I_h, then $A_n = A_{n-h}^* \circ I_h$ and $B_m = I_h \circ B_{m-h}^*$, so that, by considering that $I_h \circ I_h = I_h$,

$$C_l = A_{n-h}^* \circ I_h \circ I_h \circ B_{m-h}^* = A_{n-h}^* \circ B_m = A_n \circ B_{m-h}^*. \tag{7}$$

If A_{n-h}^* and B_{m-h}^* are not ISSs, 'subtracting' I_h from the kinematic generator of either A_n or B_m provides an easy procedure for obtaining non-binary disjoint forms of C_l. It is worth observing that A_{n-h}^* and B_{m-h}^* are not necessarily PSSs. They are so if their dimension is one or two, as all screw systems of such dimensions are PSSs. However, they may not be so, if their dimension is equal to three (for instance, 'subtracting' a 1-system from a Schoenflies subalgebra may leave three parallel screws of distinct finite pitch, which form a non-persistent 3IX-system).

A dual procedure with respect to the one described above consists in 'adding', between A_n and B_m, an ISS I_h which is 'partitioned' between them, namely an ISS that, though included in neither A_n nor B_m, is comprised in $A_n \circ B_m$. In this case, h^A adjacent screws in A_n and h^B in B_m may be suitably chosen so that $h^A + h^B = h$ and so as to form a kinematic generator of I_h. By letting $A_{n-h^A}^*$ and $B_{m-h^B}^*$ be the portions of A_n and B_m that do not contain the screws of I_h, it is then

$$C_l = A_n \circ B_m = A_{n-h^A}^* \circ I_h \circ B_{m-h^B}^* = A_n \circ I_h \circ B_m. \tag{8}$$

The described procedures are sufficient to generate non-binary PSSs, but they are not necessary. Obviously, they may be applied any time a PSS exhibits a form of arbitrary cardinality in which at least a pair of adjacent ISSs either has nonzero intersection or it encompasses a partitioned ISS.

PSSs may play an important role in both mobility analysis and mechanism synthesis. As far as mobility analysis is concerned, it is well known that mobility formulas that only rely on the number of links and the number and type of joints (such as the Kutzbach/Grübler equation) are applicable only if the *dimension* of the vector

space generated at any instant by the screws associated with the mechanism joints is invariant for arbitrary motions of the chain in a convenient finite neighborhood of the reference configuration. For this reason, such formulas are reputed to hold only for 'trivial' mechanisms, i.e. closed-loop linkages whose joints belong to mechanical generators of ISSs [4]. Distinct formulas must be resorted to for 'exceptional' linkages [4–6], which are closed loops emerging from the serial composition of the mechanical generators of two different ISSs.[4] Indeed, the theory of PSSs allows one to treat these two classes of mechanisms within a unified framework. The set of screw systems that preserve their *space dimension* for full-cycle motions of their kinematic generators is, in fact, much wider than that of ISSs. In particular, this property is shared by all PSSs. Consequently, the mobility of *all* linkages whose joints are assembled so as to form mechanical generators of PSSs (cf. Section 4) may be certified to be computable by a Kutzbach/Grübler-type criterion. In this perspective, the difference between 'trivial' and 'exceptional' mechanisms fades away. Since the composition of two ISSs always generates a PSS, 'exceptional' kinematic chains are mechanical generators of binary PSSs and as such they may be studied.

In the field of robot design, one of the most successful approaches for the synthesis of spatial parallel manipulators is based on the theory of constraint wrenches [7]. These are the wrenches reciprocal to the screws associated with the joints of the mechanism legs. Each leg generates a constraint system partially restraining the output displacement, so that the overall motion space of the end-effector is that reciprocal to the sum of the constraint spaces produced by all legs. The key issue to generate a specific output consists in synthesizing legs able to produce, for finite motions of the end-effector, predetermined and persistent constraint systems, typically convenient sets of forces and torques. In this perspective, PSSs are a valuable tool, since a leg that is the mechanical generator of a PSS is *guaranteed* to generate a constraint system that remains congruent to the reference one for full-cycle motions. For this reason, many mechanical generators of PSSs, even though not recognized as such, were widely used in the literature for the synthesis of innovative parallel manipulators (cf. Section 4 and [7]). The exhaustive classification of all PSSs opens up further possibilities.

4 Persistent Screw Systems of Dimension 3, 4 and 5

Figure 2 shows three examples of 5-dimensional PSS. The forms in Figs. 2(a)–(b) emerge from the composition of two helicoid-planar ISSs. The former, i.e. $\mathscr{Y}(\mathbf{u},h)\circ\mathscr{Y}(\mathbf{u}',h')$, is binary and conjoint, whereas the latter, which is obtained from the previous one by 'subtracting' the subalgebra $\mathscr{Y}(\mathbf{u},h)\cap\mathscr{Y}(\mathbf{u},h')=\mathscr{P}(\mathbf{u}\times\mathbf{u}')$, is

[4] Hervé [4] classified mechanisms into 'trivial', 'exceptional' and 'paradoxical'. The first two classes group the most common mechanisms and they are believed to be the only ones for which general mobility criteria may be contrived. 'Paradoxical' mechanisms, such as the Bennet one, require ad-hoc mobility analyses, since their motion capabilities rely on very specific choices of geometric parameters.

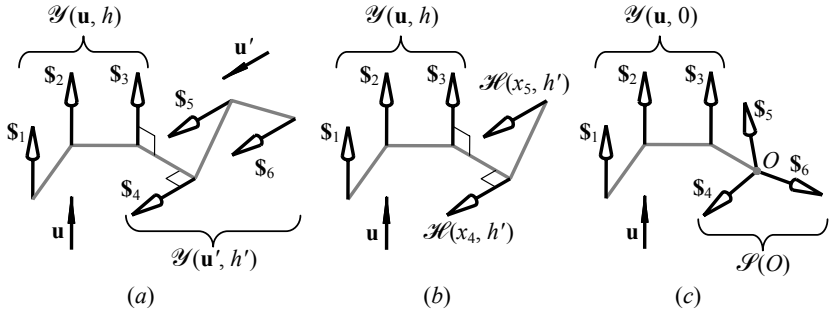


Fig. 2 Examples of PSSs: (a) $\mathscr{Y}(\mathbf{u},h)\circ\mathscr{Y}(\mathbf{u}',h')$, 5I-PSS; (b) $\mathscr{Y}(\mathbf{u},h)\circ\mathscr{H}(x_4,h')\circ\mathscr{H}(x_5,h')$, 5I-PSS; (c) $\mathscr{Y}(\mathbf{u},0)\circ\mathscr{S}(O)$, $5G_0$-PSS.

ternary and disjoint. Both forms are, for full-cycle motions (and out of singular configurations), reciprocal to an ∞-pitch screw parallel to $\mathbf{u}\times\mathbf{u}'$ (i.e. a constraint torque), thus originating a persistent special 5-system. While the direction $\mathbf{u}\times\mathbf{u}'$ changes during motion, the screw system always preserves its *dimension* and *principal pitches*. Hunt [8] described the special properties of these forms already in 1973. His observations are at the basis of the concepts developed in this paper. The form in Fig. 2(c) is the result of the composition of a planar ISS with a spherical one, namely $\mathscr{Y}(\mathbf{u},0)\circ\mathscr{S}(O)$. It is binary and conjoint. A ternary and disjoint form may be obtained, for example, by replacing $\mathscr{S}(O)$ with two 0-pitch screws converging in O. The resulting PSS is always reciprocal to a 0-pitch screw (i.e. a constraint force), thus originating a persistent 5-system of zero pitch (briefly labeled as $5G_0$-PSS). Because of their invariant properties, the mechanical generators of all forms in Fig. 2 have been widely used (mainly, with $h=h'=0$) in both mobility analysis [5,6] and mechanism synthesis [7–9]. For instance, the chains in Figs. 2(a)–(b) played an important role in the design of homokinetic couplings and translational parallel manipulators, whereas the chain in Fig. 2(c) has been used for the design of orientational parallel robots. Further details and more application examples may be found in [7].

Figure 3 shows three more examples of PSSs, less trivial than the ones in Fig. 2. The forms in Figs. 3(a–b) are reciprocal to a screw of pitch $-h$ for full-cycle motions, thus generating 5-systems of the general type. In particular, the form $\mathscr{H}(x,h)\circ\mathscr{Y}(\mathbf{u},h)\circ\mathscr{H}(x',h)$ is ternary and disjoint, whereas the form $\mathscr{C}(x)\circ\mathscr{Y}(\mathbf{u},h)\circ\mathscr{C}(x')$, with $x\perp\mathbf{u}\perp x'$, is ternary and conjoint. Finally, Fig. 3(c) shows a ternary and disjoint form of persistent 4-system of the general type, namely $\mathscr{C}(x_1)\circ\mathscr{R}(x_2)\circ\mathscr{H}(x_3,h_3)$, with the geometric parameters complying with the requisites † in Table 2.

Tables 1 and 2 list *all* possible forms of PSSs of dimension 3 and 4 (all screw systems of dimension 2 are obviously persistent). The derivation of such forms, as well as the proof that the aforementioned lists are *exhaustive*, will be reported in a future paper (together with geometrical details, here omitted due to space limitations).

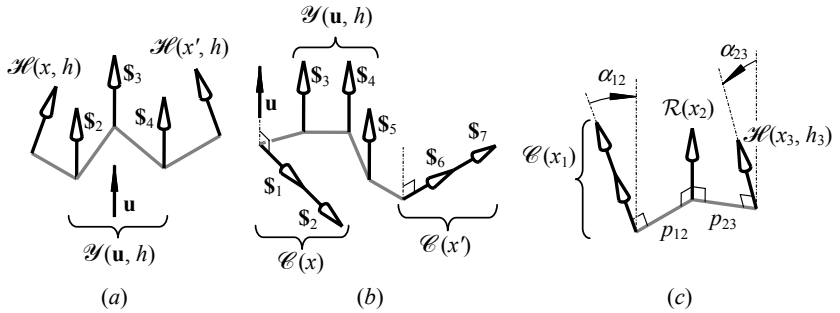


Fig. 3 Examples of PSSs: (a) $\mathscr{H}(x,h)\circ\mathscr{Y}(\mathbf{u},h)\circ\mathscr{H}(x',h)$, 5G-PSS; (b) $\mathscr{C}(x)\circ\mathscr{Y}(\mathbf{u},h)\circ\mathscr{C}(x')$, 5G-PSS; (c) $\mathscr{C}(x_1)\circ\mathscr{R}(x_2)\circ\mathscr{H}(x_3,h_3)$, 4G-PSS.

Table 1 3-dimensional persistent screw systems.

Type	Forms	Requisites	Attributes
3I	$\mathscr{H}(x_1,h_1)\circ\mathscr{R}(x_2)\circ\mathscr{H}(x_3,h_3)$	$*$	Ternary, disjoint
3II	$\mathscr{S}(O)$ – spherical ISS		Invariant
3V	$\mathscr{Y}(\mathbf{u},h)$ – helicoid-planar ISS		Invariant
3VI	$\mathscr{T}_3$ – spatial-traslation ISS		Invariant
3VIII	$\mathscr{C}(x_1)\circ\mathscr{H}(x_2,h_2)$	$x_2 \nparallel x_1$	Binary, disjoint
	$\mathscr{H}(x_1,h_1)\circ\mathscr{P}(\mathbf{s}_2)\circ\mathscr{H}(x_3,h_3)$	$x_3 \nparallel x_1,\ \mathbf{s}_1\times\mathbf{s}_3\cdot\mathbf{s}_2=0$	Ternary, disjoint
3IX	$\mathscr{T}_2(\mathbf{u}_1)\circ\mathscr{H}(x_2,h_2)$	$x_2 \nparallel \mathbf{u}_1, \mathbf{s}_2\cdot\mathbf{u}_1\neq 0$	Binary, disjoint
3X	$\mathscr{C}(x_1)\circ\mathscr{P}(\mathbf{s}_2)$	$\mathbf{s}_2 \nparallel x_1$	Binary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{H}(x_2,h_2)$	$x_2 \parallel x_1$	Binary, disjoint
	$\mathscr{T}_2(\mathbf{u}_1)\circ\mathscr{H}(x_2,h_2)$	$x_2 \perp \mathbf{u}_1$	Binary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{C}(x_2)$	$x_1 \parallel x_2$	Binary, conjoint
	$\mathscr{T}_2(\mathbf{u}_1)\circ\mathscr{C}(x_2)$	$x_2 \perp \mathbf{u}_1$	Binary, conjoint

$* P_{21,2}\equiv P_{23,2},\ h_1\sin^2\alpha_{23}=h_3\sin^2\alpha_{21},\ p_{2i}=h_i\cot\alpha_{2i},\ i=1,3.$

5 Conclusions

This paper presented the concept of *persistent screw systems* (PSSs), which generalize that of *invariant* screw systems. The latter are the subalgebras of the Lie algebra $se(3)$ of the Euclidean group. PSSs are not subalgebras of $se(3)$, but they still exhibit remarkable invariant properties for full-cycle motions, namely the space dimension and the pitch of the principal screws. It has been shown that PSSs may play an important role in both mobility analysis and mechanism synthesis. Sufficient conditions for the generation of PSSs were presented. They provide a valuable tool to easily obtain PSSs of arbitrary dimension. Deriving PSSs that do not comply with such conditions is, on the contrary, a more complex task. The paper listed all PSSs of dimensions 3 and 4, whereas the comprehensive classification of PSSs of dimension 5 is the subject of current research.

Table 2 4-dimensional persistent screw systems.

Type	Forms	Requisites	Attributes
4G	$\mathscr{C}(x_1)\circ\mathscr{R}(x_2)\circ\mathscr{H}(x_3,h_3)$	†	Ternary, disjoint
	$\mathscr{H}(x_1,h_1)\circ\mathscr{R}(x_2)\circ\mathscr{R}(x_3)\circ\mathscr{H}(x_4,h_4)$	‡	Quater., disjoint
4I	$\mathscr{H}(x_1,h_1)\circ\mathscr{S}(O),\ \mathscr{P}(\mathbf{s}_1)\circ\mathscr{S}(O)$		Binary, disjoint
	$\mathscr{R}(x_1)\circ\mathscr{S}(O)$	$O\notin x_1$	Binary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{S}(O)$	$O\in x_1$	Binary, conjoint
	$\mathscr{C}(x_1)\circ\mathscr{R}(x_2)\circ\mathscr{R}(x_3)$	$O\in\{x_1,x_2,x_3\}$	Ternary, disjoint
4II	$\mathscr{H}(x_1,h_1)\circ\mathscr{Y}(\mathbf{u}_2,h_2)$	$x_1\nparallel\mathbf{u}_2$	Binary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{Y}(\mathbf{u}_2,h_2)$	$x_1\perp\mathbf{u}_2$	Binary, conjoint
	$\mathscr{C}(x_1)\circ\mathscr{H}(x_2,h_2)\circ\mathscr{H}(x_3,h_3)$	$x_2\perp x_1,x_3\parallel x_2,h_3=h_2$	Ternary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{H}(x_2,h_2)\circ\mathscr{P}(\mathbf{s}_3)$	$x_2\perp x_1,\mathbf{s}_3\perp x_2$	Ternary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{P}(\mathbf{s}_2)\circ\mathscr{H}(x_3,h_3)$	$\mathbf{s}_2\nparallel x_1,x_3\perp x_1,x_3\perp\mathbf{s}_2$	Ternary, disjoint
4III	$\mathscr{X}(\mathbf{u})$ – Schoenflies ISS		Invariant
4IV	$\mathscr{H}(x_1,h_1)\circ\mathscr{T}_2(\mathbf{u}_2)\circ\mathscr{H}(x_3,h_3)$	$x_1\nparallel x_3,\ (\mathbf{s}_1\times\mathbf{s}_3)\nparallel\mathbf{u}_2$	Ternary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{P}(\mathbf{s}_2)\circ\mathscr{H}(x_3,h_3)$	$\mathbf{s}_2\nparallel x_1\nparallel x_3,\ \mathbf{s}_1\times\mathbf{s}_2\cdot\mathbf{s}_3\neq 0$	Ternary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{T}_2(\mathbf{u}_2)\circ\mathscr{H}(x_3,h_3)$	$x_3\nparallel x_1\perp\mathbf{u}_2,\ (\mathbf{s}_1\times\mathbf{s}_3)\nparallel\mathbf{u}_2$	Ternary, part.-joint
4V	$\mathscr{C}(x_1)\circ\mathscr{C}(x_2)$	$x_1\nparallel x_2$	Binary, disjoint
	$\mathscr{H}(x_1,h_1)\circ\mathscr{T}_2(\mathbf{u}_2)\circ\mathscr{H}(x_3,h_3)$	$x_1\nparallel x_3,\ (\mathbf{s}_1\times\mathbf{s}_3)\parallel\mathbf{u}_2$	Ternary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{P}(\mathbf{s}_2)\circ\mathscr{H}(x_3,h_3)$	$\mathbf{s}_2\nparallel x_1\nparallel x_3,\ \mathbf{s}_1\times\mathbf{s}_2\cdot\mathbf{s}_3=0$	Ternary, disjoint
	$\mathscr{C}(x_1)\circ\mathscr{T}_2(\mathbf{u}_2)\circ\mathscr{H}(x_3,h_3)$	$x_1\nparallel x_3,\ (\mathbf{s}_1\times\mathbf{s}_3)\parallel\mathbf{u}_2$	Ternary, part.-joint
	$\mathscr{C}(x_1)\circ\mathscr{T}_2(\mathbf{u}_2)\circ\mathscr{C}(x_3)$	$x_1\nparallel x_3,\ (\mathbf{s}_1\times\mathbf{s}_3)\parallel\mathbf{u}_2$	Ternary, conjoint

† $P_{12,2}\equiv P_{23,2},\ p_{23}=h_3\cot\alpha_{23},\ p_{12}\cos\alpha_{23}\sin\alpha_{23}=p_{23}\cos\alpha_{12}\sin\alpha_{12}$.
‡ $P_{12,2}\equiv P_{23,2},\ P_{23,3}\equiv P_{34,3},\ p_{ij}=h_j\cot\alpha_{ij},\ p_{23}\cos\alpha_{ij}\sin\alpha_{ij}=p_{ij}\sin\alpha_{23},\ (i,j)=(2,1),(3,4)$.

Acknowledgements The authors thank the Program for the Improvement of University Teaching, by the Mexican Ministry of Public Education, for providing the funds that made it possible for the first author to research at the University of Guanajuato, Mexico.

References

1. Hunt, K.H.: *Kinematic Geometry of Mechanisms*. Clarendon Press, Oxford (1978).
2. Gibson, C.G., Hunt, K.H.: Geometry of screw systems – 1. Screws: Genesis and geometry. *Mech. Mach. Theory*, **25**(1), 1–10 (1990).
3. Gibson, C.G., Hunt, K.H.: Geometry of screw systems – 2. Classification of screw systems. *Mech. Mach. Theory*, **25**(1), 11–27 (1990).
4. Hervé, J.M.: Analyse structurelle des mécanismes par groupe des déplacements. *Mech. Mach. Theory*, **13**(4), 437-450 (1978).
5. Fanghella, P., Galletti, C.: Mobility analysis of single-loop kinematic chains: An algorithmic approach based on displacement groups. *Mech. Mach. Theory*, **29**(8), 1187-1204 (1994).
6. Rico, J.M., Gallardo, J., Ravani, B.: Lie algebra and the mobility of kinematic chains. *J. Rob. Syst.*, **20**(8), 477–499 (2003).
7. Kong, X., Gosselin, C.M.: *Type Synthesis of Parallel Mechanisms*. Springer, Berlin (2007).
8. Hunt, K.H.: Constant-velocity shaft couplings: A general theory. *ASME J. Eng. Ind.*, **95B**(2), 455–464 (1973).
9. Rico, J.M., Cervantes-Sánchez, J.J., Tadeo-Chávez, A., Pérez-Soto, G.I., Rocha-Chavarría, J.: New considerations on the theory of type synthesis of fully parallel platforms. *ASME J. Mech. Des.*, **130**(11), Paper No. 112302 (2008).

Localisation of the Instantaneous Axis of Rotation in Human Joints

B. Bru and V. Pasqui

Institut des Systèmes Intelligents et de Robotique; Université Pierre et Marie Curie, CNRS UMR 7222, 4 Place Jussieu, 75252 Paris Cedex 05, France; e-mail: {bru, pasqui}@isir.fr

Abstract. Many applications require to reconstruct the human movement from data with the highest accuracy. The use of a robust method for the determination of the Instantaneous Axis of Rotation (IAR) is needed for the development of a realistic biomechanics model of Human. Current methods use markers attached to the skin in order to determine the motion kinematic. The present study proposes a new method for the determination of the IAR during human movements. Our specific aim is to provide an easy-to-implement method which works whatever the kind of joint. The presented method was compared with a functional method (SCoRE) by way of simulations then within the context of human movements. Results obtained are satisfactory and could find direct applications for biomechanical modelling and human simulation.

Key words: Human instantaneous axis of rotation, motion capture.

1 Introduction

The location of the Instantaneous Axis of Rotation (IAR) is of primary importance for the reconstruction of human movements. Indeed an accurate determination of the IAR location is required for the development of a realistic biomechanics model, for the determination of inertial parameters, for the success of a total joint prosthesis placement, etc.

Many researchers have published on this subject [1–5]. The functional approach allows a real personalisation of the model. Many methods such as Helical Axis (HA) [4, 5], Symmetrical Center of Rotation Estimation (SCoRE) method [1, 2] have been developed. These methods try to determine the constant location of the Centre of Rotation (CR) with certitude, making the hypothesis that human joints are rotoid or spherical. However some authors have shown that human joints cannot be modelled in such a way [6].

The purpose of this paper is to present a simple and robust experimental method to determine IAR between two human segments, which does not make any assumption about the kind of joint that is modelled. The description of the SCoRE method and the presented method will be detailed in Section 2. A comparison of these two

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_21, © Springer Science+Business Media B.V. 2010

methods will be made in Section 3 by the way of simulation then within the context of human movements in Section 4.

2 Determination of the Instantaneous Axis of Rotation Location

The determination of IAR location is a recurrent subject in biomechanics. However all these methods consider that human joints have a fixed joint centre like mechanical joints such as rotoid or ball joints. The International Society of Biomechanics (ISB) recommends using the HA method for most human joints [3]. Monnet shows that the SCoRE and HA methods generate similar results for angular velocities superior or equal to 0.25 rad/s [1]. However, all these methods model the IAR by a constant point (CR) and a direction vector but, according to Graichen's work [6], such a point does not exist. Taking this remark into account, we have developed a method which locates the IAR without hypothesis on the nature of the joint. Even if the gain of precision is relatively slight in the case of dynamic modelling, it is very important for the use of wearable orthesis. Because the SCoRE method is not sensitive to low angular velocity [1], we have decided to compare the presented method to the SCoRE method. In the first part of this section, the SCoRE method will be succintly presented and in the second part, the method that we present will be detailed.

2.1 The SCoRE Method

The SCoRE method determines a common fixed point between the distal segment and the proximal segment of the joint. This point is the centre of rotation. The position of the joint centre is constant in the frame attached to the proximal segment and in the frame attached to the distal segment, they are respectively noted $\overrightarrow{PC}$ and $\overrightarrow{DC}$. From the matrices $\left[{}^0R_P\right]$ and $\left[{}^0R_D\right]$ which respectively represent the rotation of the frame $\mathscr{R}_P(P,x_P,y_P,z_p)$ and of the frame $\mathscr{R}_D(D,x_p,y_p,z_p)$ in the fixed reference frame $\mathscr{R}_0$, the following equation can be written:

$$\left[\overrightarrow{OP}\right] + \left[{}^0R_P\right]\left[\overrightarrow{PC}\right] = \left[\overrightarrow{OD}\right] + \left[{}^0R_d\right]\left[\overrightarrow{DC}\right] \tag{1}$$

The unknown variables can be determined in a 6×1 matrix, then (1) becomes:

$$\underbrace{\left[\left[{}^0R_d\right]\ \left[-{}^0R_p\right]\right]}_{[A]\ (3\times6)}\underbrace{\begin{bmatrix}\overrightarrow{DC}\\ \overrightarrow{PC}\end{bmatrix}}_{6\times1} = \underbrace{\left[\overrightarrow{OP}-\overrightarrow{OD}\right]}_{[B]\ (3\times1)} \tag{2}$$

Grouping together all the recorded human movement data, the two vectors $\overrightarrow{PC}$ and $\overrightarrow{DC}$ are estimated in a least square sense:

$$\begin{bmatrix} \overrightarrow{DC} \\ \overrightarrow{PC} \end{bmatrix} = ([A]^T [A])^{-1} [A]^T [B]$$

Having $\overrightarrow{PC}$ and $\overrightarrow{DC}$, we can reconstruct the CR coordinates at each time using (1).

2.2 Our Method

Suppose an absolutely rigid body S. There is an axis $\Delta = (M, \overrightarrow{\Omega})$ where the rigid body S turns and translates relatively to a fixed frame R_0 among this axis.

Let M be a point on Δ, then:

$$\overrightarrow{V}_{(M \in S/R_0)} = \lambda . \overrightarrow{\Omega}_{S/R_0} \tag{3}$$

and:

$$\overrightarrow{V}_{(A \in S/R_0)} = \overrightarrow{V}_{(M \in S/R_0)} + \overrightarrow{\Omega}_{S/R_0} \wedge \overrightarrow{MA} \tag{4}$$

Our method consists in determining the location of Δ which represents the IAR. Let us multiply (4) by $\overrightarrow{\Omega}_{S/R_0}$, then (4) becomes:

$$\overrightarrow{\Omega}_{S/R_0} \wedge \overrightarrow{V}_{(A \in S/R_0)} = \overrightarrow{\Omega}_{S/R_0} \wedge (\overrightarrow{\Omega}_{S/R_0} \wedge \overrightarrow{MA}) \tag{5}$$

with: $\overrightarrow{\Omega}_{S/R_0} \wedge \overrightarrow{V}_{M \in S/R_0} = 0$ because $\overrightarrow{\Omega}_{S/R_0}$ and $\overrightarrow{V}_{M \in S/R_0}$ are colinear (3).

Let $\overrightarrow{MA} = \overrightarrow{MA^*} + \overrightarrow{A^*A}$ in (5) where A^* is the orthogonal projection of A on Δ, we obtain:

$$\overrightarrow{A^*A} = \frac{\overrightarrow{V}_{(A \in S/R_0)} \wedge \overrightarrow{\Omega}_{S/R_0}}{\left\| \overrightarrow{\Omega}_{S/R_0} \right\|^2} \tag{6}$$

Let us apply this to the relative movement between two rigid bodies S_1 and S_2. $A_i \in S_1$ and $B_j \in S_2$ are certain fixed points on the system and A_i^* and B_j^* are respectively the orthogonal projections of A_i and B_j points on the IAR between the two bodies as illustrated in Fig. 1. For this example, Equation (6) becomes:

$$\begin{cases} \overrightarrow{A_i A_i^*} = \dfrac{\overrightarrow{V}_{(A_i \in S1/S2)} \wedge \overrightarrow{\Omega}_{S1/S2}}{\left\| \overrightarrow{\Omega}_{S1/S2} \right\|^2} \\ \overrightarrow{B_j B_j^*} = \dfrac{\overrightarrow{V}_{(B_j \in S2/S1)} \wedge \overrightarrow{\Omega}_{S2/S1}}{\left\| \overrightarrow{\Omega}_{S2/S1} \right\|^2} \end{cases} \tag{7}$$

$\overrightarrow{V}_{(A_i \in S1/S2)}$ could be determined using composition of movement:

$$\overrightarrow{V}_{(A_i \in S1/S2)} = \overrightarrow{V}_{(A_i \in S1/R_0)} - \overrightarrow{V}_{(A_i \in S2/R_0)} \tag{8}$$

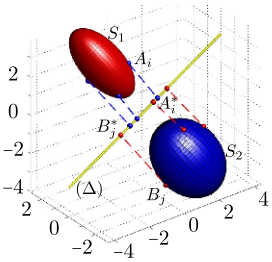


Fig. 1 Relative motion between two rigid bodies S_1 and S_2. A_i and B_j are certain fixed points on S_1 and S_2. The A_i^* and B_j^* points are respectively the orthogonal projections of the A_i and B_j points. (Δ) is defined by the alignment of the A_i^* and B_j^* points.

and property of the rigid body:

$$\overrightarrow{V}_{(A_i \in S2/R_0)} = \overrightarrow{V}_{(B_j \in S2/R_0)} + \overrightarrow{\Omega}_{S2/R_0} \wedge \overrightarrow{B_j A_i} \tag{9}$$

Applying the same computation for $\overrightarrow{V}_{(B_j \in S2/S1)}$, (7) becomes:

$$\begin{cases} \overrightarrow{A_i A_i^*} = \dfrac{(\overrightarrow{V}_{(A_i \in S1/R_0)} - \overrightarrow{V}_{(B_j \in S1/R_0)} - \overrightarrow{\Omega}_{S2/R_0} \wedge \overrightarrow{B_j A_i}) \wedge \overrightarrow{\Omega}_{S1/S2}}{\left\| \overrightarrow{\Omega}_{S1/S2} \right\|^2} \\ \overrightarrow{B_j B_j^*} = \dfrac{(\overrightarrow{V}_{(B_j \in S2/R_0)} - \overrightarrow{V}_{(A_i \in S2/R_0)} - \overrightarrow{\Omega}_{S1/R_0} \wedge \overrightarrow{A_i B_j}) \wedge \overrightarrow{\Omega}_{S2/S1}}{\left\| \overrightarrow{\Omega}_{S2/S1} \right\|^2} \end{cases} \tag{10}$$

We easily obtain the coordinates of each A_i^* and B_j^* points in the reference frame. All A_i^* and B_j^* points should be aligned with each other and let us locate the IAR. Even if the use of one orthogonal point is enough in classical mechanical engineering, we compute the orthogonal projection of each marker. This ensures precision of the IAR location given that data recorded from human movements are noisy.

3 Simulation

In order to check the ability of the proposed method to provide an accurate location of the IAR, it is compared to the SCoRE method in the case of simulations.

3.1 Experimental Protocol

The ability of the proposed method is first evaluated through simulations by considering the case of the motion of two homogeneous rigid bodies linked by a different joint. We have considered two different joints: rotoid and spherical joints. The re-

Table 1 Regression coefficient r^2.

Data	$r^2\,(x,y)$	$r^2\,(x,z)$
Simulation	1 ± 0	1 ± 0
Experiment	0.993 ± 0.006	0.991 ± 0.008

lative motion of these two bodies is simulated on Matlab. In order to check the alignment of the orthogonal points obtained with the method presented in Section 2.2, the coordinates of the A_i^* and B_j^* points are fitted by 3D linear regression using principal component analysis. Regression coefficients r^2 between the x and y coordinates and between x and z are computed. In order to check that these two methods locate the IAR at the same position, the rang of the following $[M]$ matrix is computed.

$$[M] = [A_i^*(t),\, B_j^*(t),\, s(t),\, \overrightarrow{n}(t)] \tag{11}$$

where $s(t)$ and $\overrightarrow{n}(t)$ represent the CR and the direction vector determined with the SCoRE method. If the subspace of this matrix is equal to 1 then both methods locate the IAR at the same position.

3.2 Simulations Results

The 3D linear regression of the orthogonal projection points are resumed in Table 1. As expected in the case of simulation, determination coefficients are equal to 1 and indicate that A_i^* and B_j^* points are totally aligned.

Results of rotoid and spherical joints simulations are presented in Figs. 2 and 3. Results show that both methods (SCoRE and the presented method) have good accuracy and find the same IAR location at all times whatever the kind of joint (rotoid and spherical joint) indeed the matrix $[M]$ is equal to 1 at all times for both joint simulations. All the results show that the presented method is accurate for the computation of the location of the IAR.

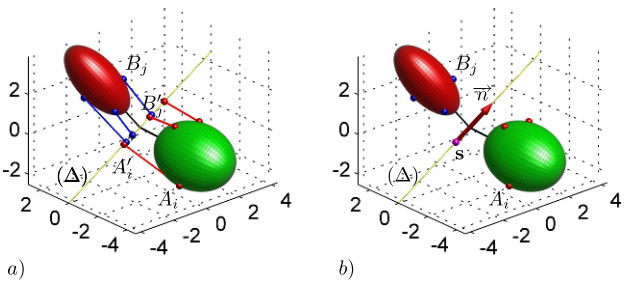


Fig. 2 Comparison of the presented method (a) and SCoRE method (b) during a motion of two rigid bodies linked by a rotoid joint.

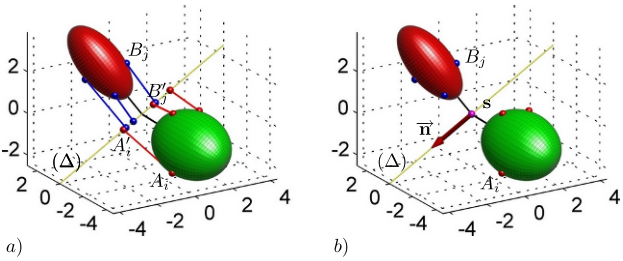


Fig. 3 Comparison of the presented method (a) and SCoRE method (b) during a motion of two rigid bodies linked by a spherical joint.

As these two methods find identical results in the case of simulations, they will be tested and compared within the context of human movements.

4 Experimental Study on Human Subjects

4.1 Protocol and Apparatus

Ten male subjects participated in this study after giving written informed consent. They were instructed to perform three series of ten flexion-extension cycles. Each movement was separated by a 2 s rest period. The recorded movement was 5 s long.

A 3 cx1 units Codamotion system (Charnwood Dynamics Ltd., Rothley, United Kingdom, http://www.codamotion.com) was used to collect kinematic data at 400 Hz. Six active markers were used for determining the position of the IAR of the elbow (3 on the forearm and 3 on the upper arm). In order to reduce the artefacts due to the deformation of the soft tissues intervening between the bone and the markers, active markers were positioned on the Medial Epicondyle, the Lateral Epicondyle and the Deltoid Tuberosity for the upper arm and on the Ulnar Styloid Process, the Base of the Olecranon and the Radial Styloid Process for the forearm [7].

4.2 Data Processing and Modelling

Firstly, kinematic data of the six active markers were filtered at 9 Hz using a 4th-order zero-lag low-pass Butterworth filter. Then, in order to check the constant distance between each marker of the same body segment, a least square algorithm was applied on the 3 active markers of each body segment. Finally the $\overrightarrow{\Omega}$ and the IAR location estimations were computed using respectively the relation $[\Omega] = [R]^t[\dot{R}]$ and the relation (10).

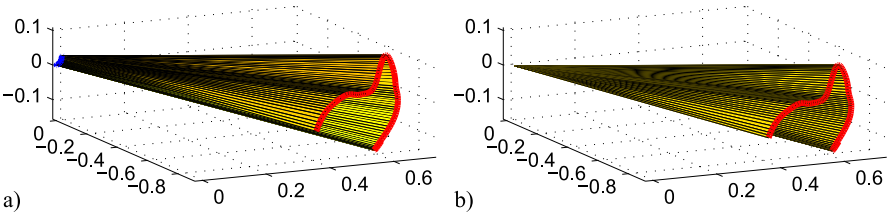


Fig. 4 IAR cartography performed with (a) the presented method and (b) with the SCoRE method.

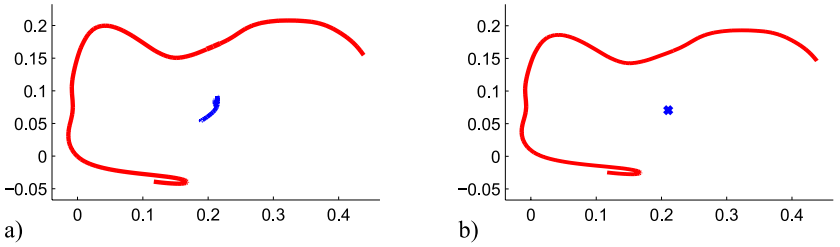


Fig. 5 Cross-sections of the IAR cartography performed with (a) the presented method and (b) the SCoRE method.

4.3 Results

The 3D linear regression of the orthogonal projection points are resumed in Table 1. The very high r^2 and a very low standard deviation: 0.993 ± 0.006 and 0.991 ± 0.008 indicate that A_i^* and B_j^* points can be considered as totally aligned.

The IAR cartography of the elbow of one subject is illustrated in Fig. 4(a) whereby the presented method was used, and in Fig. 4(b) where the SCoRE method was used. These figures represent the evolution of the IAR in a 3D space. Results show that even in the case of a flexion-extension movement (without a pronation-supination movement), the elbow joint cannot be considered as a rotoid joint. Indeed it is clear that the shape of the IAR cartography is a cone (Fig. 4).

Figure 5 represents the smallest (blue line) and the biggest (red line) cross-sections of the IAR cartography performed with the presented method and the SCoRE method, respectively. Concerning the biggest cross-section, its shape is represented by a semi-circle. These results confirmed the results of the previous paragraph, the axis of rotation described during a flexion-extension movement cannot be modelled as a fixed axis. Conversely, as expected, with a SCoRE method the smallest cross-section is represented by a point. In the case of the presented method, the smallest cross-section is not a point but rather a curved line. This result confirms Graichen's work [6]. Human joints cannot be modelled as an ICR.

4.4 Discussion

Figure 4 and Table 1 show that both methods have similar results for the orientation of the IAR however their locations are different. This results proves that the SCoRE method simplify the modelisation of human joints as traditional joints. Figures 2, 3 and 4 show that the presented method is more robust because no hypothesis on the nature of the joint is done and it is not limited to the modelisation of classical joints.

Moreover our method does not require pre-tests with conventional movements as the method SCoRE needs. Our method requires a set of 3 markers per segment whereas at least 5 markers per segment is required for the SCoRE method [1].

5 Conclusions and Future Work

This paper presented an experimental method to determine the IAR location in human joints. The presented method is simpler and more robust than the methods in use today. Simulations and experiments have demonstrated its ease of implementation with low computational requirements and its ease of use with only three markers per segment. No pre-test is needed to use this method and no hypothesis on the nature of joint is required. The method will find direct applications for biomechanical modelling and human simulation. To complete this issue, we will try to generalise the IAR evolution in order to obtain a parametric model of the IAR cartography.

References

1. Monnet, T., Desailly, E., Begon, M., Valle, C., Lacouture, P.: Comparison of the score and HA methods for locating in vivo the glenohumeral joint centre. *J. Biomech.* **40**(15), 3487-3492 (2007).
2. Ehrig, R. M., Taylor, W. R., Duda, G. N., Heller, M. O.: A survey of formal methods for determining the centre of rotation of ball joints. *J. Biomech.* **39**(15), 2798-2809 (2006).
3. Wu, G., van der Helm, F. C. T., DirkJan Veeger, H. E. J., Makhsous, M., Van Roy, P., Anglin, C., Nagels, J., Karduna, A. R., McQuade, K., Wang, X., Werner, F. W., Buchholz, B., International Society of Biomechanics.: ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion – Part II: Shoulder, elbow, wrist and hand. *J. Biomech.* **38**(5), 981-992 (2005).
4. Guillard, G.: Un espace de reprsentation pour ltude conjointe de la morphologie et de la fonctionnalit des surfaces articulaires. PhD Thesis, University of Rennes (2005).
5. Woltring, H. J., Huiskes, R., de Lange, A., Veldpaus, F. E.: Finite centroid and helical axis estimation from noisy landmark measurements in the study of human joint kinematics. *J. Biomech.* **18**(5), 379-389 (1985).
6. Graichen, H., Stammberger, T., Bonel, H., Englmeier, K. H., Reiser, M., Eckstein, F.: Glenohumeral translation during active and passive elevation of the shoulder – A 3D open-MRI study. *J. Biomech.* **33**(5), 609-613 (2000).
7. Van Sint Jan, S.: *Color Atlas of Skeletal Landmark Definitions*. Churchill Livingstone, Elsevier, Philadelphia (2007).

A Kinematic Observation and Conjecture for Creating Stable Constructs of a Peptide Nanoparticle

Z. Shahbazi[1], T.A.P.F. Pimentel[2], H. Ilies[1], K. Kazerounian[1] and P. Burkhard[2]

[1]*Mechanical Engineering Department,* [2]*Institute of Materials Science and Department of Molecular and Cell Biology, University of Connecticut, Storrs, CT, USA; e-mail: z.shahbazi@engr.uconn.edu, tais.pimentel@uconn.edu, ilies@engr.uconn.edu, kazem@engr.uconn.edu, peter.burkhard@uconn.edu*

Abstract. The integration of traditional mechanism theory with biology and medicine at the molecular level enables the use of polypeptide chains to design, manipulate and fabricate biomimetic artificial nano-machines. In this paper we establish a procedure to analyze the mobility of protein molecules based on predicting the formation of hydrogen bonds, which, in turn, is the primary cause for mobility reduction in proteins. We improved our graph-based approach by including side chain and mixed chain hydrogen bond prediction capabilities as well as an energy analysis of these interactions. This method has been emploied to evaluate the internal mobility and identify the rigid and flexible segments of the protein molecules. This computational procedure has been used to assess the stability and mobility of three specific candidate proteins to build a nanobiodevice. Our simulations show that only one of these structures is capable of producing a stable nanoparticle, which is in agreement with the experimental results.

Key words: Protein molecule, hydrogen bond, kinematics, mobility, peptide nanoparticle.

1 Introduction

Researchers are studying existing elements of nature in order to design and fabricate intelligent bio-nano devices for various medical and nanotechnological applications. A novel approach in this field is the use of various biomolecular elements (e.g. protein molecules, DNA strands, etc.) that are capable of generating motion, exerting force, or producing a signal as fundamental machine components. In order to bring these ideals into reality, a comprehensive understanding of these individual biomolecular elements, which serve as building blocks for nanodevices, must be carried out. Therefore, mathematical models and computational simulations play a vital role in gaining an in-depth understanding of the structure, mobility and function of these potential nanodevices.

Proteins are recognized as fundamental components of biological machines. They are built from series of up to 22 different amino acids. Each amino acid has a common structural features, including an α carbon to which an amino group, a carboxyl group, and a variable side chain are bonded. Once the protein chain is held together by peptide bonds, the linked series of amino acids make an open loop

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_22,

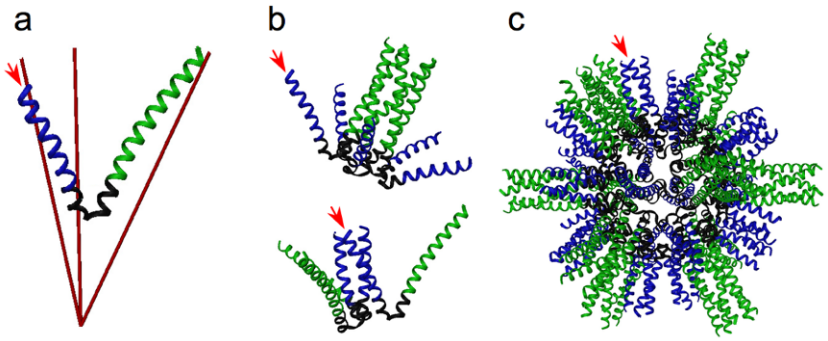


Fig. 1 Core self-assembly domain. A monomer sequence composed of a trimeric coiled-coil (blue), a linker segment (black), and a pentameric coiled-coil (green). The monomer (a), pentamer and trimer (b), and 60mer icosahedron nanoparticle (c), are shown in the same orientation by the red arrow. Residues in the black linker region are close enough to interact with each other. The residues in the blue and green section are too far away from each other to interact.

serial chain. Proteins fold due to interatomic forces and during this process some new bonds between atoms of the non-adjacent residues are created and form the secondary structure of protein molecule [1]. Among the recent bonds, hydrogen bonds are of particular importance due to their key role in determining the three dimensional structure of proteins. Modeling protein molecules as kinematic chains provides the foundation for developing powerful approaches to manipulate matter at the molecular level.

In this paper we improved our newly released flexibility analysis method of protein molecule [2] by including side chain and mixed chain hydrogen bond prediction capabilities as well as an energy analysis of these interactions. Furthermore, we use these hydrogen bond prediction capabilities to predict the rigid regions of a peptide nanoparticles.

The nanoparticle shown in Fig. 1 has been designed and fabricated by the Burkhard research group [3]. The device consists of 60 single self-assembled peptide chains. The peptide nanoparticles have regular icosahedral symmetry, and a diameter of about 20 nm. The synthetic peptide monomer sequence consists of a pentameric coiled-coil domain at the N-terminus joined by a short linker segment to a *de novo* designed minimal trimeric coiled-coil domain at the C-terminus (Fig. 1a). Upon refolding, the monomers self-assemble to form peptide nanoparticles (Figs. 1b and c). The linker segment (shown in black in Fig. 1) is crucial to the orientation of the oligomerization domains and thus for the formation of the nanoparticle. These peptide nanoparticles hold a great promise for their potential utilization in a number of biomedical applications. For example, the repetitive display of antigenic sequences at the nanoparticle surface may potentially be developed into a novel vaccine platform. Recently, they have been functionalized and used as vaccines for SARS [4] and Malaria [5] and as a platform to elicit immune response against poor antigens [6].

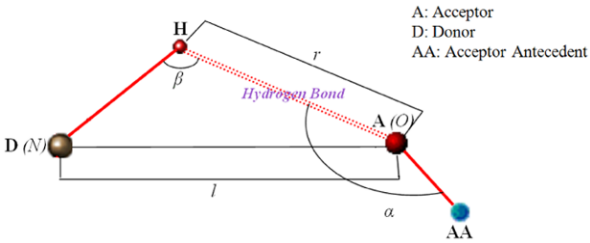


Fig. 2 Geometric parameters.

Depending on the amino acid composition of the linker segment, the flexibility between the two oligomerization domains is predicted to change. If the linker is too flexible, it generates dissimilar primary elements (Fig. 1a), which, instead of a stable homogenous nanoparticle, results in a nonhomogenous nanoparticle or even in some cases an aggregation with no icosahedral structural arrangement. Therefore the best choice to create this nanoparticle is a linker segment with the least flexibility. Of the natural proteins having the desired structure, the mobility of three candidate proteins to assemble into such nanoparticle has been studied and the one predicted through our computational mobility analysis has further been validated using experimental results.

2 Computational Model

2.1 Hydrogen Bond

A hydrogen bond, is the result of an interaction between a donor atom and an acceptor atom. Hydrogen bonds are not as strong as covalent bonds. However these comparatively weak bonds are able to make proteins fold into stable conformations [7]. They transform the open loop linkage into closed kinematic loops. This alteration significantly reduces the mobility of the protein chain. However the resulting internal motion of the protein enables it to perform its function. Different formation criteria have been developed to predict hydrogen bonds occurrence in the protein molecules [2]. In this paper, in order to identify these interactions a set of possible hydrogen bonds has been found based on meeting the suggested geometric criteria in Table 1. The energy threshold has further been used to define which hydrogen bonds are to be included in the mobility analysis.

2.1.1 Geometry of Hydrogen Bonds

The study of protein chain geometry can lead to the prediction of the occurrence of hydrogen bonds since the bonds form between atoms in particular configurations. The relative placement (position and orientation) of the acceptor atom with respect to the donor atom can be rigorously defined through a set of geometric parameters as illustrated in Fig. 2.

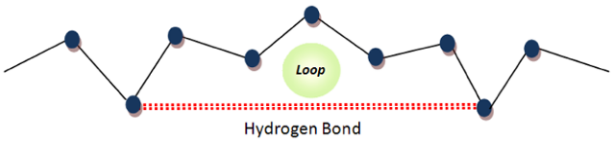


Fig. 3 Loop generated by a hydrogen bond.

Table 1 Geometric criteria.

	α^o	β^o	$r(\text{Å})$	$l(\text{Å})$
α-helices	[110,180]	[110,180]	< 2.5	
β-sheets	[120,180]	[110,180]	< 2.5	
Overall main-chain	[90,180]	[100,180]	< 2.5	< 3.5
Overall main-chain, side-chain and mixed	[90,180]	[90,180]	< 2.5	< 3.5

The existing geometric criteria on the formation of hydrogen bonds in protein molecules have been reviewed in [2]. It has been shown that the criteria reported in Table 1 better correlate the number of predicted hydrogen bonds with those established by biological principles comparing to other existing criteria. The atom positions are required for detecting the potential hydrogen bonds. The initial coordinates of the protein molecules can be retrieved from the Protein Data Bank (PDB). A shortfall of these data files is that the position of hydrogen atoms are usually not recorded. Therefore we used PROTOFOLD, our own protein simulation software [12], to identify the spatial coordinates of all hydrogen atoms. Using the atom coordinate data output by PROTOFOLD, all angular and linear parameters were computed for potential donor and acceptor atoms. Geometric criteria presented in Table 1 have been used to identify the set of possible hydrogen bonds.

2.1.2 Energy of Hydrogen Bonds

A predicted hydrogen bond solely based on geometric constraints is likely to lead us to bonds with high level of energy. When a hydrogen and donor atom are too close to one another, they don't exert any attraction force and do not contribute to stabilization and rigidity of protein molecule. Hence, they can not be considered in the rigidity analysis of protein molecules. Energy analysis of these interactions is therefore required to decide which bond should be included in the study of rigid regions and mobility of protein molecule. The hydrogen bond energy depends on the chemistry of the donor and acceptor atoms as well as their orientation [8–11]. The hydrogen bond energy can be obtained utilizing equation 1 [8]. In this equation: $d_0 = 2.8$ Å and $V_0 = 8$ kcal/mol. The geometric parameters are shown in Fig. 2. The ϕ angle is between the normals of two planes defined by the sp^2 centers.

$$E_{HB} = v0[5(d_0/d)^{12} - 6(d_0/d)^{10})]g(\beta,\alpha,\phi) \quad (1)$$

$$g(\beta,\alpha,\phi) = \cos^2\beta\cos^2(\alpha - 109.5) \qquad for sp^3 donor - sp^3 acceptor$$

$$g(\beta,\alpha,\phi) = \cos^2\beta\cos^2\alpha \qquad for sp^3 donor - sp^2 acceptor$$

$$g(\beta,\alpha,\phi) = \cos^4\beta \qquad for sp^2 donor - sp^3 acceptor$$

$$g(\beta,\alpha,\phi) = \cos^2\beta\cos^2(max[\alpha,\phi]) \qquad for sp^2 donor - sp^2 acceptor$$

Once the energies of all possible hydrogen bonds are calculated, those who have $E_{HB} < 0$ have been labeled as actual hydrogen bonds.

2.2 Mobility Analysis

The identification of hydrogen bonds using geometric criteria presented in Table 1 and energetic criterion in Eq. (1) allows us to analyze the protein mobility. Conceptually, these hydrogen bonds can transform an open loop kinematic chain into a more complex chain that may contain multiple closed sub-chains, some of which can become rigid due to the occurrence of hydrogen bonds. Once we find a hydrogen bond the connectivity matrix has to be updated by replacing two primary links connected by hydrogen bond, by a new rigid link. The next step is converting the reduced connectivity matrix into a graph in order to find the loops. This graph is then parsed to find its cycles because every cycle in the graph will correspond to a loop in the kinematic chain of the protein. Once the graph cycles or independent loops are determined, we use the Grübler-Kutzbach criterion to identify the rigid domains of all the closed loops. If the resulting degree of freedom of a closed loop is zero or less, that particular loop is kinematically over-constrained, and it can be an independent rigid body by itself. When such a loop is detected, we replace all its links with a single link, and we update the connectivity matrix by removing the corresponding rows and columns and by adding one row and one column for the link replacing the loop. This repeats until we find no more over-constrained loops. Figure 3 shows how a hydrogen bond will make a loop within protein structure. Details of this work and some examples are presented in [2].

3 Design and Manipulation of a Peptide Nanoparticle

3.1 Computational Model of the Nanoparticle

Our computational procedure has been applied to the subsets of several proteins, namely 2IC8, 1JIL and 2I88. These protein sequences are three candidates for creating the linker region of the monomer forming the nanoparticle. The primary goal of mobility analysis of these proteins is to recognize which one can make a more stable and uniform nanoparticle. The structural properties of each protein as well as the detected number of hydrogen bonds are given in Table 2. In this table the first column defines the protein PDB code and the residue numbers corresponding

Table 2 Candidate proteins to build nanoparticle.

Protein (PDB code)	Potential Hbonds			Final Hbonds			DOF	Reduced DOF
	M-M	M-S	S-S	M-M	M-S	S-S		
2IC8 (AA147-AA191)	35	1	1	26	1	0	87	11
1JIL (AA293-AA324)	24	2	2	19	0	2	61	9
2I88 (AA345-AA381)	28	2	1	24	2	0	73	7

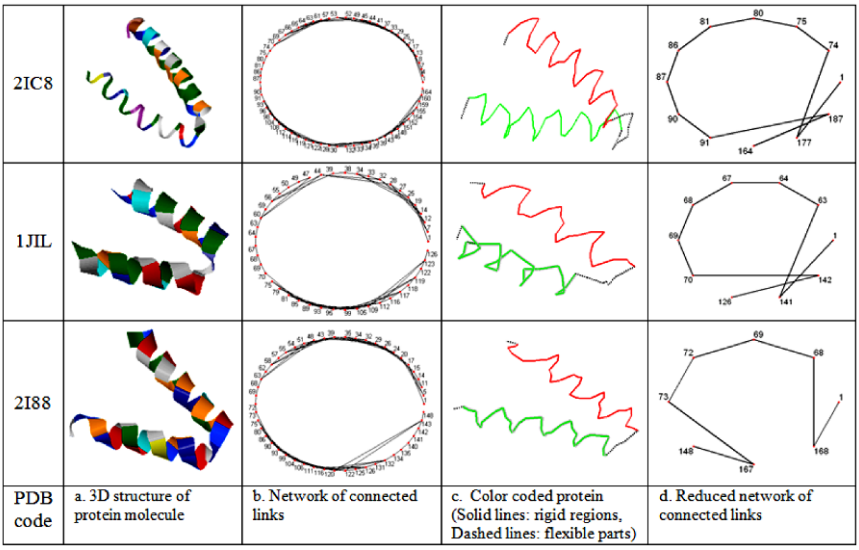


Fig. 4 Mobility analysis of three candidate for nanobiodevice.

to the selected part of protein. The second column shows the potential hydrogen bonds predicted by geometric criteria for main chain - main chain (M-M), main chain - side chain (M-S) and side chain - side chain (S-S) bonds. The third column shows the hydrogen bonds who meet the energy threshold and will be considered in mobility analysis of the protein molecule. The fourth column shows the number of degrees of freedom based on our kinematic model [12]. The number of reduced numbers of degrees of freedom obtained as a result of the approach proposed in this paper are given in the last column of Table 2. It should be observed that this mobility analysis suggests a dramatic change in the original protein models that results in kinematically equivalent protein models. The results are illustrated in Fig. 4. The network of links connected by covalent or hydrogen bonds are shown in the third column. Each of the nodes in these graphs represent a link and the lines show existing bonds between different links. The last column of this figure shows the reduced number of links that results from our analysis. Note the different final DOF of each protein. Our results show that for each protein α helices are rigid (as expected) and they are connected by a flexible linkage in the middle. The only difference is in the

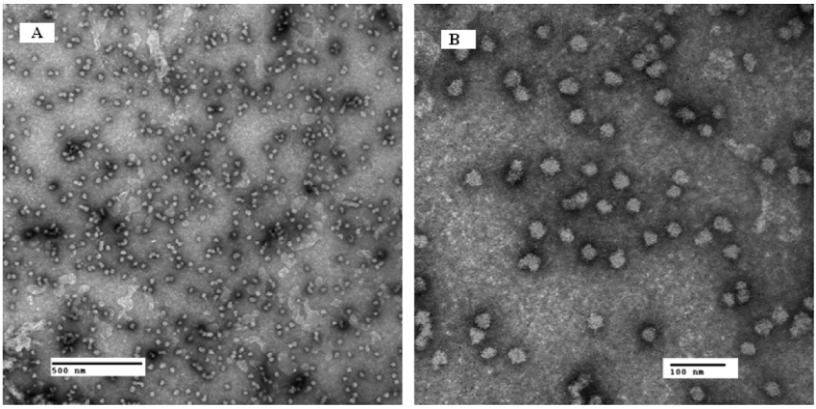


Fig. 5 (a) Transmission electron microscopy image of T2i88 nanoparticle at 49 000X. The sample was negatively stained with 1% uranyl acetate. (b) Same as (a) but at 150 000X.

flexibility of the linker. To get a stable and homogenous particle we need to have less flexibility in the linker. Therefore since 2I88 is more rigid than 2IC8 and 1JIL, it has more chance to self-assemble into a homogenous nanoparticle.

3.2 Experimental Method to Build the Nanoparticle

The recombinant protein T2I88 was produced mainly as previously described [4]. Briefly, the protein was expressed in *Escherichia coli* and purified by metal affinity chromatography. After a multistep dialysis to remove urea the peptide nanoparticles were formed. Transmission Electron Microscopy (TEM) was used to investigate shape and size of the peptide nanoparticles. TEM sample preparation was done as previously described [4] and electron micrographs were captured with a FEI Tecnai Biotwin 120 kV transmission electron microscope equipped with an AMT2k. The same experimental procedure was applied to 2IC8 and 1JIL and as predicted by our model, homogeneous nanoparticles were not obtained. Instead the formation of aggregation was observed (data not shown).

As seen in the electron micrograph (Fig. 5) the peptide chain containing the segment 2I88 in its linker region forms homogeneous nanoparticles. These nanoparticles appeared spherical with a diameter of 36.5 ± 3.5 nm (calculated using ImageJ software). This is in good agreement with computational analysis as this linker segment is more rigid and thus keeps the oligomerizations domains in place.

4 Conclusions

Analysis of a protein structure can indicate rigid and flexible regions likely to undergo conformational changes as part of protein's function. This makes predicting

protein mobility an important class of problems in structural and computational biology. In this paper a novel approach is introduced for characterizing intrinsic mobility of protein molecules that results from the formation of hydrogen bonds in the protein kinematic model. Hydrogen bonds are located using geometric criteria as well as a bond energy threshold. The underlying kinematic model has been implemented in the PROTOFOLD computational platform. This method has been applied to model several potential building blocks and determine their feasibility for creating a specific nanodevice. Our simulations suggest that 2I88 has a more rigid linker than other proteins and that makes it the best candidate to create a homogeneous, stable particle, which is in complete agreement with our experimental results.

Acknowledgements The research work reported here was made possible by grants No. 0733107 and 0856401 of National Science Foundation. The authors thank Dr Marie Cantino and Mr Stephen Daniels for assistance with the transmission electron microscope.

References

1. Branden, C. and Tooze, J.: *Introduction to Protein Structure*, second ed., Garland Publishing (1999).
2. Shahbazi, Z. Ilies, H. and Kazerounian, K.: Hydrogen bonds and kinematic mobility of protein molecules. *Mechanisms and Robotics Journal* (in press) (2009).
3. Raman, S.K., Machaidze, G., Lustig, A., Aebi, U. and Burkhard, P.: Structure-based design of peptides that self-assemble into regular polyhedral nanoparticles. *Nanomedicine* **2**, 95–102 (2006).
4. Pimentel, T.A.P.F., Yan, Z., Jeffers, S.A., Holmes, K.V., Hodges, R.S. and Burkhard, P.: Peptide nanoparticles as novel immunogens: Design and analysis of a prototypic severe acute respiratory syndrome vaccine. *Chem. Biol. Drug Des.* **73**, 53–61 (2009).
5. Kaba, S.A., Brando, C., Guo, Q., Mittelholzer, C., Raman, S., Tropel, D., Aebi, U., Burkhard, P. and Lanar, D.E.: A non-adjuvanted polypeptide nanoparticle vaccine confers long-lasting protection against rodent malaria. *J. Immunol.* **183**, 7268–7277 (2009).
6. Schroeder, U., Graff, A., Buchmeier, S., Rigler, P., Silvan, U., Tropel, D., Jockusch, B.M., Aebi, U., Burkhard, P. and Schoenenberger, C.A.: Peptide nanoparticles serve as powerful platform for the immunogenic display of poorly antigenic actin determinants. *J. Mol. Bio.* **389**, 1368–1381 (2009).
7. Wales, T.E. and Fitzgerald, M.C.: The energetic contribution of backbone–backbone hydrogen bonds to the thermodynamic stability of a hyperstable P22 Arc repressor mutant. *J. Am. Chem. Soc.* **123**, 7709–7710 (2001).
8. Dahiyat, B.I., Gordon, D.B. and Mayo, S.L.: Automated design of the surface positions of protein helices. *Protein Sci.* **6**, 1333–1337 (1997).
9. Choi, H., Kang, H. and Park, H.: New angle-dependent potential energy function for backbone-backbone hydrogen bond in protein-protein interactions. *J. Comput. Chem.* (2009).
10. Favrin, G., Irback, A. and Mohanty, S.: Oligomerization of amyloid Abeta16-22 peptides using hydrogen bonds and hydrophobicity forces. *Biophys. J.* **87**, 3657–3664 (2004).
11. Greathouse, J.A., Durkin, G.S., Larentzos, J.P. and Cygan, R.T.: Implementation of a Morse potential to model hydroxyl behavior in phyllosilicates. *J. Chem. Phys.* **130**, 134713 (2009).
12. Kazerounian, K., Latif, K. and Alvarado, C.: Protofold: A successive kinetostatic compliance method for protein conformation prediction. *J. Mech. Des.* **127**, 712–717 (2005).

Forward Kinematic Problem of 5-PRUR Parallel Mechanisms Using Study Parameters

Mehdi Tale Masouleh[1], Manfred Husty[2] and Clément Gosselin[3]

[1,3]*Département de Génie Mécanique Université Laval, Québec, QC, G1V 0A6, Canada; e-mail: mehdi.tale-masouleh.1@ulaval.ca, gosselin@gmc.ulaval.ca*
[2]*Unit Geometry and CAD, Institute for Basic Sciences in Engineering, University of Innsbruck, Technikerstraße 13, A-6020 Innsbruck, Austria; e-mail: manfred.husty@uibk.ac.at*

Abstract. This paper investigates the forward kinematic problem of five-degree-of-freedom parallel mechanisms generating the 3T2R motion containing five identical legs of the PRUR type. The general mechanism originates from the type synthesis performed for symmetrical 5-DOF parallel mechanisms. The kinematic modeling of the mechanism is addressed using the so-called Study parameters and Gröbner bases and the final system of equations is solved using homotopy continuation. Finally, it is shown that the mechanism may have more real solutions than the 6-DOF Stewart–Gough platform.

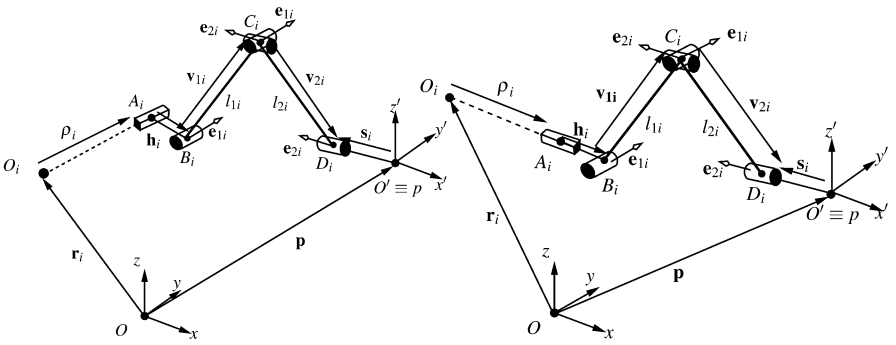
Key words: Symmetrical 5-DOF parallel mechanisms, 3T2R motion pattern, Stewart–Gough platform, forward kinematic problem (FKP), Study parameters, Gröbner base, homotopy continuation.

1 Introduction

Five-degree-of-freedom (5-DOF) parallel mechanisms are a class of parallel mechanisms with reduced degrees of freedom which, according to their mobility, fall into three classes: (1) three translational and two rotational freedoms (3T2R), (2) three rotational and two planar translational freedoms ($3R2T_p$) and (3) three rotational and two spherical translational freedoms ($3R2T_s$) [1]. Geometrically, the 3T2R motion can be made equivalent to guiding a combination of a directed line and a point on it. Accordingly, the 3T2R mechanisms can be used in a wide range of applications for a *point-line* combination including, among others, 5-axis machine tools, welding, conical spray-gun.

Recently, the type synthesis of symmetrical 5-DOF parallel mechanisms has been revised [1–5] and some new architectures have been proposed. In the context of this paper, the term symmetric refers to the limb type and not to the geometry, such as centro-symmetrical simplifications. It is worth noticing that most existing 5-DOF parallel manipulators are built using a 5-DOF passive leg which constrains some actuated 6-DOF limbs [6].

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_23, © Springer Science+Business Media B.V. 2010



(a) $\underline{\text{C}}\text{UR} \equiv \Gamma = 1$ (b) $\underline{\text{P}}\text{RUR} \equiv \Gamma = 0$

Fig. 1 Kinematic representation of two possible arrangements for a $\underline{\text{P}}$RUR limb.

To the best knowledge of the authors, up to now, very few kinematic studies have been conducted on symmetrical 5-DOF parallel mechanisms and researchers have mainly worked on the type synthesis. This is probably due to their short history. Thus, in this research, the FKP of symmetrical 5-DOF parallel mechanisms, more precisely 5-$\underline{\text{P}}$RUR, is addressed which can be regarded as one of the most challenging topics in the kinematics of parallel mechanisms. In this notation, $\underline{\text{P}}$ stands for an actuated prismatic joint, R for a revolute joint and U for a universal joint.

In this paper, instead of the Cartesian coordinates and the Euler angles, the FKP is explored in a 7-dimensional projective space, P^7, called *kinematic image space*. In this approach, the kinematic modeling of the mechanism is based on the so-called *Study parameters* that map displacements of the three-dimensional Euclidean space to points on a quadric, called the *Study quadric*, in a seven dimensional projective space.

The remainder of this paper is organized as follows. The architecture and the general kinematic properties of the 5-DOF parallel mechanism which originated from the type synthesis performed in [1,5] are broadly outlined. The kinematic modeling of the mechanism is explored by means of Study's kinematic mapping. Then the system of equations for the FKP in terms of Study's parameters is solved via a *homotopy continuation* algorithm. Finally, the mapping between Study's parameters and the three-dimensional kinematic space is addressed.

2 Architecture

Figures 1 and 2 provide respectively representations of two possible arrangements for a $\underline{\text{P}}$RUR limb and a CAD model for a 5-DOF parallel mechanism, called *Pentapteron* [7], that can be used to produce all three translational DOFs, plus two independent rotational DOFs (3T2R) of the end-effector, namely (x, y, z, ϕ, θ).

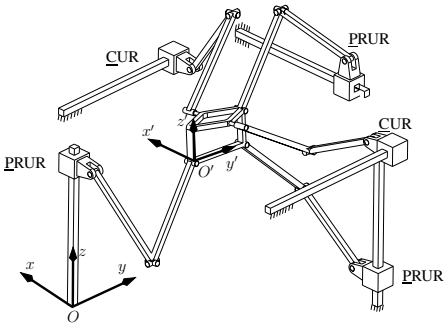


Fig. 2 Schematic representation of a 5-PRUR parallel mechanism called *Pentapteron*.

Throughout this paper $i = 1, \ldots, 5$, unless otherwise specified. In the latter notation, (x, y, z) represent the translational DOFs with respect of the fixed frame O, illustrated in Fig. 1, and (ϕ, θ) stand respectively for the orientation DOFs around axes x and y, respectively. The rotation from the fixed frame O_{xyz} to the moving frame $O'_{x'y'z'}$ is defined as follows: a first rotation of angle ϕ is performed around the x-axis followed by the second rotation about the y-axis by angle θ. The Pentapteron is an orthogonal 5-DOF parallel mechanism arising from the type synthesis presented in [1, 5] and consists of 5 legs of the PRUR type linking the base to a common platform. Pentapteron was first revealed in [7] where the corresponding kinematic arrangement of the limbs is fully described. In addition, the axes of the first R joints in all the legs are arranged to be parallel to the direction of a group of two of the linear actuated joints. Therefore, two types of kinematic arrangements are possible, as depicted in Fig. 1, for the legs: (a) the parallel type, Fig. 1(a), and the perpendicular type, Fig. 1(b). In fact, $\Gamma = 0$ and $\Gamma = 1$ differ in some kinematic properties such as constant-orientation workspace and IKP formulation. In a 5-PRUR parallel mechanism, the axes of all the R joints are always parallel to a plane defined by its normal vector $\mathbf{e}_3 = \mathbf{e}_1 \times \mathbf{e}_2$ where $\mathbf{e}_1$ and $\mathbf{e}_2$ are unit vectors defining the direction of R joints. From *screw theory*, it follows that the mechanism has no possibility to perform a rotation about an axis which is orthogonal to a plane spanned by $[\mathbf{e}_1, \mathbf{e}_2]$.

3 FKP Formulation Using Study's Kinematic Mapping

3.1 Kinematic Modeling of the Principal Limb

Study's kinematic mapping is a mapping of an element γ of the Euclidean displacement group SE(3) into a 7-dimensional projective space, P^7 [8]. The homogeneous coordinate vector of a point in P^7 is $X = [x_0 : x_1 : x_2 : x_3 : y_0 : y_1 : y_2 : y_3]^T$. The kinematic pre-image of X is the displacement α described by the transformation matrix:

$$\Omega = \frac{1}{\Delta}\begin{bmatrix} \Delta & 0 & 0 & 0 \\ p & x_0^2+x_1^2-x_2^2-x_3^2 & 2(x_1x_2-x_0x_3) & 2(x_1x_3+x_0x_2) \\ q & 2(x_1x_2+x_0x_3) & x_0^2-x_1^2+x_2^2-x_3^2 & 2(x_2x_3-x_0x_1) \\ r & 2(x_1x_3-x_0x_2) & 2(x_2x_3+x_0x_1) & x_0^2-x_1^2-x_2^2+x_3^2 \end{bmatrix} \quad (1)$$

$$p = 2(-x_0y_1+x_1y_0-x_2y_3+x_3y_2), q = 2(-x_0y_2+x_1y_3+x_2y_0-x_3y_1),$$
$$r = 2(-x_0y_3-x_1y_2+x_2y_1+x_3y_0),$$

and $\Delta = x_0^2+x_1^2+x_2^2+x_3^2$. Note that the lower right three by three sub-matrix is a proper orthogonal matrix if

$$x_0y_0+x_1y_1+x_2y_2+x_3y_3 = 0 \quad (2)$$

and not all x_i are zero. If these conditions are fulfilled then $[x_0 : \cdots : y_3]^T$ are called *Study parameters* of the displacement α. Equation (2) defines a quadric $S_6^2 \subset P^7$ and the range of the kinematic mapping $\varkappa$ is this quadric minus the three dimensional subspace defined by $E: x_0 = x_1 = x_2 = x_3 = 0$. In the latter, $S_6^2$is called *Study quadric* and E is the *exceptional* or *absolute generator*. One can normalize the parameters such that $\Delta = 1$, then the coordinate x_0 represents the cosine of the half rotation angle. Note that there are other possibilities to normalize, but these are not used within this paper.

Now, we turn our attention to the kinematic modeling of a PRUR limb whose fixed and mobile frames are respectively attached to the first and the last R joint based on the Denavit–Hartenberg (D–H) convention. This limb is referred to as the *principal limb*. All quantities related to the principal limb are noted with p as an index. In this case, the first step toward representing the principal limb kinematic model using Study parameters is to define the D–H parameters and the corresponding transformation matrices of this limb . One can readily obtain the D–H parameters of the principal limb as: $a_{i=\{1,2,3\}} = \{l_{1p}, 0, l_{2p}\}, d_{i=\{1,2,3\}} = 0, \alpha_{i=\{1,2,3\}} = \{0, \frac{\pi}{2}, 0\}$. Applying the D–H convention the two following transformations matrices are obtained:

$$\Sigma_{i=\{1,2,3,4\}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos u_i & -\sin u_i & 0 \\ 0 & \sin u_i & \cos u_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \Gamma_{i=\{1,2,3\}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ a_i & 1 & 0 & 0 \\ 0 & 0 & \cos\alpha_i & -\sin\alpha_i \\ d_i & 0 & \sin\alpha_i & \cos\alpha_i \end{bmatrix} \quad (3)$$

where u_i is the ith joint coordinate. Thus the FKP of the principal limb, F_p, with respect to joint variables, $v_i = \tan\left(\frac{u_i}{2}\right)$, $k = 1, \ldots, 4$, and the design variables can be expressed as follows:

$$\mathbf{F}_p = \left(\prod_{k=1}^{3} \Sigma_k \Gamma_k\right) \Sigma_4. \quad (4)$$

Reaching this step, the so-called *kinematic mapping* should be applied to $\mathbf{F}_p = [f_{ik}]_{i,k=1,\ldots,4}$ which amounts to computing the Study parameters, $X = (x_0 : x_1 : x_2 : x_3 : y_0 : y_1 : y_2 : y_3)$, from matrix $\mathbf{F}_p$. For that, Study in [9] presented a singular-free procedure in which the homogeneous quadruple $\mathbf{x} = (x_0 : x_1 : x_2 : x_3)$ can be obtained

from at least one set of four proportions [8]. Here, among the four proportions the following proportion is chosen:

$$x_0 : x_1 : x_2 : x_3 = 1 + f_{22} + f_{33} + f_{44} : f_{43} - f_{34} : f_{24} - f_{42} : f_{32} - f_{23}. \tag{5}$$

Then, the four remaining Study parameters $\mathbf{y} = (y_0 : y_1 : y_2 : y_3)$ can be computed from:

$$\begin{aligned} 2y_0 &= \ \ f_{21}x_1 + f_{31}x_2 + f_{41}x_3, \quad 2y_1 = -f_{21}x_0 + f_{41}x_2 - f_{31}x_3 \\ 2y_2 &= -f_{31}x_0 - f_{41}x_1 + f_{21}x_3, \quad 2y_3 = -f_{41}x_0 + f_{31}x_1 - f_{21}x_2. \end{aligned} \tag{6}$$

Upon substituting the components of $\mathbf{F}_p$ into Eq. (5) and (6), it turns out that all the computed Study parameters share a common factor which is:

$$\frac{1 + v_1v_2v_3v_4 - v_3v_4 - v_2v_4 - v_1v_4 - v_1v_2 - v_1v_3 - v_2v_3}{\left(1 + {v_1}^2\right)\left(1 + {v_2}^2\right)\left(1 + {v_3}^2\right)\left(1 + {v_4}^2\right)}. \tag{7}$$

Using the fact that Study parameters are homogeneous allows to omit this common factor and one may obtain the following for the rotational part of the Study parameters:

$$x_0 = 2(1 + v_1v_2v_3v_4 - v_3v_4 - v_2v_4 - v_1v_4 - v_1v_2 - v_1v_3 - v_2v_3) \tag{8}$$
$$x_1 = 2(1 + v_1v_2v_3v_4 + v_1v_3 - v_1v_2 + v_1v_4 + v_2v_3 + v_2v_4 - v_3v_4) \tag{9}$$
$$x_2 = 2(v_1v_2v_4 + v_1v_2v_3 + v_1 - v_1v_3v_4 + v_2 - v_3 - v_4 - v_2v_3v_4) \tag{10}$$
$$x_3 = 2(v_1v_2v_4 - v_1v_2v_3 + v_1 - v_1v_3v_4 + v_2 + v_3 + v_4 - v_2v_3v_4). \tag{11}$$

The objective now is to combine the above expressions for $\mathbf{x}$ and $\mathbf{y}$ by eliminating passive variables, i.e., v_i, in order to obtain two independent expressions – since for a fixed P the limb has two free-DOF – for the FKP of one limb with respect to Study's parameters. Before proceeding to the elimination of the passive variables, matrix $\mathbf{F}_p$ is expanded and it reveals that $f_{44} = 0$ which amounts to $\Omega_{44} = 0$:

$$\mathscr{C} = x_0^2 - x_1^2 - x_2^2 + x_3^2 = 0. \tag{12}$$

Thus, before any elimination, a quadric, called *constraint equation* $\mathscr{C}$, is found which is free of any design parameters and can be regarded as the first constraint equation. This leads to deduce that five PRUR limbs share a common rotational constraint as confirmed by the type synthesis performed for such a mechanism in [1]: *The mechanism has a rotational constraint about an axis defined by* $\mathbf{e}_1 \times \mathbf{e}_2$. Equation (12) is a statement equivalent to the latter written in terms of Study's parameters and indicates that the rotational constraint of this mechanism lies on a quadric. The existence of this quadric together with the homogeneous condition, $\Delta = 1$, results in two independent rotational parameters for the mechanism. Returning now to the FKP and Eqs. (8-11), skipping mathematical details, upon applying a step by step resultant method to Eqs. (8-11) and eliminating successively v_3, v_2, v_4 and v_1, leads to three sixth degree polynomials, $\{\mathscr{T}_1, \mathscr{T}_2, \mathscr{T}_3\}$, with respect to the Study and

design parameters. Expressions for $\{\mathscr{T}_1, \mathscr{T}_2, \mathscr{T}_3\}$ are rather long and are not presented. In summary, the FKP of the principal limb can be defined by the following *ideal*:

$$I = < \mathscr{T}_1, \mathscr{T}_2, \mathscr{T}_3, S_2^6, \mathscr{C} > \tag{13}$$

i.e., three sixth degree polynomials and two quadrics S_2^6 and $\mathscr{C}$, the Study and the constraint quadric, respectively. The above formulation channels us to apply a *Gröbner base algorithm* [10], a powerful algebraic geometry algorithm, to the ideal I. From an algebraic geometry stand point, Gröbner bases can be viewed as an organized way to generate new polynomials in the ideal selecting a subset that retains exactly the same solution set as the original polynomials, and determining a valid set of monomial identities that complete the definition of an eigenvalue problem [10]. Upon computing the Gröbner basis of the ideal I with respect to the monomial order $x_0 \prec x_1 \prec x_2 \prec x_3 \prec y_0 \prec y_1 \prec y_2 \prec y_3$, and after several mathematical manipulations, the following expression is found:

$$\begin{aligned}\mathscr{F}_p(X) = &-8(l_{1p}^2 + l_{2p}^2)\left(y_1 x_1 + y_2 x_2\right)^2 + 8(l_{1p}^2 - l_{2p}^2)\left(y_1 x_2 - y_2 x_1\right)\left(-x_0 y_3 + y_0 x_3\right) \\ &+ 16\left({y_2}^2 + {y_1}^2\right)\left({y_0}^2 + {y_3}^2\right) + (l_{1p}^2 - l_{2p}^2)^2\left({x_2}^2 + {x_1}^2\right)^2 = 0.\end{aligned} \tag{14}$$

Considering that both moving links have equal lengths, $l_{1p} = l_{2p} = l_p$, leads to:

$$\mathscr{F}_p(X) = (y_2^2 + y_1^2)(y_0^2 + y_3^2) - l_p^2(y_1 x_1 + y_2 x_2)^2 = 0. \tag{15}$$

Equation (14) is of degree four, instead of six for $\mathscr{T}_{i=\{1,2,3\}}$, and could be regarded as the simplest expression describing the FKP of the principal limb in terms of the Study parameters, X. Thus, reaching this step, two expressions are in place, Eq. (14), or Eq. (15) when $l_{1i} = l_{2i}$, and Eq. (12), which fully constrain the mechanism in space.

3.2 FKP Formulation of the 5-PRUR, Extension from the Principal Limb

In order to find the FKP corresponding to four other limbs, $j = 2, \ldots, 5$, a transformation in the Study parameters X should be made in both *base* and *moving* frame and can be done by following the procedure described in [8]. Consider $\mathbf{b}_j = (b_{0j} : \ldots : b_{7j})$ and $\mathbf{m}_j = (m_{0j} : \ldots : m_{7j})$ as the Study parameters describing the j^{th} limb placement in the base (fixed frame) and in the moving platform (mobile frame), respectively, which are attached to the principal limb. It should be noted that depending on the direction of the prismatic axis, the input parameter for the FKP for the P appears within different components of $\mathbf{b}_i$. Since the axes of the R joints fixed to the base, $\mathbf{e}_1$, are all parallel, $\mathbf{b}_j$ consists of a pure translation which requires to have $b_{0j} = b_{1j} = b_{2j} = b_{3j} = b_{4j} = 0$ for its Study parameters. The same follows for the axes of the R joints attached to the platform, $\mathbf{e}_2$, and leads

to: $m_{0j} = m_{1j} = m_{2j} = m_{3j} = m_{4j} = 0$. Thus, based on the transformation matrices presented in [8], one may obtain:

$$ {}^{j}\mathbf{T}_b = \begin{bmatrix} \mathbf{I}_{4\times4} & \mathbf{0}_{4\times4} \\ \mathscr{B}_j & \mathbf{I}_{4\times4} \end{bmatrix}, \quad {}^{j}\mathbf{T}_m = \begin{bmatrix} \mathbf{I}_{4\times4} & \mathbf{0}_{4\times4} \\ \mathscr{M}_j & \mathbf{I}_{4\times4} \end{bmatrix} \tag{16} $$

so that $\mathscr{B}_j$ and $\mathscr{M}_j$ are the following skew-symmetric matrices:

$$ \mathscr{B}_j = \begin{bmatrix} 0 & -b_{5j} & -b_{6j} & -b_{7j} \\ b_{5j} & 0 & -b_{7j} & b_{6j} \\ b_{6j} & b_{7j} & 0 & -b_{5j} \\ b_{7j} & -b_{6j} & b_{5j} & 0 \end{bmatrix}, \quad \mathscr{M}_j = \begin{bmatrix} 0 & -m_{5j} & -m_{6j} & -m_{7j} \\ m_{5j} & 0 & m_{7j} & -m_{6j} \\ m_{6j} & -m_{7j} & 0 & m_{5j} \\ m_{7j} & m_{6j} & -m_{5j} & 0 \end{bmatrix}. \tag{17} $$

Thus the transformation becomes:

$$ X_j = ({}^{j}\mathbf{T}_m \, {}^{j}\mathbf{T}_b)^{-1} X \tag{18} $$

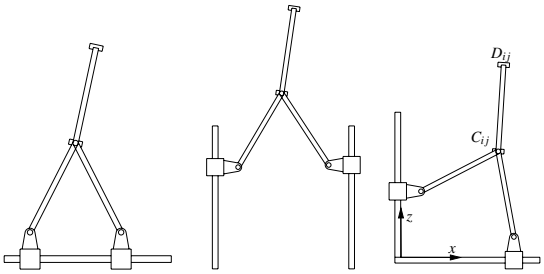
and the FKP of the j^{th} limb, $\mathscr{F}_j$, can be found by replacing X by X_j and the parameters corresponding to the principal limb, p, by the one of the j^{th} limb:

$$ \mathscr{F}_j = \mathscr{F}_p \left(X \longmapsto X_j, p \longmapsto j \right) \tag{19} $$

From Eq. (18) it follows that the Study rotational parameters, $(x_0 : x_1 : x_2 : x_3)$, remain the same for all limbs. In summary, the FKP for the mechanism is equivalent to 8 equations in 8 unknowns:

$$ \mathscr{G} = \left\langle \mathscr{F}_p, \mathscr{F}_2, \ldots, \mathscr{F}_5, S_2^6, \mathscr{C}, \Delta - 1 \right\rangle. \tag{20} $$

For the FKP, the above system of equations should be solved for X with design values of: $\{\mathbf{b}_j, \mathbf{m}_j, l_{1j}, l_{2j}, l_{1p}, l_{2p}\}$. It should be noted that the input of the mechanism, i.e., the elongation of the prismatic actuators are included in $\mathbf{b}_j$. Although the degree of the FKP expressions is reduced from 6 to 4, $\mathscr{T}_{i=1,2,3}$ versus $\mathscr{F}_p$, the degree is still too high to apply the resultant method to $\mathscr{G}$ in order to find a univariate expression for the FKP. This channels us to use numerical algebraic geometry which consists of the intersection of algebraic geometry and numerical analysis. A number of numerical algebraic algorithms have been implemented in *Bertini* [11]. *Bertini* is a software for solving polynomial systems using the *Homotopy Continuation* approach. Using Bertini, numerous random examples in solving the set of Eqs. (20) reveal that the FKP of this 5-DOF parallel mechanism admits up to 1680 finite solutions (complex and real). Applying the so-called witness set method, Bertini has the ability to detect positive dimensional solution sets. Using this feature of Bertini, it can be confirmed that regardless of the inputs and geometric parameters of the mechanism the following complex sets are always a solution of Eq. (20):



(a) Two $\underline{\mathrm{P}} \parallel x$ (b) Two $\underline{\mathrm{P}} \parallel z$ (c) $\underline{\mathrm{P}} \parallel x$+$\underline{\mathrm{P}} \parallel z$

Fig. 3 Simplified kinematic arrangements with merged U joints.

$$\begin{array}{ll} \{x_0 = 0, x_3 = 0, x_1 = -ix_2, y_1 = -iy_2\}, & \{x_0 = 0, x_3 = 0, x_1 = \ ix_2, \ y_1 = \ iy_2\} \\ \{x_1 = 0, x_2 = 0, x_0 = -ix_3, y_0 = -iy_3\}, & \{x_1 = 0, x_2 = 0, x_0 = \ ix_3, y_0 = \ iy_3\}. \end{array} \tag{21}$$

As the equations are linear, the dimension formula shows directly that each set represents a three dimensional complex space. In fact, 1680 is not the *lowest* upper bound for the number of real solutions. By numerous random examples, a maximum of 208 real solutions were found. Providing an upper bound for number of the real solutions requires the development of a univariate expression for the FKP which is an extremely complicated task. It was generally believed that the 6-DOF Gough–Stewart platform with up to 40 solutions for the FKP is the parallel mechanism which possesses the largest number of real solutions for its FKP. Obtaining 208 real solutions leads to conclude that arriving at a univariate expression for the FKP of this parallel mechanism should be extremely difficult especially considering the complexity of deriving a univariate expression for the forty solutions of the FKP of the Gough–Stewart platform [12]. For a simplified design having a pair of combined attachment points, Fig. (3), the FKP is explored using the three-dimensional kinematic space, i.e., Cartesian coordinates, which is beyond the scope of this paper. The latter study revealed that the univariate expression for the FKP can be expressed in terms of angle θ and is of degree 220. Using Bertini for this simplified design confirms that the mechanism has up to 220 finite solutions, i.e., complex and real. Finally, in order to ensure the validity of the solutions obtained by Bertini, all the real solutions are substituted into the IKP expressed in terms of the three-dimensional kinematic space. This test reveals that for all solutions converted to Cartesian coordinates and Euler angles the same set of prismatic actuator elongations can be found. The transformation of Study parameters into Cartesian coordinates and Euler angles is elaborated in the last section of this paper.

4 Cartesian Representation of Study Parameters

The first step is to compute the rotational DOFs (ϕ, θ). To this end the lower three by three sub matrix of Ω, noted as Ω_t, should be made equivalent to $\mathbf{Q}$, i.e., a one by one component correspondence. Skipping mathematical derivations, the inspection of the components of $\mathbf{Q}$ and those of Ω_t' leads to a unique solution for θ and ϕ:

$$\theta = \arctan2(x_1x_3 + x_0x_2,\ x_2x_3 - x_0x_1), \tag{22}$$

$$\phi = \arctan2(x_2x_3 + x_0x_1,\ x_1x_3 - x_0x_2) \tag{23}$$

To compute the position of the platform, (x, y, z), for a given set of Study parameters, $(x_0 : x_1 : x_2 : x_3 : y_0 : y_1 : y_2 : y_3)$, one should rewrite Eq. (6) as follows:

$$\begin{aligned} 2y_0 &= x_1x + x_2y + x_3z, & 2y_1 &= -x_0x + x_2z - x_3y \\ 2y_2 &= -x_0y - x_1z + x_3x & 2y_3 &= -x_0z + x_1y - x_2x. \end{aligned} \tag{24}$$

The above system of equations is of rank three, i.e., the equations are linearly dependent. Thus taking three equations from the above leads to obtaining a unique set of solutions for (x, y, z). Then, the position of the platform, $\mathbf{p}$, with respect of the base frame presented in Fig. 1 becomes:$\mathbf{p} = [y, z, x]^T$. Following the same procedure, one can transform the vectors describing the geometry of the base and platform, written in terms of Study's parameters, $\mathbf{b}_i$ and $\mathbf{m}_i$, respectively, into the vectors describing them in the Cartesian coordinates, $\mathbf{r}_i$ and $\mathbf{s}_i$. Skipping the mathematical derivations one obtains:

$$\mathbf{r}_i = [-2b_{6i},\ -2b_{7i},\ -2b_{5i}]^T, \qquad \mathbf{s}_i = [-2m_{7i},\ 2m_{5i},\ -2m_{6i}]^T. \tag{25}$$

5 Study Parameters Representation of the Cartesian Coordinates

Without loss of generality, assume the homogeneous condition to be $\Delta = 1$. Then, skipping mathematical derivation, it follows that:

$$x_3 = (-1)^{m_1} \frac{\sqrt{1 + \sin(\theta + \phi)}}{2} \quad x_0 = (-1)^{m_2} \frac{\sqrt{1 - \sin(\theta + \phi)}}{2}. \tag{26}$$

In the above $m_1 = \{0, 1\}$ and $m_2 = \{0, 1\}$ stand for the two distinct solutions. Handling the values for x_0 and x_3 leads to:

$$x_1 = (x_3 \sin\theta - x_0 \cos\theta), \quad x_2 = (x_0 \sin\theta + x_3 \cos\theta). \tag{27}$$

The transformation for the fixed parameters $\mathbf{r}_i$ and $\mathbf{s}_i$ can be obtained readily using Eq. (25). Handling the value of rotational parameters, i.e, x_i, $i = 0, \ldots, 3$, translational parameters, y_i, $i = 0, \ldots, 3$ can be found by back substitution into Eq. (24).

As it can be observed from the above, this mapping admits two distinct solutions for x_3 and x_0 for a given pose of the platform in the Cartesian space. These two distinct solutions can be classified as follows: $\theta\phi > 0$ then $m_1 = m_2$ or $\theta\phi \leq 0$ then $m_1 \neq m_2$. Thus from the above it follows that the mapping from the Study parameters to the Cartesian space is one to one and the converse, i.e., from Cartesian space to Study's parameters is two to one. This is called double covering of the Euclidean displacement group (SE(3)). More precisely: the dual quaternions are a double covering of SE(3).

6 Conclusion

This paper investigated the FKP of 5-DOF parallel mechanisms (3T2R) with a leg kinematic arrangement of type PRUR. By means of Study's kinematic mapping, the FKP is explored in a seven-dimensional projective space. Study's kinematic mapping leads to a simpler expression for the FKP of the so-called principal limb which is of degree 4, $\mathscr{F}_p(X)$, instead of 24 in the case of three-dimensional kinematic space, i.e., Cartesian coordinates. Using *Bertini*, a package for solving polynomial systems using Homotopy continuation, the FKP was solved and revealed that for a general design the degree of finite solutions is 1680, which provides some insight into the degree of the univariate polynomial for the FKP. Then, by a random numerical test, 208 real solutions were found. This example demonstrated that symmetrical 5-DOF parallel mechanisms admit more real solutions to their FKP than the 6-DOF parallel mechanism, the Gough–Stewart platform, with forty solutions. Future works include the solution of the FKP in a univariate form for a general design and the study of the singular configurations.

References

1. X. Kong and C. Gosselin. *Type Synthesis of Parallel Mechanisms*. Springer Tracts in Advanced Robotics, Vol. 3, Springer, Heidelberg (2007).
2. Z. Huang and Q. C. Li. General methodology for type synthesis of symmetrical lower-mobility parallel manipulators and several novel manipulators. *The International Journal of Robotics Research*, **21**(2), 131–145 (2002).
3. Y. Fang and L. W. Tsai. Structure synthesis of a class of 4-DoF and 5-DoF parallel manipulators with identical limb structures. *The International Journal of Robotics Research*, **21**(9), 799–810 (2002).
4. Z. Huang and Q. C. Li. Type synthesis of symmetrical lower-mobility parallel mechanisms using the constraint-synthesis method. *The International Journal of Robotics Research*, **22**(1), 59–79 (2003).
5. X. Kong and C. Gosselin. Type synthesis of 5-DOF parallel manipulators based on screw theory. *Journal of Robotic Systems*, **22**(10), 535–547 (2005).
6. J. Wang and C. Gosselin. Kinematic analysis and singularity representation of spatial five-degree-of-freedom parallel mechanisms. *Journal of Robotic Systems*, **14**(12), 851–869 (1997).

7. C. Gosselin, M. T. Masouleh, V. Duchaine, P. L. Richard, S. Foucault, and X. Kong. Parallel mechanisms of the multipteron family: Kinematic architectures and benchmarking. In *IEEE International Conference on Robotics and Automation*, Roma, Italy, 10–14 April, pp. 555–560 (2007).
8. M. L. Husty and H. P. Schröcker. Algebraic geometry and kinematics. In *Nonlinear Computational Geometry*, Emiris, I. Z., Sottile F. and Theobald T. (Eds.), pp. 85–106 (2007).
9. E. Study. Von den Bewegungen und Umlegungen. *Math. Ann.*, **39**, 441–566 (1891).
10. D. A. Cox, J. B. Little, and D. O'shea. *Using Algebraic Geometry*. Springer Verlag (2005).
11. D. J. Bates, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler. Bertini: Software for Numerical Algebraic Geometry. Available at http://www.nd.edu/∼sommese/bertini.
12. M. L. Husty. An algorithm for solving the direct kinematics of general Stewart–Gough platforms. *Mechanism and Machine Theory*, **31**(4), 365–379 (1996).

PART 4

The Development of a Reconfigurable Parallel Robot with Binary Actuators

D. Schütz, A. Raatz and J. Hesselbach

Institute of Machine Tools and Production Technology, Braunschweig, Germany; e-mail: {d.schuetz, a.raatz, j.hesselbach}@tu-bs.de

Abstract. Performing activities autonomously and independent of human interaction during the production and handling process of goods is often achieved by robots. Since many of these handling tasks require a manipulator that reaches only a few points, a binary robot, such as the structure presented in this contribution, is suitable. These parallel robots with binary actuation can be designed at low cost and controlled by a simple programmable logic controller (PLC). Conformance of a predetermined positioning task can be achieved through the development of a synthesis approach. Furthermore, a calibration strategy based on additional components is introduced. These modules allow an adjustment of geometrical parameters to ensure positioning tasks with high accuracy.

Key words: Binary robot, reconfigurable mechanism, reconfigurable robot, modular robot system, mechanical calibration.

1 Introduction

Binary actuators are characterized by a simple design and high repeatability, since their two states are based on mechanical end positions. Using these kind of actuators in a kinematic structure allows the design of a cost-effective robot. The workspace of these robots with n binary actuators is discretized to a total of 2^n points. A position control is not necessary and the restriction of the moving properties permits the use of a rudimentary control scheme. Thus a simple PLC is more than adequate as the vector of the drive coordinates $\mathbf{q}$ simply consist of a series of binary digits.

In the field of binary robots, research groups with different motivations for building binary robots exist. Plante et al. work with new lightweight actuators to reduce the total weight of the entire manipulator [1]. The objective of this concept is the establishment of binary robots for space activities, such as the manipulator from Sujan and Dubowsky, which must be designed to minimize the total weight in spite of redundant actuators [2]. Chirikjian has analyzed the kinematics of binary robots using hybrid structures with large numbers of actuators (ca. 30), where it was shown that the number of positions increase exponentially [3]. The objective is to realize

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_24,

a binary robot which is able to replace a continuous workspace manipulator. This approach leads to a layout of a serial kinematic structure, including several basic modules which consist of closed loops. Another example is the elastically averaged parallel manipulator with bistable Dielectric Elastomer Actuators (DEA) for MRI-guided prostate cancer therapy, which has been demonstrated by Plante and Dubowsky [4].

The more binary actuators are incorporated inside a kinematic structure, given that an application needs to reach plenty of points, the more the simplicity of binary robot control can be reduced. However, parallel robots with binary actuators can be used for handling tasks such as Pick-and-Place operations with few targets. The motivation behind this work is the development of a parallel robot with a minimum of binary actuators that is sufficient for a specific handling task. Furthermore, if the handling task changes in the future, the binary robot should be designed in such a way that a reconfiguration is possible and economical. Thus, a continuous system can almost be replaced by a modular binary robot system built with low-cost components such as pneumatic cylinders.

For easy and fast installation of these binary robots, a configuration software tool is required which uses a synthesis approach to find the geometrical parameter set which allows the structure to reach the positions of a specific task. Therefore the single parts of the binary robot must have a modular design for a prompt installation and increased reusability.

After installation of a robot, it is foreseen that there are differences between the prescribed and attained positions. The reason of these differences are unavoidable tolerances during the production of the robot parts and the assembly of the manipulator. Thus, there is a need for parameter calibration to enhance the absolute positioning accuracy of the parallel robot [5, 6]. As there is no way to implement an optimized parameter set inside the binary robot control, a method is presented which allows the modification of the structure parameters themselves. In addition to this, the specific calibration procedure is presented.

In this article a simple RPRPR Fivebar with three revolute joints (R) and two actuated prismatic joints (P) is used as an example to demonstrate the techniques for the synthesis and calibration procedure, which are needed for the installation of a specific binary robot configuration. Furthermore, this structure can be used for a four point task so that a prototype of the RPRPR Fivebar is planed to prove that four points can be accurantly reached with only two digital singnals.

2 The Fivebar Structure

An example of a binary robot with a minimum number of parts and with more than two possible positions is the RPRPR Fivebar. This manipulator consists of two linear actuators, for example pneumatic cylinder, and three passive revolute joints. The tool center point (TCP) can be located at the center of the revolute joint which joins the two actuators. Figure 1 illustrates the four reachable positions of this manipulator,

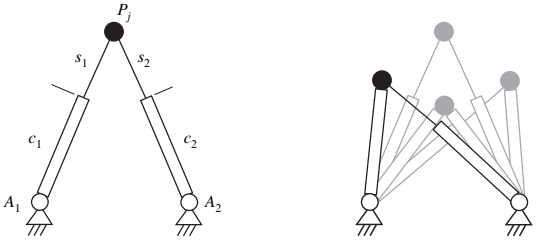


Fig. 1 The RPRPR Fivebar with binary actuators and its four possible positions.

Table 1 Geometrical parameters of the RPRPR Fivebar.

Description	Parameter
Coordinates of the fixed revolute joints	$A_i = \left(x_{A,i}; y_{A,i}\right)$
Cylinder length	c_i
Stroke of the piston	s_i
Coordinates of the TCP	$P_j = \left(x_{TCP,j}; y_{TCP,j}\right)$

which depend on the drive coordinates $\mathbf{q}$ representing the states of the both cylinders (where 0 stands for retracted piston and 1 stands for an extended piston).

As the geometrical parameters are not obvious at first glance, a software tool for synthesis can be used to assist in the configuration of this binary robot for a given four point application. Such a tool can be used to find the parameters for the RPRPR Fivebar (Table 1) to reach the points of a specific task.

3 Four Point Synthesis Approach

An analysis of a specific application is needed to set the geometrical parameters of the RPRPR Fivebar, as the regarded reconfigurable binary robot can only reach four points in total. The first step is to identify the minimal connection length of all four task relevant points P_j with $j = 1,\ldots,4$ which leads to a quadrilateral. Thereby, counter clockwise numbering simplifies further examination. In the constellation illustrated in Fig. 2, intersection points A_i with $i = 1,2$ can be developed from the perpendicular bisector of two opposite sides of the quadrilateral [7].

If these intersection points are not located inside the quadrilateral, they can be further used as the positions of the fixed revolute joints of the RPRPR Fivebar structure. To calculate the parameters c_i and s_i, the destination point P_j where the pistons of the cylinders are retracted, $\mathbf{q} = [0,0]^T$, must be identified. This point will be referred to as T for the rest of the synthesis. Therefore, the relative position of the joints A_i to the connection lines of the destination points is used to choose the point of P_j as T.

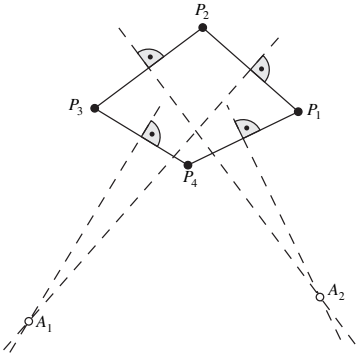


Fig. 2 Construction of the fixed joint positions A_i.

For example, to make a statement about the relative position of point A_1 to $\overrightarrow{P_3P_4}$, the coordinates of all these points are first combined in a matrix $\mathbf{M}$ as shown in Eq. (1) by the rule of Sarrus.

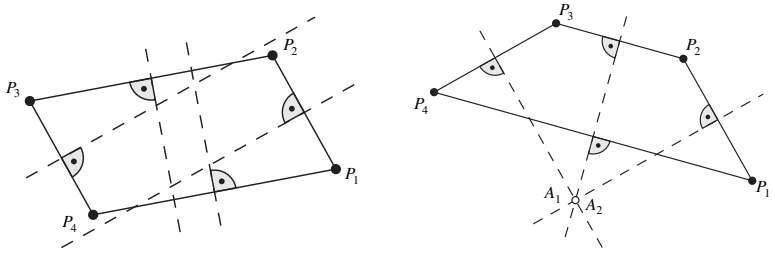
$$\mathbf{M} = \begin{bmatrix} x_{A,1} & x_{P,3} & x_{P,4} \\ y_{A,1} & y_{P,3} & y_{P,4} \\ 1 & 1 & 1 \end{bmatrix} \tag{1}$$

The determinant of the matrix $\mathbf{M}$ can then be computed and the result gives valuable information about the location of A_1. For a positive determinant, $\det(\mathbf{M}) > 0$, A_1 lies on the left side of $\overrightarrow{P_3P_4}$. If the determinant is negative, $\det(\mathbf{M}) < 0$, the point lies on the right side of the vector $\overrightarrow{P_3P_4}$. The point is part of $\overrightarrow{P_3P_4}$ if the determinant is zero. Hence, the identification of T is possible using knowledge about the location of the fixed revolute joints relative to the counter clockwise vector loop of the quadrilateral. There are two of the four vectors where one point A_i lies on the right side. The connecting point between these two vectors is point T. A function of the case differentiation to identify point T from P_j has been implemented in the synthesis algorithm.

The length from A_i to T represents the cylinder length c_i, as the piston of each cylinder is retracted. The distances of the fixed revolute joints to the other destination points P_j must either be equal to the respective cylinder length c_i or equal to the cylinder length plus the stroke s_i, the extended piston position of the cylinders.

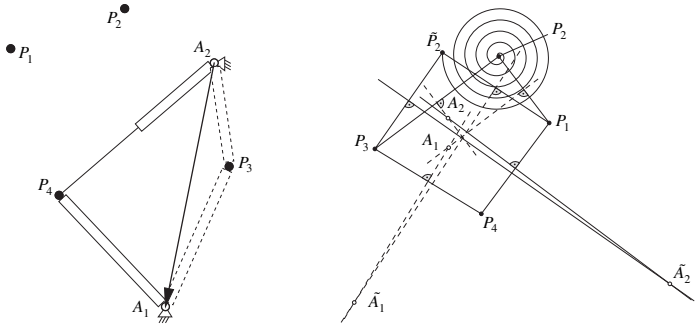
For some quadrilaterals it is impossible to find locations of the fixed revolute joints. For example, where the perpendicular bisector of two opposite sides of the quadrilateral do not intersect. This occurs if these lines lie parallel to each other. This case applies for quadrilaterals with a shape of a parallelogram or a rhombus (Fig. 3(a)). For a symmetrical trapezoid (Fig. 3(b)) the two points A_i lie inside each other and no suitable structure can be built.

In addition to restrictions such as installation space, joint angle, and singularities, it is necessary to ensure that the regarded four points are part of one workspace configuration. If one of the four points is part of the opposite workspace configuration



(a) Parallelogram/Rhombus (b) Symmetrical trapezoid

Fig. 3 Example quadrilaterals with no usable solution.



(a) Unworkable solution as the destination points lie in both workspace configurations.

(b) Achimedean spiral algorithm: P_2 has been moved to a position $\tilde{P}_2$ of a suitable solution.

Fig. 4 Different workspace configurations and a strategy to get a suitable solution.

as shown in Fig 4(a), it cannot be reached, as the binary robot is not able to drive through a singularity.

To find a solution, even though one of the described geometrical conditions is applied, one of the points P_j must be able to be moved freely. This strategy is possible if the specific task position can be changed. Then the single point will be moved, for example on an Archimedean spiral (Fig. 4(b)), until a suitable parameter set with $\tilde{A}_i$ can be found.

4 (Re-)Configuration of Binary Robots

Should the synthesis yield a suitable parameter set for the RPRPR Fivebar the cylinders can be ordered with the computed strokes s_i. To build the binary robot in the elaborated configuration, it is essential to support the installer with the first assembly, as the discretized workspace of the binary robot must fulfill the accuracy requirements of the destination points of the task. To use the same robot components for different structures, the maximal possible forces for each part of the modular robot system must be taken into account. Moreover, the objective is to integrate different pneumatic cylinders for different requirements such as high repeatability, high dynamics or low costs [8].

4.1 Robot Components for (Re-)Configuration

Since the production and assembly are subjected to tolerances, it must be assured that the four destination points will be reached with a sufficient accuracy. Therefore, a parameter calibration is typically used to correct these deviations. The identification of the parameter deviations can be done with well known optimization methods, such as the Levenberg–Marquardt algorithm [9]. The simplified control of the binary robots makes it impossible to change the parameters within the controller. The only way to change the values of the parameters is to modify the structure itself. Additional components have been developed for a prototype of the RPRPR Fivebar, which allow the adjustment of some of the coordinates of the fixed revolute joints A_i and the length of the cylinders c_i. The stroke of the pneumatic cylinders cannot be modified after production, as they are designed with a damping system at their end positions. The correction of the task points P_j have to be compensated by modifying the other parameters.

4.2 Mechanical Calibration of Binary Robots

After the first assembly of the RPRPR Fivebar the actual points which the binary robot can reach need to be measured. The difference between the actual points $P_{j,a}$ and the nominal points $P_{j,n}(\mathbf{k})$ of the task depending on the parameter vector $\mathbf{k}$, including the coordinates of A_i and the lengths of c_i, can be written in a residuum r_j as shown in Eq. (2).

$$r_j(\mathbf{k}) = P_{j,a} - P_{j,n}(\mathbf{k}) \tag{2}$$

Combining the deviations of all four points in one vector $\mathbf{r}$ and multiplying it with its transposed $\mathbf{r}^T$ results in a scalar F (Eq. 3).

$$F = \mathbf{r}^T \cdot \mathbf{r} \tag{3}$$

The cost function F can be minimized within an optimization procedure by modifying the components of the parameter vector $\mathbf{k}$, until F is within a predefined range. The optimization function outputs a parameter vector $\mathbf{k}_{opt}$ with modified parameter values. The next step of a typical calibration procedure will now be the implementation of these parameters inside of the control [10]. This will improve the robot's ability to reflect the actual system.

In the case of a binary robot, only the geometrical parameters of the structure can be adjusted by the presented additional components. Hence, it is a matter of a mechanical calibration. In order to reach the nominal destination points $P_{j,n}$, the differences of Eq. (4) between the nominal and the optimized parameter values represent the relative parameter variation, which is necessary to improve the accuracy of the binary robot.

$$\Delta\mathbf{k} = \mathbf{k} - \mathbf{k}_{opt} \tag{4}$$

5 Operational Status

The presented methods have already been realized within a software tool that allows the geometrical parameters of the RPRPR Fivebar to be synthesized. An algorithm has been implemented that moves one point of the quadrilateral on an Archimedean spiral until a suitable solution can also be found. After the structure has been configured and installed for a specific task, the software supports the installer with the required calibration of the binary robot. Here the reachable points $P_{j,a}$ are used as an input to calculate the relative parameter adjustments $\Delta\mathbf{k}$. To be able to use this software tool, a prototype system should be developed on which the elaborated methods can be tested and validated. Therefore, specific robot parts are designed such as revolute joints (Fig. 5) with the opportunity to build other kinematic structures with the same parts by connecting different binary actuators or links. In addition to the presented procedure for a mechanical calibration of parallel robots with binary actuators, components for parameter modification inside the structure are necessary. Thus, these parts allow the adjustment of the coordinates of the fixed revolute joints A_i and the correction of the cylinder alignment relative to these fixed revolute joints which is represented by the parameter c_i. With these parts it will be possible to configure and reconfigure other parallel binary robots besides the RPRPR Fivebar.

6 Conclusions

Planar parallel robots with binary robots are suitable for simple handling tasks in which only a few destination points need to be reached. As they feature a discrete workspace, methods for their synthesis and their calibration are essential. In particular if the number of needed destination points is equal to the number of possible robots positions. To accomplish these tasks, this article introduces methods

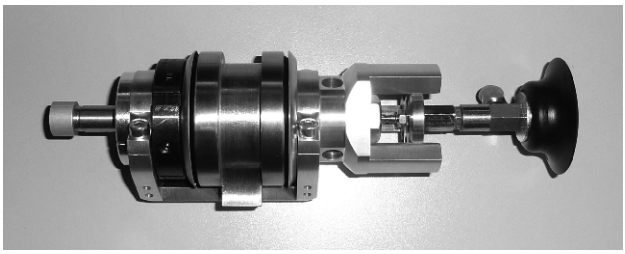


Fig. 5 Prototype of a revolute joint with pneumatic cylinder for a movement in z-direction and vakuum gripper.

to synthesize and to calibrate a binary robot with the well known RPRPR Fivebar structure. An approach for a four point synthesis is presented, which allows to set the geometrical parameters of this structure such that a predefined four point constellation can be realized. A modification of parameter calibration is introduced as mechanical calibration for binary robots so that the actual point positions satisfy the required accuracy. Due to the binary nature of the controller, additional components are developed which allow for a modification of selective parameters.

References

1. Plante, J. S., Santer, M., Dubowsky, S., Pellegrino S.: Compliant bistable dielectric elastomer actuators for binary mechatronic systems. In: *Proceedings of the ASME Conference on Mechanisms and Robotics* (2005).
2. Sujan, V., Dubowsky, S.: Design of a lightweight hyper-redundant deployable binary manipulator. *ASME Journal of Mechanical Design*, **126**(1), 29–39 (2004).
3. Chirukjian, G.: Kinematic synthesis of mechanisms and robotic manipulators with binary actuators. *Journal of Mechanical Design*, **117**, 573–580 (1995).
4. Plante, J. S., Dubowsky, S.: The calibration of a parallel manipulator with binary actuation. In: *Advances in Robot Kinematics: Analysis and Design*, J. Lenarcic and P. Wenger (Eds.), Springer, Dordrecht, pp. 391–398 (2008).
5. Bernhardt, R., Albright, S. L.: *Robot Calibration*. Chapman & Hall, London, 1 edition (1993).
6. Zhuang, H., Roth, Z.: Method for kinematic calibration of stewart platforms. *Journal of Robotic Systems*, **10**(3), 391–405 (1993).
7. Dizioglu, B.: *Getriebelehre: Bd. 2: Maßbestimmung*. Vieweg (1967).
8. Schütz, D., Soetebier, S., Last, P., Raatz, A.: Adapted task configuration of a deltapod. In: *ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots*, IEEE, Piscataway, NJ, pp. 545–549 (2009).
9. Scales, L. E.: *Introduction to Non-Linear Optimization*. Macmillan Computer Science Series, Macmillan, London (1985).
10. Last, P., Budde, C., Schütz, D., Hesselbach, J., Raatz, A.: Parallel robot calibration by working mode change. In: *Advances in Robot Kinematics: Analysis and Design*, J. Lenarcic and P. Wenger (Eds.), Springer, Dordrecht, pp. 371–380 (2008).

On the Design of 5R Serial Manipulators with Isotropic Positioning

L. Baron

École Polytechnique, Montréal, Canada; e-mail: luc.baron@polymtl.ca

Abstract. This paper study the serial composition of two isotropic serial chains aiming at isotropic positioning tasks, and resulting in two new 5R serial manipulators. The operating point of manipulator A can undergo the full surface of a sphere of near isotropic positioning, while maintaining a low constant overall conditioning. The Gram–Schmidt procedure can orthogonalized the rows of the positioning jacobian matrix arising from a selected point over that sphere. The resulting manipulator *B* can reach isotropic positioning, while undergoing only the trivial circle.

Key words: Isotropy, kinematic design, serial manipulator.

1 Introduction

This paper introduces two new 5-revolutes (5R) serial manipulators having isotropic and near isotropic positioning capability. A manipulator is said *isotropic* if its Jacobian matrix can reach isotropic values – i.e. the minimum condition number of unity – at least at a single posture within its workspace [1]. The design of manipulators able to reach isotropy is of particular interest because they exhibit their best kinematic performances in such postures, namely *isotropic postures* [2]. Consequently, many research works have been conducted toward finding architectures capable to reach isotropy and their corresponding isotropic postures. When the concept of isotropy is applied to serial manipulators, two basic properties have been pointed out in the literature. First, the isotropy is not affected by a rotation of the first joint, and hence, a point P of the end-effector (EE) can undergo a *trivial path* – a circle – while being at the same isotropic posture [1]. Second, the isotropy of an orienting serial chain is not affected by a rotation of the last joint [3].

For positioning tasks, a 2R, a 3R and a 4R planar serial manipulators have, respectively, 2, 2 and 4 isotropic postures along a trivial path [4, 2]. A 4R spatial serial manipulator has also isotropic postures along a trivial path [4]. For orienting tasks, a 3R, 4R and 5R spherical serial manipulators have isotropic postures along a trivial path [5, 6].

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_25,

Recently, a 6R spherical serial manipulator has been proposed [7], for which the EE can undergo a non-trivial path – any EE orientation – while being at the same isotropic posture. In fact, this manipulator is obtained from the serial composition of two isotropic 3R orienting manipulators. The non-trival path comes from the trivial path of the first joint together with the trivial path of the intermediate joints of the two sub-chains, i.e., joints 3 and 4. Similarly, a 4R spatial serial manipulator has been proposed [10], for which a point P can undergo a *non-trivial path* – the surface of a sphere – while being at the same isotropic posture. Here again, this manipulator is obtained from the serial composition of an isotropic 3R orienting manipulators followed by an isotropic 2R planar manipulator with a shared intermediate revolute joint, thus resulting in a 4R spatial serial manipulator with isotropic positioning.

For both positioning and orienting tasks, it is required to render dimensionless the jacobian matrix by dividing both sides of the velocity equation of point P by a *characteristic length* of the manipulator. This length is defined as that minimizing the condition number of the dimensionless jacobian [4]. Resorting to this characteristic length, the nR planar serial isotropic manipulators have isotropic postures along a trival path [4]. Similarly, the 6R spatial serial manipulator, called *diestro*, has also isotropic postures along a trival path [8].

In this paper, we study the serial composition of two isotropic serial manipulators, namely a 3R orienting manipulator followed by a planar 3R positioning manipulator. Since the last joint of the orienting manipulator and the first joint of the positioning manipulator are both irrelevant to the isotropy of each individual manipulators, they could shared their intermediate joints, thus resulting in a 5R manipulator that we will investigate its possible isotropy. The two proposed manipulators have either near isotropic positioning postures along a non-trivial path – the surface of a sphere – or isotropic positioning postures along a trival path. These manipulators are aimed at a class of tasks requiring only 5 degrees-of-freedom, i.e., the 3D positioning of the EE together with two orientation angles – the pitch and yaw only, while the roll around the last link is not feasible.

2 Kinematic Model

The Denavit–Hartenberg (DH) parameters of the two new 5R serial manipulators are listed in Table 1 and depicted in Figs. 1 and 2. The angular velocity of the end-effector (EE), namely $\boldsymbol{\omega}$, is given as

$$\boldsymbol{\omega} = \mathbf{A}\dot{\boldsymbol{\theta}}, \quad \dot{\boldsymbol{\theta}} \equiv [\dot{\theta}_1\ \dot{\theta}_2\ \dot{\theta}_3\ \dot{\theta}_4\ \dot{\theta}_5]^T, \tag{1}$$

in terms of the 3×5 *orienting Jacobian matrix* $\mathbf{A}$ defined as

$$\mathbf{A} \equiv [\mathbf{e}_1\ \mathbf{e}_2\ \mathbf{e}_3\ \mathbf{e}_4\ \mathbf{e}_5], \tag{2}$$

with $\mathbf{e}_i$ being the unit vector along the joint axis i. Moreover, the velocity vector of point P of the EE, namely $\dot{\mathbf{p}}$, is given as

Table 1 DH parameters of the dimensionless positioning 5R serial manipulator with arbitrary l, arbitrary β_1, and $\beta_5 = \frac{1}{\sqrt{3}}$, where $\beta_2 = 0$ for manipulator A and $\beta_2 = \frac{\sqrt{2}}{2} - \beta_5$ for manipulator B.

i	θ_i	a_i	b_i	α_i
1	θ_1	0	$\beta_1 l$	$\frac{\pi}{2}$
2	θ_2	$\beta_2 l$	0	$-\frac{\pi}{2}$
3	θ_3	l	0	0
4	θ_4	l	0	0
5	θ_5	$\beta_5 l$	0	0

$$\dot{\mathbf{p}} = \mathbf{B}\dot{\boldsymbol{\theta}}, \quad \dot{\mathbf{p}} \equiv [\dot{x}\ \dot{y}\ \dot{z}]^T, \tag{3}$$

in terms of the 3×5 *positioning Jacobian matrix* $\mathbf{B}$ defined as

$$\mathbf{B} \equiv [\mathbf{e}_1 \times \mathbf{r}_1\ \mathbf{e}_2 \times \mathbf{r}_2\ \mathbf{e}_3 \times \mathbf{r}_3\ \mathbf{e}_1\mathbf{e}_4 \times \mathbf{r}_4\ \mathbf{e}_5 \times \mathbf{r}_5] \tag{4}$$

with $\mathbf{r}_i$ being the position of P relative to the origin of frame F_i expressed in frame F_1 [3, see Chap. 4]. Vectors $\mathbf{r}_i$ are not provided here, but can be readily obtained from the DH parameters. Manipulators A and B are dimensionless, since the lenght l can be arbitrarily chosen. Moreover, the ratio β_1 can also be arbitrarily chosen. The other link lengths come from the geometry of the equilateral triangle corresponding to the isotropic posture of the planar 3R manipulator, links 3 and 4 being along the sides of the triangle of arbirary lenght l, while link 5 goes from the coner to the center of the triangle, a length of $a_5 = \beta_5 l = \frac{1}{\sqrt{3}} l$.

3 Isotropic Design

A $m \times n$ rectangular Jacobian matrix $\mathbf{M}$ is isotropic if [4]:

$$\mathbf{M}\mathbf{M}^T = \sigma^2 \mathbf{1}_{m\times m} \quad m < n \tag{5}$$

$$\mathbf{M}^T\mathbf{M} = \sigma^2 \mathbf{1}_{n\times n} \quad m \geq n \tag{6}$$

for which the condition number is unity.

3.1 Manipulator A with Near Isotropic Positioning

For a positioning task, manipulator A has a near isotropic positioning Jacobian matrix $\mathbf{B}$ such that

$$\mathbf{B}\mathbf{B}^T \approx \sigma^2 \mathbf{1}_{3\times 3} \tag{7}$$

if

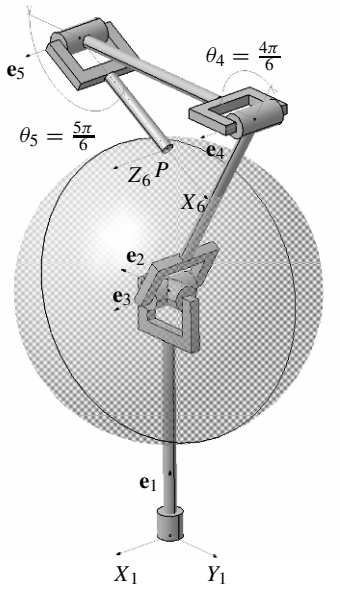


Fig. 1 Manipulator A with near isotropic positioning over a sphere of radius $\beta_5 l$ with $\beta_2 = 0$ and $\beta_5 = \frac{1}{\sqrt{3}}$ for $\forall\theta_1$, θ_3 subject to $\theta_2 = \frac{\pi}{2}$, $\theta_4 = \frac{4\pi}{6}$ and $\theta_5 = \frac{5\pi}{6}$.

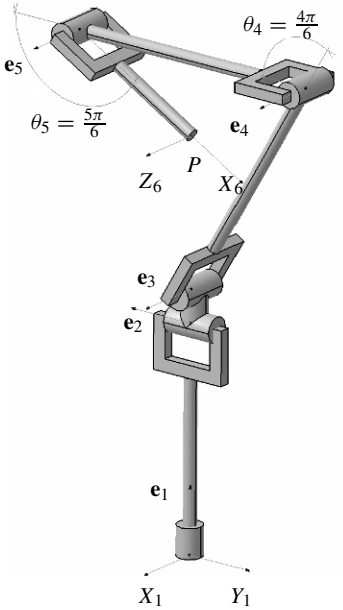


Fig. 2 Manipulator B with isotropic positioning with $\beta_2 = \frac{\sqrt{2}}{2} - \beta_5$ and $\beta_5 = \frac{1}{\sqrt{3}}$ for $\forall\theta_1$ subject to $\theta_2 = \frac{\pi}{2}$, $\theta_3 = -\frac{\pi}{6}$, $\theta_4 = \frac{4\pi}{6}$ and $\theta_5 = \frac{5\pi}{6}$.

$$\theta_2 = \frac{\pi}{2}, \quad \theta_4 = \frac{4\pi}{6}, \quad \theta_5 = \frac{5\pi}{6}, \quad \forall\theta_1, \theta_3. \tag{8}$$

Under the near isotropic conditions of Eq. (8), the condition number, namely $cond(\cdot)$, of matrix $\mathbf{B}$ gives the minimum

$$cond(\mathbf{B}) = \sqrt{3/2}, \quad \forall\theta_1,\ \theta_3. \tag{9}$$

As shown in Fig. 1, manipulator A is able to reach the minimum condition number of $\sqrt{3/2}$ along a non-trivial path, i.e., point P can undergo the surface of a sphere of radius $l/\sqrt{3}$, while being at the same near isotropic positioning posture. The manipulator is more likely to maintain good condition numbers, when P undergo an arbitrary trajectory in the vicinity of this surface. For an orienting task, manipulator A has an orienting Jacobian matrix $\mathbf{A}$ such that

$$\mathbf{A}\mathbf{A}^T \approx \sigma^2 \mathbf{1}_{3\times 3}, \tag{10}$$

which never reach isotropic values for the isotropic conditions of Eq. (8). In general, matrix $\mathbf{A}$ takes the form

$$\mathbf{A} = \begin{bmatrix} 0 & s_1 & c_1 s_2 & c_1 s_2 & c_1 s_2 \\ 0 & -c_1 & s_1 s_2 & s_1 s_2 & s_1 s_2 \\ 1 & 0 & -c_2 & -c_2 & -c_2 \end{bmatrix}. \tag{11}$$

Apparently, matrix $\mathbf{A}$ is independent of θ_3, θ_4 and θ_5. Upon substituting Eq. (8) into (11), the condition number of matrix $\mathbf{A}$ gives the minimum

$$cond(\mathbf{A}) = \sqrt{3}, \quad \forall \theta_1,\ \theta_3,\ \theta_4,\ \theta_5. \tag{12}$$

Consequently, matrix $\mathbf{A}$ is very close to isotropy. It is independent of θ_1, θ_3, θ_4 and θ_5, and hence, depends only on θ_2.

3.2 Manipulator B with Isotropic Positioning

In order to obtain a manipulation capable to reach positioning isotropy at least in some isolated postures, we need to render $\mathbf{B}$ isotropic in those postures. First, we select a convenient position of point P over the sphere of near isotropic postures, e.g.:

$$\theta_1 = 0, \quad \theta_3 = -\frac{\pi}{6}, \tag{13}$$

which gives the following matrix $\mathbf{B}$:

$$\mathbf{B} = \begin{bmatrix} 0 & -\frac{1}{\sqrt{3}} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{3}} & -\frac{1}{2\sqrt{3}} & -\frac{1}{2\sqrt{3}} \\ 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} \end{bmatrix} l \tag{14}$$

Second, we use the *Gram–Schmidt orthogonalization procedure* [9] in order to render $\mathbf{B}$ isotropic. In 5-dimensional space, the three rows of $\mathbf{B}$, namely $\mathbf{b}_1^T$, $\mathbf{b}_2^T$ and $\mathbf{b}_3^T$ respectively, need to be orthogonal and of the same length. Apparently from Eq. (14), the norm of the first row is different from the other two, i.e., $\mathbf{b}_1^T\mathbf{b}_1 \neq \mathbf{b}_2^T\mathbf{b}_2 = \mathbf{b}_3^T\mathbf{b}_3$, while the three rows are already orthogonal, i.e., $\mathbf{b}_1 \perp \mathbf{b}_2 \perp \mathbf{b}_3 \perp \mathbf{b}_1$. Once $\mathbf{b}_1$ normalized, the resulting matrix, namely $\mathbf{B}'$, is given as:

$$\mathbf{B}' = \begin{bmatrix} 0 & -\frac{1}{\sqrt{2}} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{3}} & -\frac{1}{2\sqrt{3}} & -\frac{1}{2\sqrt{3}} \\ 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} \end{bmatrix} l, \tag{15}$$

where $\mathbf{B}'$ is now isotropic

$$\mathbf{B}\mathbf{B}^T = \sigma^2 \mathbf{1}_{3\times 3} \tag{16}$$

with $\sigma^2 = \frac{1}{2}l^2$, and hence,

$$cond(\mathbf{B}) = 1, \quad \forall \theta_1. \tag{17}$$

Matrix $\mathbf{A}$ remains identical, with $cond(\mathbf{A}) = \sqrt{3}$. Finally, vectors $\mathbf{r}'_i$ are computed from the columns of $\mathbf{B}'$ and $\mathbf{A}$, namely $\mathbf{c}'_i$ and $\mathbf{e}_i$, as:

$$\mathbf{r}'_i = \mathbf{c}'_i \times \mathbf{e}_i, \tag{18}$$

from which the DH parameters of manipulator B are readily found (as listed in Table 1) and shown in Fig. 2.

3.3 Arbitrary 5R Manipulator with Isotropic Positioning

An arbitrary position over the sphere of near isotropic postures can be selected, e.g.,

$$\theta_1 = \frac{\pi}{4}, \quad \theta_3 = -\frac{\pi}{4}, \tag{19}$$

for which we obtain:

$$\mathbf{B} = \begin{bmatrix} 0.1057 & -0.3943 & -0.3943 & 0.1057 & 0.2887 \\ 0.1057 & -0.3943 & 0.3943 & -0.1057 & -0.2887 \\ 0 & 0 & 0.1494 & -0.5577 & 0.4082 \end{bmatrix} l, \tag{20}$$

Apparently, $\mathbf{b}_1^T\mathbf{b}_1 \neq \mathbf{b}_2^T\mathbf{b}_2 = \mathbf{b}_3^T\mathbf{b}_3$ and $\mathbf{b}_2 \perp \mathbf{b}_3 \perp \mathbf{b}_1$, but $\mathbf{b}_1$ is not orthogonal to $\mathbf{b}_2$, since $\mathbf{b}_1^T\mathbf{b}_2 \neq 0$. We can apply the Gram–Schmidt procedure on $\mathbf{b}_1$ as:

1. $\mathbf{b}'_1 = \mathbf{b}_1 - \frac{1}{\mathbf{b}_2^T\mathbf{b}_2}\mathbf{b}_2\mathbf{b}_2^T\mathbf{b}_1$ (orthogonalization of $\mathbf{b}_1$);
2. $\mathbf{b}'_1 = \frac{\|\mathbf{b}_3\|}{\|\mathbf{b}'_1\|}\mathbf{b}'_1$, $\mathbf{b}'_2 = \frac{\|\mathbf{b}_3\|}{\|\mathbf{b}_2\|}\mathbf{b}_2$, $\mathbf{b}'_3 = \mathbf{b}_3$ (normalization of $\mathbf{b}'_1$ and $\mathbf{b}_2$ as $\mathbf{b}_3$);

or on $\mathbf{b}_2$ as:

1. $\mathbf{b}'_2 = \mathbf{b}_2 - \frac{1}{\mathbf{b}_1^T\mathbf{b}_1}\mathbf{b}_1\mathbf{b}_1^T\mathbf{b}_2$ (orthogonalization of $\mathbf{b}_2$);
2. $\mathbf{b}'_2 = \frac{\|\mathbf{b}_3\|}{\|\mathbf{b}'_2\|}\mathbf{b}'_2$, $\mathbf{b}'_1 = \frac{\|\mathbf{b}_3\|}{\|\mathbf{b}_1\|}\mathbf{b}_1$, $\mathbf{b}'_3 = \mathbf{b}_3$ (normalization of $\mathbf{b}'_2$ and $\mathbf{b}_1$ as $\mathbf{b}_3$);

Once the Gram–Schmidt procedure applied on $\mathbf{b}_1$, we obtain

$$\mathbf{B}' = \begin{bmatrix} 0.1418 & -0.5291 & -0.3527 & 0.0945 & 0.2582 \\ 0.1157 & -0.4320 & 0.4320 & -0.1157 & -0.3162 \\ 0 & 0 & 0.1494 & -0.5577 & 0.4082 \end{bmatrix} l, \tag{21}$$

which is readily verified to be isotropic. Using the selected position of Eq. (19), matrix $\mathbf{A}$ is given as

$$\mathbf{A} = \begin{bmatrix} 0 & 0.7071 & -0.7071 & -0.7071 & -0.7071 \\ 0 & -0.7071 & -0.7071 & -0.7071 & -0.7071 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{22}$$

allowing the computation of matrix $\mathbf{R}'$ (columns made of vectors $\mathbf{r}'_i$) using Eq. (18) as

$$\mathbf{R}' = \begin{bmatrix} 0.1157 & 0 & 0.1057 & -0.3943 & 0.2887 \\ -0.1418 & 0 & -0.1057 & 0.3943 & -0.2887 \\ 0.0 & 0.6796 & 0.5549 & -0.1487 & -0.4062 \end{bmatrix} l, \tag{23}$$

from which another set of DH parameters is found. Again here, the resulting manipulator is able to reach the positioning isotropy at the selected position over the sphere, and undergo a trivial path with θ_1.

4 Positioning and Orienting Tasks

For this kind of tasks, a *dimensionless* Jacobian matrix, namely $\mathbf{J}$, is required, which use a characteristic length L of the manipulator in order to normalize matrix $\mathbf{B}'$, i.e.,

$$\mathbf{J} = \begin{bmatrix} \mathbf{A} \\ \frac{1}{L}\mathbf{B}' \end{bmatrix}, \quad L = \frac{1}{2}, \quad \forall \theta_1, \theta_3 \tag{24}$$

Upon substituting Eq. (24) into (6), and using any arbitrary position over the sphere, e.g., Eq. (13) or Eq. (19), we have

$$cond(\mathbf{J}) = \sqrt{3}, \quad \forall \theta_1, \theta_3, \tag{25}$$

where the characteristic length $L = \frac{1}{2}$ is the one allowing the minimum condition number of Eq. (25).

5 Conclusions

New 5R serial manipulators have been proposed for isotropic positioning tasks. Although manipulator A is not isotropic, its operating point can undergo the surface of a sphere, while having $cond(\mathbf{B}) = \sqrt{3/2}$ and $cond(\mathbf{A}) = \sqrt{3}$, yet a rather interesting behavior. In order to effectively reach isotropic positioning, we proposed to orthgonalize the three rows of $\mathbf{B}$ with the Gram–Schmidt procedure from an arbitrary position over that sphere, thus resulting in manipulators, like manipulator B. The latter reaches isotropic positioning at the selection position, i.e., $cond(\mathbf{B}) = 1$ and $cond(\mathbf{A}) = \sqrt{3}$, making an overall $cond(\mathbf{J}) = \sqrt{3}$, again a rather interesting behavior.

Acknowledgements The author acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada under grant RGPIN-203618.

References

1. Salisbury, J.K. and Craig, J.J.: Articulated hands: Force control and kinematic issues. *Int. J. of Robotic Research*, **1**, 4–17 (1982).
2. Chablat, D. and Angeles, J.: On isotropic sets of points in the plane: Application to the design of robot architectures. In: *Advances in Robot Kinematic*, pp. 73–82 (2000).
3. Angeles, J.: *Fundamentals of Robotic Mechanical Systems*. Springer-Verlag, Berlin (2003).
4. Angeles, J.: The design of isotropic manipulator architectures in the presence of redundancies. *Int. J. of Robotic Research*, **11**, 96–201 (1992).
5. Chablat, D. and Angeles, J.: The computation of all 4R serial spherical wrists with an isotropic architecture. *Int. J. of Mechanical Design*, **125**, 275–280 (2003).
6. Akrout, K., Baron, L. and Wang, X.: Existence d'une infinité non dénombrable de positions isotropes pour les manipulateurs 5R sphériques. In: *Proceedings 10th IFToMM World Congress*, Besancon, France (2008).
7. Akrout, K. and Baron, L.: Manipulateur 6R sphérique isotrope pour toute orientation de son effecteur. *Transactions of the Canadian Society for Mechanical Engineering* (2010).
8. Ranjbaran, F., Angels, J. and Gonzalez-Palacios, M.A.: The mechanical design of a seven-axes manipulator with kinematic isotropy. *Int. J. of Intelligent and Robot Systems*, **14**, 21–41 (1995).
9. Golub, G.H. and Van Loan, C.F.: *Matrix Computations*. John Hopkins University Press, Baltimore (1989).
10. Akrout, K. and Baron, L.: Positio: A 4R serial manipulator having a surface of positioning isotropy. *IFToMM-RoManSy Symposium*, Udine, Italy, submitted (2010).

A Virtual Mechanism Enhanced Approach for Object Tracking with Humanoid Robot Head

Damir Omrčen and Aleš Ude

[1] *J. Stefan Institute, Slovenia; e-mail: damir.omrcen@ijs.si*
[2] *J. Stefan Institute, Slovenia and ATR Computational Neuroscience Laboratories, Japan; e-mail: ales.ude@ijs.si*

Abstract. In this work we present a very systematic approach to control a humanoid robot head for 3-D object tracking. The proposed approach is based on the concept of virtual mechanism, where the real head is enhanced with a virtual link that connects the eyes with a point in 3-D space. The use of virtual mechanism results in a more systematic description of a task and allows us to use "off the shelf" control algorithms. Additionally, it can be easily implemented on different types of active humanoid heads. We implemented the proposed approach on a humanoid head with two rigidly connected cameras in each eye (wide-angle and telescopic). The experimental results show that the proposed control algorithm can be used to maintain the view of an observed object in the foveal (telescopic) image using information from the peripheral view. Unlike other methods proposed in the literature, our approach indicates how to exploit the redundancy of the robot head.

Key words: Active stereo vision, foveated vision, humanoid head, virtual mechanism, redundancy resolution.

1 Introduction

Robots with humanoid heads are not only more social, but they also posses an active vision system. Already in 1991, Ballard [1] described the implications of having a visual system that could actively position the camera coordinates in response to physical stimuli. Active vision is much more challenging, however, it is also much more adaptable and flexible (e.g. using active vision wider field of view can be achieved).

Human eye movements and gaze direction have a high communicative value [2, 3]. For example, gaze direction is a good indicator of the locus of visual attention. For example, when an anthropomorphic robot moves its eyes and neck to fixate on an object, an observer can conclude that the robot has become interested in that object.

Many visual tasks require both high resolution and a wide field of view. High acuity is needed for recognition tasks, while a wide field of view is needed for object detection, for tracking multiple objects, etc. A common trade-off found in biological

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_26,

systems is to sample part of the visual field at a high enough resolution to support the first set of tasks and to sample the rest of the field at an adequate level to support the second set. This is seen in animals with foveated vision, where the density of photoreceptors is highest at the center of the retina and falls off rapidly towards the periphery. Designers of a number of humanoid robots (Cog [3, 4], Kismet [2], Armar III [5], DB [6], etc.) attempted to mimic the foveated structure of the human eye by using two rigidly connected cameras in each eye. Here, the optical axes of the narrow- and wide-angle cameras are not aligned which makes the control problem more complex. In this paper we discuss the control of such humanoid visual systems.

Metta et al. use learning methods to control the gaze direction of the eyes [7, 8]. The gaze direction is also affected by the head movement, therefore the eyes are counter-rotated based on a reactive control using inertial sensor in the head [8]. Bernardino et al. [9] proposed a simplified decomposed controller, which restricts the diversity of all possible movements. In [10] a tracking system with log-polar cameras was proposed. The system is based on the estimation of redundant 2-D motion parameters.

Ude et al. [11] proposed a similar decomposed controller, realized as a network of PD controllers. They are based on simplified mappings between visual coordinates and joint angles, rather than on a full kinematic model. Authors additionally implemented a transformation which assures that the object, which needs to be tracked, is kept in the center of the narrow-angle cameras, even though it is tracked by the wide-angle cameras. However, this transformation is only valid if the object is sufficiently far from the cameras.

A robotic head is an example of a mechanism that has some DOFs in one common kinematic chain (e.g. the neck) while some of DOFs are specific to each branch of a system (e.g. each eye pan). This kind of systems can be treated as a branching mechanism [12]. The most general way to control a branching mechanism is to treat both branches equally so that the movement of the common DOFs is defined by the tasks of both sub-branches equally, while the motion of a particular branch only depends on a particular task. In the other case, one of the branches can be dominant and can define the motion entirely, while the motion of the other branch adapt to this motion.

A humanoid head is redundant with respect to the task of fixating on an object. A redundant manipulator has more DOFs than what is required to solve the task and has, therefore, the ability to move in the joint space without affecting the motion in the task space. It is beneficial to exploit the redundancy of the robot head. For example, when a robot neck is close to the joint limit the robot can use other joints (e.g. the eyes) to accomplish the task (see joint limit avoidance in [13] or obstacle avoidance in [14]). Similarly, joint torques can be optimized [15] and lower energy consumption can be achieved (or low fatigue in case of a human). In order to decrease the error in the secondary task execution two approaches are proposed in [16]. Nevertheless, we can apply different voluntary head motions such as nodding and still assure a stable gaze on the object.

In the paper we propose a very systematic approach for controlling a robot head. The proposed controller assures that an object tracked by the wide-angle cameras

is in the center of the narrow-angle ones regardless of the object/camera distance. Additionally, we demonstrate how to exploit robot's redundancy in such a way that the robot can perform additional tasks which are typical for humans. The other approaches proposed in the literature [7–9] only limit the variety of human movements by applying additional constraints.

2 Methods

The initial goal of our work is to obtain high resolution images of an object in the robot's environment. Because the peripheral cameras have a very wide field of view, we cannot extract detailed object features from these images. Just like humans must fixate on an object to discriminate fine detail, our foveal cameras must be pointed in the direction of a given object in order to provide sufficient resolution. To accurately track an object, we need to solve both perceptual and control problems:

- The perceptual problem deals with the estimation of location of a target object. We estimate 3-D position of the object based on kinematics of the robot head and stereo *wide*-angle camera image information.
- The control problem addresses the control of the head. It assures that an object is kept in the center of the *narrow*- or wide-angle cameras using the estimated target position. To solve the control problem, we have introduced a virtual mechanism. Virtual mechanism is an auxiliary mechanism that points from both robot eyes to a 3-D point in space. It allows us to properly define a task (the gaze direction) and solve the control problem more systematically.

To evaluate our approach we carried out several experiments with a humanoid head (Fig. 1) similar to the head used on Armar III humanoid robot [5]. The head has seven mechanic degrees of freedom (DOFs) and two eyes. The eyes have a common tilt and can pan independently. The visual system is mounted on a four DOFs neck, which is realized as a Pitch-Roll-Yaw-Pitch mechanism. Each eye is equipped with two cameras (wide- and narrow-angle) to allow visuomotor behaviors such as tracking and saccadic motions. The head features human-like characteristics in motion and response.

2.1 The Perceptual Problem – Acquiring 3-D Position of Object

The perception problem deals with the problem of acquiring 3-D position of an object in space. The problem can be solved using two images from the wide-angle stereo cameras and the head kinematics [17].

To define the 3-D object position a good intrinsic and extrinsic camera models are crucial. Extrinsic camera parameters describe the position and orientation of the cameras in space (kinematic model of the head). Here, we consider the poses of

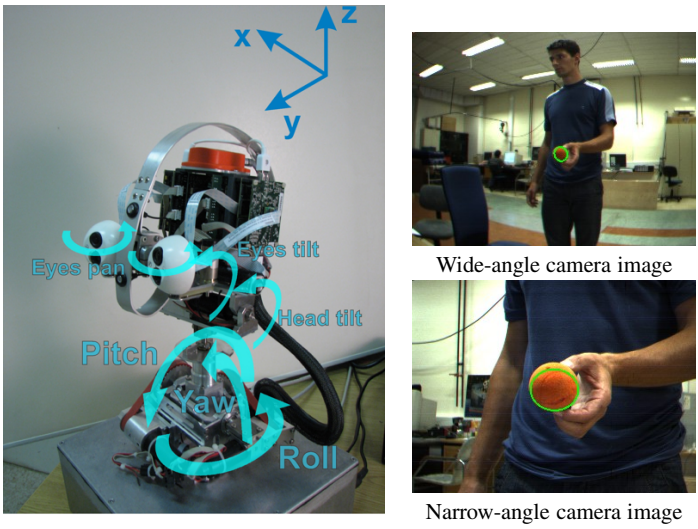


Fig. 1 Humanoid robot head and snapshots of wide- and narrow-angle cameras.

the wide-angle cameras since those are used for object detection. The accuracy of the head kinematics can be quite low on light weight humanoid robot head. The intrinsic camera parameters such as focal length, resolution, optical center etc., can be estimated using a chess board with a rather good accuracy.

Based on the assumption that the intrinsic and extrinsic camera parameters are known and that we are able to detect the position of an object in both camera images than the 3-D position of an object can be easily calculated using standard vision techniques [17].

2.2 The Control Problem – Virtual Mechanism Approach

The task of the robot head is to keep an object in the center of the camera images. The head has to assure proper gaze direction of both eyes (cameras).

Let us explain the problem and a solution on a simpler planar example. Figure 2 shows an example where the eyes are turned towards the object. When the head moves the orientation of eyes has to change (φ) in order to keep the gaze on the object, even if the object remains still. So the robot task can be defined as the angle of the eye φ, which is a function of the object position and also the eye position. Since the eye position is configuration dependent the task can be defined as:

$$\mathbf{task}_{\mathbf{1\,DOF}} = \varphi = \mathbf{f}(\text{head conf., object pos.}).$$

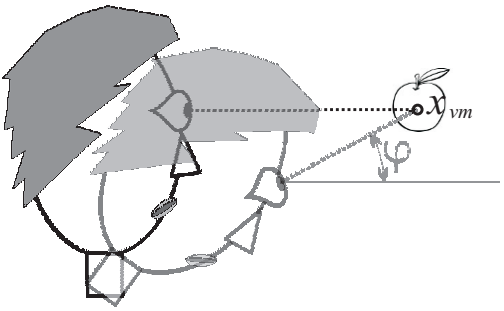


Fig. 2 Gaze direction changes during head movement. Virtual mechanism touches the object.

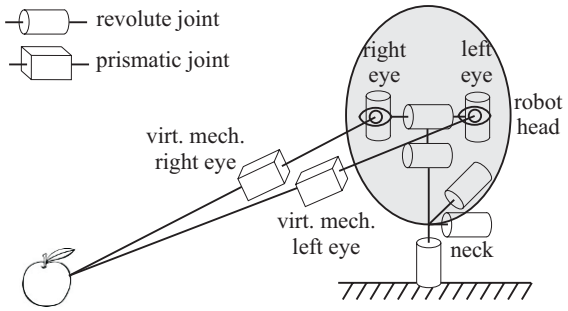


Fig. 3 Schematics of humanoid head enhanced with virtual mechanisms.

In the above task definition *the head configuration is involved in the task definition*. However, this is not the most common way to describe a task – in general, a task is not a function of the robot configuration. Therefore, it is very complex to implement well-known control approaches to control the robot head. To solve this problem in a more systematic way, we propose the use of a virtual mechanism.

Let us expand our humanoid head with an additional virtual link (mechanism) from the robot eye. This virtual link can be treated as an additional prismatic joint which is fixed to the eye (see schematics in Fig. 3). By adding the virtual link we add additional DOF to the system.

The task of the system can now be reconfigured. Instead of controlling the direction of the eye, we can now control the position of the end of the virtual link x_{vm}. It is required that the end of the eye's virtual extension touches the object which is to be tracked (see Fig. 2). The task can now be defined as a simple position tracking problem and is not a function of the head position:

$$\mathbf{task}_{\mathbf{2\ DOFs}} = \boldsymbol{x}_{\mathbf{vm}} = \mathbf{f}(\text{object pos.}).$$

In the original task definition we used 1 DOF to describe the task (φ). By introducing the virtual mechanism we add one additional DOF to the system (the length of the virtual mechanism); however, we also add one DOF to the description of the

task (2 DOFs for the $\boldsymbol{x}_{vm}$ position in plane). By introducing virtual mechanism the degree of redundancy remains the same.

In the spatial 3-D case the circumstances are similar. In this case the task of the robot head is to control the orientation of both cameras and point them directly towards the object. Here, in the original task description the task had four DOFs (two camera angles per eye) and the robot had seven DOF. After the introduction of two virtual mechanisms (one per eye) the number of DOFs of the robot has been increased by two; however, the degree of the task has also been increased by two (positioning task needs three DOFs per eye). The degree of redundancy remains the same.

The main purpose of the virtual mechanism is in the systematization and simplification of the task description. Instead of specifying the desired pointing direction, we can consider the problem as a classic inverse kinematics task, which can be solved by classic control approaches. Note that the angle of the cameras still have to change during the head movement, however, the description of the task (x_{vm}) is not affected by the head movement.

2.3 Controller Design

Having additional virtual mechanism in each eye the kinematics of the head is given by the following form:

$$\boldsymbol{x}_{vm} = \begin{bmatrix} \boldsymbol{x}_{vmL} \\ \boldsymbol{x}_{vmR} \end{bmatrix} = f(\boldsymbol{q}_{head}, l_{virt.m.}), \tag{1}$$

where $\boldsymbol{x}_{vm()}$ denotes the position of the end of the virtual link of the left and right eye, while $\boldsymbol{q}_{head}$ and $l_{virt.m.}$ denote the head joint angles and the lengths of the virtual mechanisms, respectively. To simplify the notation we treat the lengths of the virtual mechanisms as additional joint variables, such that $\boldsymbol{q} = [\boldsymbol{q}_{head}, l_{virt.m.}]$. Note, that some of the head joints only contribute to the left eye (left branch) and some only to the right (it is a kind of branching mechanism).

The relation between the joint and the task velocities is given by the robot Jacobian $\mathbf{J}$:

$$\dot{\boldsymbol{x}}_{vm} = \mathbf{J}\dot{\boldsymbol{q}}. \tag{2}$$

As already stated, the head has more DOFs than needed to accomplish a given task. To achieve good tracking performance while exploiting the redundancy, the following velocity controller can be applied:

$$\dot{\boldsymbol{q}}_c = \mathbf{J}^{\#}\dot{\boldsymbol{x}}_{vm_c} + \mathbf{N}\dot{\boldsymbol{q}}_n, \tag{3}$$

where $\dot{\boldsymbol{q}}_c$ denotes the vector of joint velocities, $\mathbf{J}^{\#}$ is the weighted generalized inverse of $\mathbf{J}$, $\dot{\boldsymbol{x}}_{vm_c}$ is the desired velocity in the task space, $\mathbf{N}$ is the projection onto the null space of $\mathbf{J}$, and $\dot{\boldsymbol{q}}_n$ is the desired joint velocity in the null space.

To control the position of the virtual link, the following $\dot{\boldsymbol{x}}_{vm_c}$ controller is proposed:

$$\dot{\boldsymbol{x}}_{vm_c} = \begin{bmatrix} \dot{\mathbf{r}} + K_p \boldsymbol{e}_L \\ \dot{\mathbf{r}} + K_p \boldsymbol{e}_R \end{bmatrix},$$

where $\boldsymbol{e}_{()}$, $\boldsymbol{e}_{()} = \mathbf{r} - \boldsymbol{x}_{vm()}$, is the task space tracking error for the corresponding eye and $\mathbf{r}$ is the desired task space position, i.e. the position of the object that has to be tracked and is acquired by the wide-angle cameras as indicated in Section 2.1.

The topology of the robot head is a tree-like structure with one main branch and two subbranches (see Fig. 3). In this work both branches have the same priority, which is also the case in humans. Similarly, the symmetry of the head indicates the same priority of the subbranches. To control the branching mechanism the contoller proposed in [12] has been used.

3 Results

The proposed approach has been verified on the real humanoid head described in Section 2 and shown in Fig. 1. Since many visual tasks require both high resolution and a wide field of view our system is equipped with two cameras per eye where the wide-angle camera is placed above the narrow-angle one. Snapshots from both cameras are shown in Fig. 1.

We performed a series of experiments in which a human demonstrator holds a colored object in his hand and move it in front of the humanoid head. We demonstrate the tracking of the narrow-angle cameras based on the wide angle images. These tests have been performed on different distances from the object to demonstrate robustness against changes in distance.

3.1 Object Tracking with Narrow-Angle Cameras

The object position was acquired using information from the wide-angle cameras, while the object was tracked by the narrow-angle cameras. Since both cameras are rigidly connected, the transformation from wide-angle to narrow-angle coordinate frame is constant and was acquired in the cameras calibration process.

Since the optical axes of wide- and narrow-angle cameras are not aligned, it is not trivial to assure that the object acquired by the wide-angle cameras is in the center of the narrow-angle cameras. Here, the distance to the object is crucial. In the proposed approach this is done systematically only by using the appropriate kinematic model – i.e. wide-angle cameras model for the perception and the narrow-angle cameras model for the tracking.

To demonstrate the tracking ability of the narrow-angle cameras, the demonstrator was instructed to move an object on a rectangular trajectory in a plane as shown in the table and Fig. 4:

Time	Object position
0–10 sec	In front of the head
10–20 sec	On the right side of the head
20–30 sec	On the right and up side of the head
30–40 sec	On the left and up side of the head
40–50 sec	On the left side of the head
50–60 sec	In front of the head

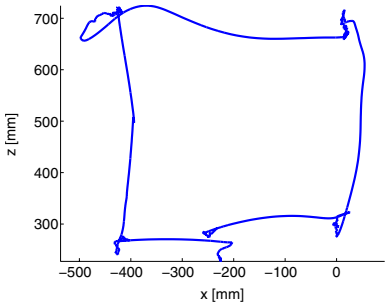


Fig. 4 Movement of the object in front of the head.

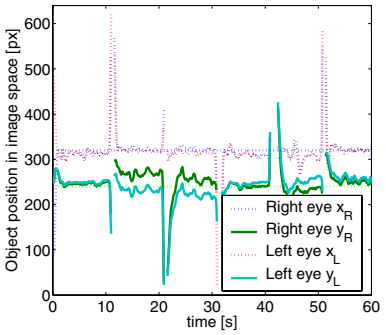


Fig. 5 Object position in narrow-angle images.

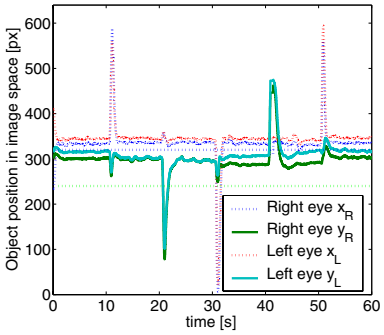


Fig. 6 Object position in wide-angle images.

Using the proposed approach, the head was able to fixate the object in the narrow-angle images (see Fig. 5). Here, the center of the image is at the pixel {320, 240}. However, as expected, the object is *not* in the center of the wide-angle images as shown in Fig. 6, due to the unaligned optical axes.

If the kinematic model of the wide-angle cameras would be used in the control part, then the object would be in the center of wide-angle images instead of in the center of narrow-angle images as can be seen in Figs. 7 and 8. It is only possible to achieve that an object is in the center of both images at one point in space, i. e. at the point where the optical axes of the narrow- and wide-angle cameras intersect. This happens at infinity for systems where optical axes of narrow- and wide-angle cameras are parallel.

To demonstrate the robustness against the distance of the object from the eyes, we performed set of experiments with different distances of object from the cameras. The table below shows the object position in the narrow-angle images. It is clear that the object is in the center regardless of the distance of the object from the cameras:

Obj.-cam. dist. (cm)	30	50	100	200	400
Horiz. obj. pos.	246	245	242	247	259
Vert. obj. pos.	317	316	327	311	318

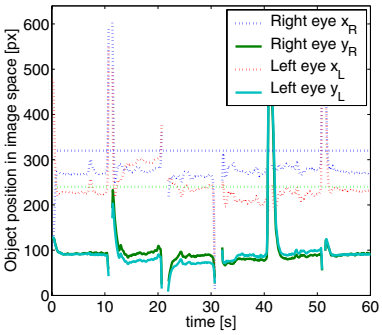


Fig. 7 Object position in narrow-angle images using wide-angle camera kinematics.

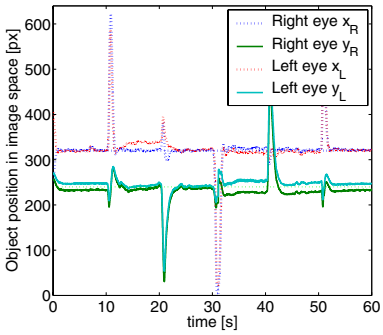


Fig. 8 Object position in wide-angle images using wide-angle camera kinematics.

4 Conclusions

We presented an approach for controlling the gaze direction of a humanoid head. The proposed approach uses wide-angle cameras to acquire 3-D position of an object in space. This information is later used in order to bring an object in the center of the narrow-angle images. To achieve that we introduced a virtual mechanism, that is the main contribution of the paper. The virtual mechanism simplifies the description of the task. This brings us essential simplification in the controller design and results in better tracking performance and ability to exploit redundancy. Experiments on a real robot head are very promising. They confirm simplicity of controller design and show very good tracking performance regardless of the distance of the object from the eyes.

There are many advantages of the proposed approach over the others described in the literature and are listed below:

- Comparing to the decomposed controllers that control each joint individually without considering the complete head kinematics (e.g. [11]), the proposed controller results in a better head motion and brings the object to the image center in a better path.
- Getting an object to the center of narrow-angle images even if it is tracked in wide-angle images is simplified and does not depend on the precise placement of the cameras or on the distance of the object from the cameras.
- The proposed approach indicates how to exploit the redundancy of the head instead of only solving it.
- When using controllers that rely on image-based visual servoing it is more complex to achieve that the desired object is in the center of the narrow-angle images, since the object is sometimes visible only in wide angle images.
- It is a very systematical approach and can be easily implemented on different types of active stereo vision mechanisms.

References

1. D. H. Ballard. Animate vision. *Artificial Intelligence*, **48**, 57–86 (1991).
2. C. Breazeal, A. Edsinger, P. Fitzpatrick, and B. Scassellati. Active vision for sociable robots. *IEEE Transactions on Systems Man and Cybernetics – Part A: Systems and Humans*, **31**(5), 443–453 (2001).
3. B. Scassellati. Eye finding via face detection for a foveated, active vision system. In *Proceedings 15th AAAI/IAAI Conference*, Madison, USA, pp. 969–976 (1998).
4. J. Yamato. Tracking moving object by stereo vision head with vergence for humanoid robot, Master's thesis. Dept. of Elect. Eng. and Comp. Sci., MIT (1998).
5. T. Asfour, K. Regenstein, P. Azad, J. Schröder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann. ARMAR-III: An integrated humanoid platform for sensory-motor control. In *IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids 2006)*, Genoa, Italy (2006).
6. C. G. Atkeson, J. G. Hale, F. Pollick, M. Riley, S. Kotosaka, S. Schaal, T. Shibata, G. Tevatia, A. Ude, S. Vijayakumar, and M. Kawato. Using humanoid robots to study human behavior. *IEEE Intelligent Systems*, **15**(4), 46–56 (2000).
7. G. Metta, A. Gasteratos, and G. Sandini. Learning to track colored objects with log-polar vision. *Mechatronics*, **14**, 989–1006 (2004).
8. F. Panerai, G. Metta, and G. Sandini. Visuo-inertial stabilization in space-variant binocular systems. *Robotics and Autonomous Systems*, 30(1-2):195–214, 2000.
9. A. Bernardino and J. Santos-Victor. Binocular visual tracking: Integration of perception and control. *IEEE Transactions on Robotics and Automation*, **15**(6), 1080–1094, 1999.
10. A. Bernardino, J. Santos-Victor, and G. Sandini. Foveated active tracking with redundant 2D motion parameters. *Robotics and Autonomous Systems*, **39**, 205–221 (2002).
11. A. Ude, C. Gaskett, and G. Cheng. Foveated vision systems with two cameras per eye. In *Proceedings IEEE International Conference on Robotics and Automation*, Orlando, USA, 2006.
12. K.-S. Chang and O. Khatib. Operational space dynamics: Efficient algorithms for modeling and control of branching mechanisms. In *Proceedings International Conference on Robotics and Automation (ICRA 2000)*, San Francisco, CA, pp. 850–856 (2000).
13. O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, **5**(1), 90–98 (1986).
14. L. Žlajpah and B. Nemec. Force strategies for on-line obstacle avoidance for redundant manipulators. *Robotica*, **21**, 633–644 (2003).
15. J. M. Hollerbach. Redundancy resolution of manipulators through torque optimization. *Journal of Robotics and Automation*, **3**(4), 308–316 (1987).
16. J. Lenarčič. On the execution of the secondary task of redundant manipulators. *Robotics and Autonomous Systems*, **30**, 231–236 (2000).
17. E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall (1998).

Tangent Space RRT with Lazy Projection: An Efficient Planning Algorithm for Constrained Motions

Terry Taewoong Um, Beobkyoon Kim, Chansoo Suh and Frank Chongwoo Park

School of Mechanical and Aerospace Engineering,
Seoul National University, Seoul 151-744, Korea;
e-mail: {terryum, beokuni0, cssuh}@robotics.snu.ac.kr, fcp@snu.ac.kr

Abstract. Rapidly-Exploring Random Trees (RRT) have been successfully used in motion planning problems involving a wide range of constraints. In this paper we develop a more robust and efficient version of the constrained RRT planning algorithm of [1]. The key idea is based on first constructing RRTs on tangent space approximations of constraint manifold, and performing lazy projections to the constraint manifold when the deviation exceeds a prescribed threshold. Our algorithm maintains the Voronoi bias property characteristic of RRT-based algorithms, while also reducing the number of projections. Preliminary results of a numerical study, together with a discussion of the potential strengths and weaknesses of our algorithm, are presented.

Key words: Rapidly-exploring random tree, constrained motion planning, lazy projection.

1 Introduction

In order for robots to function autonomously in unstructured environments, they must be able to perform a wide range of constrained motions, *e.g.*, holding a tray with two hands while maneuvering through a dynamically changing obstacle cluttered environment, or pushing a heavy object through a narrow passageway with intermittent gaps, all the while satisfying various constraints on, *e.g.*, the velocities, accelerations, forces and torques on the joints and the object being manipulated, as well as various contact constraints among the robot, manipulated object, and the environment.

A mathematical formulation of the constrained motion planning problem in its most general form is highly complex, and its solution even more challenging. In this paper we focus on a simpler and more tractable version of this problem that focuses on holonomic constraints; the formulation is still reasonably general, and should readily admit extensions to various nonholonomic and other input constraints not explicitly considered here. The setup is as follows. Let $\mathscr{M}$ be configuration space of the robot, with local coordinates q. Denote by $\mathscr{N}$ its task space, with local coordinates X, and let $f : \mathscr{M} \to \mathscr{N}$ denote its forward kinematics; in local coordinates we thus have $X = f(q)$. Finally, let $q_{init} \in \mathscr{M}$ and $q_{final} \in \mathscr{M}$ be the user-supplied ini-

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_27, © Springer Science+Business Media B.V. 2010

tial and final configurations, respectively. The objective then is to seek a continuous path from q_{init} to q_{final} that lies entirely on M, while satisfying a further set of task constraints of the form

$$\psi(X) = 0, \tag{1}$$

$$v(X) \leq 0, \tag{2}$$

where $\psi : \mathcal{N} \rightarrow \Re^p$ and $v : \mathcal{N} \rightarrow \Re^k$ are given and assumed differentiable. Note that $\psi(X) = c$ can also be expressed via the forward kinematics f as $\psi(f(q)) = g(q) = 0$. The inequality constraints can be similarly expressed in the form $v(f(q)) = h(q) \leq 0$. Assuming the forward kinematics is differentiable everywhere, $g(q) = 0$ and those elements of $h(q) \leq 0$ that are active (that is, those elements of $h(q)$ for which strict equality holds) then constitute a set of holonomic constraints on $\mathcal{M}$.

As an illustration, a planar six-bar linkage has a configuration space that can be regarded as a three-dimensional surface embedded in $\Re^6$ (more accurately, a three-dimensional surface embedded in a six-dimensional flat torus). If a large vertical obstacle were present in the middle of the workspace, an inequality constraint can be imposed requiring the robot to remain clear of this obstacle; such a constraint may, e.g., separate the configuration space into two disjoint three-dimensional regions, possibly connected by a two-dimensional surface. The resulting configuration space taking into account all the task constraints may result in a highly complex subset of $\mathcal{M}$.

More generally, trying to determine *a priori* the set of configurations defined by $g(q) = 0$ and $h(q) \leq 0$, especially for robots with a large number of degrees of freedom, is in most cases computationally prohibitive. Moreover, equality constraints of the form $g(q) = 0$ define surfaces that we shall refer to as constraint manifolds; discovering configurations that lie on constraint manifolds via random sampling of the configuration space (e.g., with sampling based planners such as RRTs, or probabilistic road maps that directly sample the configuration space) is extremely unlikely. As pointed out in [1], the lack of prior knowledge of the constraint manifold structure also precludes the use of task space control techniques of the type described in [2] as a complete solution. Alternative approaches, such as first planning a path in a lower-dimensional space (like the constraint manifold) and then attempting to follow this reference path in the full configuration space of the robot [3, 4] suffers from feasibility problems; the lower-dimensional path may not be trackable becuase of joint limits or collisions.

In response to these difficulties, Berson et al. [1] have proposed a constrained bi-directional RRT algorithm (CBiRRT) that addresses the problem of sampling on constraint manifolds (we refer the reader to [5–7] and the references for a survey of RRT methods, and randomized motion planning methods in general). The CBiRRT algorithm first samples in the configuration space $\mathcal{M}$, then uses projection operations to move samples onto constraint manifolds according to some criterion. The projection step can be performed in a number of ways, e.g., the randomized gradient descent method (RGD) [8] originally developed for closed chain motion planning and its extension to general end-effector constraints [9], or the Jacobian pseudo-

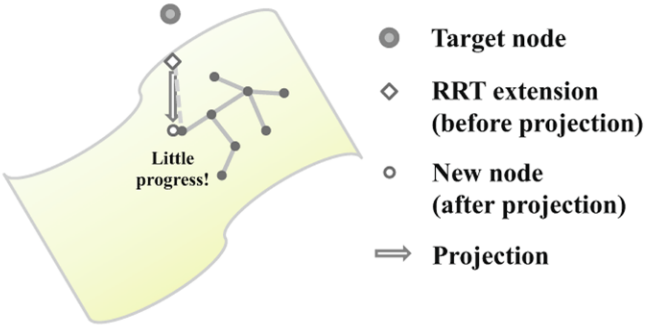


Fig. 1 Example of a target node whose projection contributes very little to extending the tree in the constraint manifold.

inverse projection method of [10]. Whatever the choice of projection method, the fact remains that the manner in which the random configuration is projected to the constraint manifold is in most cases the computational bottleneck, particularly since the Voronoi bias property of RRT algorithms – in a nutshell, the tendency of RRT algorithms to seek out unexplored regions [11] – is considerably diminished. At a more fundamental level, because the shape of the constraint manifold is not considered when generating random samples in the configuration space, performance of the algorithm can be significantly degraded (see Fig. 1).

To address the shortcomings of current RRT-based algorithms for constrained motion planning, in this paper we propose a new algorithm, the Tangent Space RRT (TS-RRT) algorithm. Whereas the CBiRRT algorithm samples target nodes in the configuration space, and projects this node to the constraint manifold, the key idea in our approach is to first sample and construct an RRT in the tangent space of the constraint manifold. When the RRT constructed in the tangent space deviates from the constraint manifold by a certain prescribed threshold, we project to the constraint manifold, construct the tangent space approximation at the projected point, and repeat the above sample-projection procedure. Three additional distinctive features of our algorithm are as follows: (i) we develop a "lazy projection" procedure to reduce the overall number of projections, inspired in part by the lazy collision checking procedure [12] that attempts to reduce the number of collision tests; (ii) we choose the initial direction for RRT construction in the tangent plane to maintain the Voronoi bias toward unexplored regions; (iii) we use local curvature information to scale the tangent space domain over which sampling is performed (for example, nearly zero curvature indicates that the constraint manifold is close to being flat, so that the tangent space approximation is valid over a larger region of the manifold).

2 Tangent Space RRT Algorithm

For space reasons we omit a discussion of the basic RRT construction algorithm; the reader is referred to [5,6] for a review. We shall take the basic RRT construction algorithm as our point of departure, and describe the Tangent Space RRT (TS-RRT) algorithm as an extension of the basic RRT algorithm. Let us first assume that the given q_{init} and q_{final} are already on $\mathscr{M}$. Recall that the main distinguishing feature of TS-RRT is that branches of exploring trees are constructed on the tangent planes of the constraint manifold instead of in the configuration space. Two tangent planes to the constraint manifold are initially constructed at q_{init} and q_{final}. Starting with q_{init} and q_{final} as the root nodes, two trees are then grown on the tangent planes via the basic RRT algorithm. We now discuss the basic components of the TS-RRT algorithm.

Projection by Newton–Raphson Method

We regard $\mathscr{M}$ as a surface embedded in some higher dimensional normed Euclidean space. Given a configuration q not on $\mathscr{M}$, the natural way to compute the distance between q and $\mathscr{M}$ is to find the point $p \in \mathscr{M}$ that minimizes the distance $\|p-q\|$, where $\|\cdot\|$ is a suitably chosen norm on the Euclidean space. Because this nonlinear optimization problem is often difficult and computationally intensive, in practice one settles for easily obtained solutions that approximately minimize the distance function. If the constraint manifold is parametrized implicitly as $g(q)=0$, one easily implementable optimization procedure is to define the error vector e according to

$$e = g(q), \tag{3}$$

i.e., if q lies on $\mathscr{M}$ then e is zero, and nonzero otherwise. For q sufficiently close to $\mathscr{M}$, $\|e\|^2$ is a valid distance function that can be easily minimized via a Newton–Raphson root-finding procedure for $g(q)$ (see Table 1.) Specifically, a first-order Tayler expansion of (3) leads to

$$e + \delta e \simeq g(q) + \frac{\partial g}{\partial q}(q)\delta q. \tag{4}$$

Here we denote the Jacobian matrix $\partial g/\partial q$ by $J(q)$. Setting the right-hand side of (4) to zero and solving for δq, we obtain the update rule

$$q_{new} \leftarrow q - J(q)^{\dagger}\delta e, \tag{5}$$

where $J(q)^{\dagger}$ denotes the pseudo-inverse of $J(q)$:

$$J(q)^{\dagger} = J(q)^{T}[J(q)J(q)^{T}]^{-1}. \tag{6}$$

Table 1 Projection by Newton–Raphson method.

	Projection(q)
1	$e \leftarrow g(q)$
2	**while**($\|e\| < \varepsilon$)
3	$q \leftarrow q - J(q)^{\dagger} e$
4	$e \leftarrow g(q)$
5	**end**
6	**return** q

Random Sampling on Tangent Planes

Given a root node q_{root} lying on $\mathscr{M}$, a basis for the tangent space to $\mathscr{M}$ at q_{root} can be obtained by applying a Gram–Schmidt procedure to the following projection matrix $P(q_{root}) \in \Re^{n \times n}$:

$$P(q_{root}) = I - J(q_{root})^{\dagger} J(q_{root}), \tag{7}$$

where n denotes the dimension of the the configuration space. The basis is used to generate a random sample on the tangent plane; note that the basis needs to be computed only once for each root node. To illustrate the procedure, suppose the constraint equation $g(q) = 0$ consists of m independent equations; in this case the tangent space is of dimension $n - m$, and an orthonormal basis $\{d_1, \ldots, d_{n-m}\}$ can be constructed at each point of $\mathscr{M}$, where each $d_i \in \Re^n$. A random sample q_{rand} on the tangent space can then be generated straightforwardly given q_{root} and the orthonormal basis.

Given a random sample node q_{rand} on the tangent plane, we find the nearest neighbor node q_{near} on the same tangent plane, and extend the tree a fixed length segment in the direction from q_{near} to q_{rand}. The extended node is then tested to see if the inequality constraints are satisfied. If it fails to satisfy the inequality constraints, the node is then abandoned, and the algorithm samples another random node on the tangent plane.

Determining the length of the fixed segment can be done in a number of ways. One effective way uses the local curvature information of the constraint manifold; the basic premise is that at points where the principal curvatures are nearly zero, the manifold is nearly flat, and thus relatively larger steps can be taken in the principal directions without deviating significantly from the manifold.

Creating a New Tangent Plane

When the distance $\|e\|$ from the extended node q_{new} to $\mathscr{M}$ exceeds a certain threshold, denoted $E_{\mathscr{M}}$, the tangent plane no longer approximates $\mathscr{M}$ with the desired accuracy. In such cases we project the extended node onto $\mathscr{M}$ using our previously described projection algorithm in Table 1, and treat the projected point as

Table 2 Create a tangent plane.

CreateTangentPlane(q)	
1	$q \leftarrow \text{Projection}(q)$
2	$P \leftarrow I - J(q)^{\dagger} J(q)$
3	$d \leftarrow q - q_{root,parent}$
4	$d \leftarrow P(q) * d$
5	$TB \leftarrow \text{GramSchmidt}(P)$
6	**return** (TB, q)

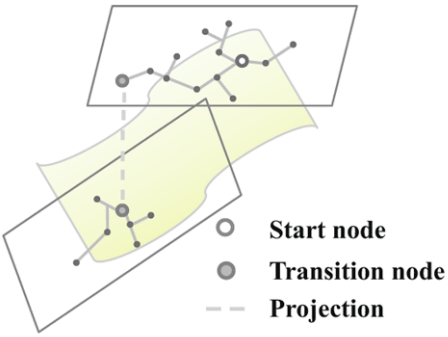


Fig. 2 In TS-RRT, tree branches are generated on tangent planes. When a newly sampled node exceeds a certain prescribed distance from the constraint manifold, that node is projected onto the manifold, and a new tangent plane is created.

a new root node for a new tangent plane generated via another Gram–Schmidt procedure. The process for generating a new root node and the accordant tangent plane is shown in Table 2 and Fig. 2.

When a new tangent plane is created, it is important to ensure that the new tangent plane not significantly overlap with the same area of $\mathscr{M}$ covered by the parent tangent plane. To prevent such overlapping, we restrict the domain for sampling a random node as follows. As described by the third and fourth lines of Table 2, we compute the vector from the root node of the parent tangent plane to the projected node that will be the new root node; the projection of this vector onto the new tangent plane is then used as the initial vector in the Gram–Schmidt process for constructing the tangent basis. In the random sampling phase, only positive weights are associated with this initial basis vector, to ensure that we do not "backtrack" to already explored regions of $\mathscr{M}$.

Selection Bias for Tangent Planes

The trees generated by the TS-RRT algorithm consist of multiple tangent planes and branch nodes, such that every extended node belongs to one of the two trees, and

also to one of the tangent planes. Thus, when we randomly sample nodes, we first need to choose a tangent plane among all of the tangent planes created. To ensure that the tangent planes created later are selected more often, one can use the number of nodes that belong to the tangent plane; that is, if a certain tangent plane has fewer nodes than the others, it has a greater chance of being selected.

Another means of biasing selection tangent planes created later is to use local curvature information of $\mathscr{M}$; the greater the extrinsic curvature, the greater the error in approximating the constraint manifold by the tangent plane (for example, the mean curvature, given by the trace of the second fundamental form, can be used as a bias factor).

Connection Test

Recall that our proposed TS-RRT algorithm is bidirectional, in the sense that two trees – one each emanating from the initial and goal nodes – are simultaneously generated. After every node extension on the constraint manifold, the TS-RRT algorithm tries to connect the extended node to the nearest node on the opposite tree. This is done by evaluating the inner product between the connecting line vector and the rows of Jq (recall that the rows of $J(q)$ are orthogonal to the surface); if the inner products are sufficiently close to zero at both ends, the connecting segment is deemed to be sufficiently close to the tangent planes. In this case uniformly spaced points along the connecting segment are tested to see if the inequality constraints are satisfied. If at any of these points the constraints are violated, or the error $\|e\|$ exceeds the prescribed threshold, no connection is established.

Repeating the process described above, the resulting trees rapidly explore the multiple tangent planes that are used to approximate $\mathscr{M}$. Once a successful connection is established, the final path through the collection of tangent planes is extracted, and the nodes in the final path projected to the constraint manifold (see Fig. 3). The pseudo-code description of our TS-RRT algorithm is given in Table 3.

3 Case Study

3.1 Two-Arm Manipulation

In this example, we consider two arms being pulled from the inside of a drawer to the outside, all the while maintaining a fixed distance between the hands. The task constraint is of the form

$$g(X) = (p_{left} - p_{right})^2 - d^2 = 0, \tag{8}$$

Table 3 TS-RRT algorithm.

	TS-RRT(q_{init}, q_{final})
1	Tree[], TangentPlanes[]
2	Tree.AddNode(q_{init}), Tree.AddNode(q_{final})
3	TangentPlanes.Add(CreateTangentPlane(q_{init}))
4	TangentPlanes.Add(CreateTangentPlane(q_{final}))
5	**while** $i < I_{max}$ **do** $i \leftarrow i+1$
6	$k \leftarrow$ SelectTangentPlane()
7	$q_{rand} \leftarrow$ RandomSampleOnTangentPlane(TangentPlanes[k])
8	$q_{near} \leftarrow$ NearestNode(k, q_{rand})
9	$q_{new} \leftarrow$ Extend(q_{near}, q_{rand})
10	**if** $h(f(q_{new})) > 0$ **then** goto **3 end**
11	**if** Connect(q_{new}) = Success **then**
12	Path $\leftarrow$ ExtractPath()
13	**return** LazyProjection(Path)
14	**else if** $q_{new} > E_{\mathcal{M}}$ **then**
15	TangentPlanes.Add(CreateTangentPlane(q_{new}))
16	**end**
17	Tree.AddNode(q_{new}), Tree.AddEdge(q_{near}, q_{new})
18	**end**
19	**return** Fail

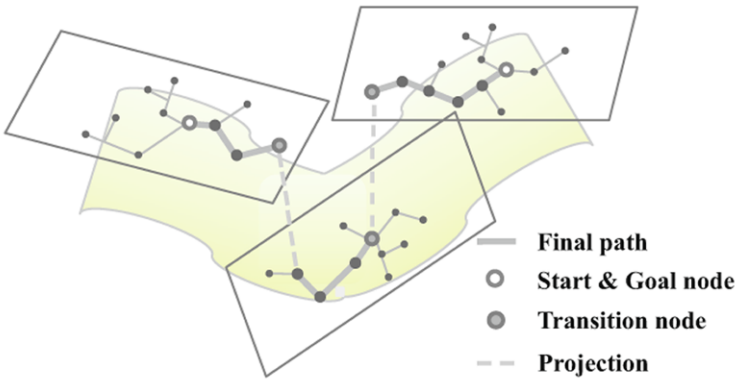


Fig. 3 The final path is extracted through multiple tangent planes. The actual path on the manifold can be obtained by projecting the nodes on the final path.

where p_{left} and p_{right} denote the Cartesian positions of the left and right hands of the robot, respectively. Collision avoidance between the robot arms and the drawer are described by an appropriate set of inequality constraints.

Preliminary tests of the TS-RRT algorithm suggest that it works more reliably and efficiently than the CBiRRT algorithm, but more extensive testing under a wider range of scenarios needs to be performed before firmer conclusions can be drawn.

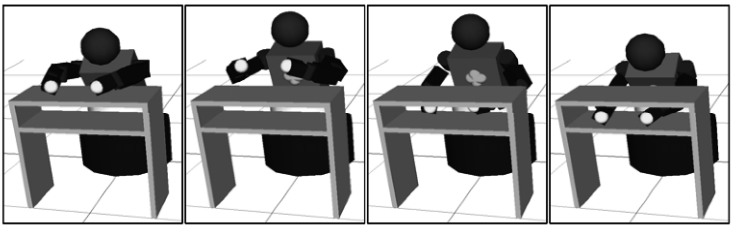


Fig. 4 Constraint on distance between two hands.

4 Conclusions and Future Work

This paper has presented a new randomized algorithm for task constrained motion planning, the TS-RRT algorithm, that is based upon the widely used concept of rapidly exploring random trees (RRT). Unlike existing RRT algorithms for constrained motion planning, the distinguishing feature of the TS-RRT algorithm is that RRTs are constructed in the tangent spaces of the constraint manifold, and projected to the manifold only when the newly sampled nodes exceed a certain threshold distance. Our method can be contrasted with existing approaches that project randomly sample in the configuration space, and project every sample back to the constraint manifold; such a procedure can result in nodes that only extend the tree minimally, possibly revisiting previously explored directions.

Preliminary numerical studies with our algorithm suggest that it is more reliable and computationally efficient than existing algorithms; more extensive testing is currently underway to compare its performance with the existing CBiRRT and other algorithms for constrained motion planning. More effective methods for exploiting local curvature information about the constraint manifold, as well as investigating ways to include, e.g., dynamics, nonholonomic constraints, and actuator force-torque limits into the planning framework, are also being investigated.

References

1. Berson, D., Srinivasa, S., Ferguson, D., and Kuffner, J., Manipulation planning on constraint manifolds, in *Proc. IEEE Int. Conf. Robotics and Automation* (2009).
2. Sentis, S. and Khatib, O., Synthesis of whole-body behaviors through hierarchical control of behavioral primitives, *Int. J. Humanoid Robots* (2005).
3. Koga, Y., Kondo, K., Kuffner, J., and Latombe, J., Planning motions with intentions. In *SIGGRAPH* (1994).
4. Yamane, K., Kuffner, J., and Hodgins, J., Synthesizing animations of human manipulation tasks. In *ACM Transactions on Graphics* (2004).
5. LaValle, S., Rapidly-exploring random trees: a new tool for path planning, TR 98-11, Computer Science Department, Iowa State University (1998).
6. LaValle, S. and Kuffner, J., Rapidly exploring random trees: progress and prospects. In *Workshop on Algorithmic Foundations of Robotics* (2000).

7. Kavraki, L., Svestka, P., Latombe, J., and Overmars, M., Probabilistic roadmaps for path planning in high-dimensional configuration spaces. In *Proc. IEEE Int. Conf. Robotics Automation* (2002).
8. LaValle, S., Yakey, J., and Kavraki, L., A probabilistic roadmap approach for systems with closed kinematic chains. In *Proc. IEEE Int. Conf. Robotics and Automation* (1999).
9. Yao, Z. and Gupta, K., Path planning with general end-effector constraints: Using task space to guide configuration space search. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems* (2005).
10. Stilman, M., Task constrained motion planning in robot joint space. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems* (2007).
11. Lindermann, S. and LaValle, S., Incrementally reducing dispersion by increasing Voronoi bias in RRTs. In *Proc. IEEE Int. Conf. Robotics and Automation* (2004).
12. Sanchez, G. and Latombe, J., A single query bi-directional probabilistic roadmap planner with lazy collision checking. In *Int. Symp. Robotics Research*, Lorne, Victoria, Australia (2001).

PART 5

Equilibrium Analysis of Tensegrity Structures with Elastic Ties

O. Anubi and C. Crane

University of Florida, Gainesville, Florida, USA; e-mail: {anubi, ccrane}@ufl.edu

Abstract. In this paper, the equilibrium analysis of a passive tensegrity structure with elastic connecting ties is addressed. The problem is modeled as a constrained optimization of the total potential energy of the system and the Lagrange multiplier method is used to deduce the equilibrium equations. The homotopy continuation method is used to completely solve the resulting nonlinear polynomial system and all the equilibrium positions are identified. Furthermore, the stability analysis of the equilibrium configurations is carried out using the second order condition and higher order condition for cases where the second order condition fails. One numerical example is provided to demonstrate the analysis

Key words: Tensegrity, equilibrium, smart structures, mechanisms.

1 Introduction

Tensegrity structures are formed entirely by a combination of purely rigid (struts) and purely flexible (ties) members. The struts are always in compression while the ties (both elastic and inelastic) are in tension. The entire configuration stands by itself and maintains its form solely because of the internal arrangements of the ties and struts [1] which is such that no pair of struts touch and the end of each strut is connected to three non-coplanar ties [2]. The word "tensegrity" is a combination of two words; tension and integrity [3, 4].

Their ability to be at equilibrium under several kinematic configurations has made them a subject of interest to many researchers. Kenner [5] established the relation that defines the configuration of regular tensegrity prisms. Tobie [1] presented procedures for the generation of tensile structures by physical and graphical means. Yin [2] obtained Kenner's result using energy considerations and found the equilibrium position for the unloaded tensegrity prisms. Stern [6] developed generic design equations to find the lengths of the struts and ties needed to create a desired geometry. Correa et al. [7] used the principle of virtual work to address the problem of the determination of the equilibrium position of a tensegrity structure when external forces and external moments act on the structure and verified the results obtained

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_28,

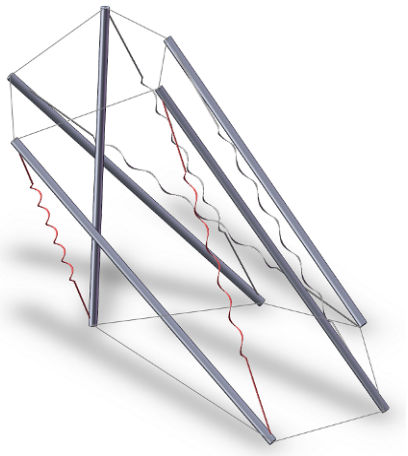


Fig. 1 Five strut tensegrity structure with elastic connecting ties.

using a method based on Newton's Third Law.

The stiffness of tensegrity structures has also been of interest to many researchers. Roth and Whiteley [8] extended rigidity concepts of bar frameworks to tensegrity. They also set up some important implications between tensegrity frameworks and their equivalent bar frameworks. Their results were applied to both planar and spatial tensegrity systems. Connelly [9] studied some important properties of rigid tensegrity frameworks using an energy approach which led to a redefinition of the rigidity concept based on positive definiteness of the stress matrix associated with the stress vector. Further research in this area was carried out by Pellegrino, Calladine, and Tibert [10–15] who developed an algebraic method to determine the number of mechanisms and equilbrium configurations as well as a base for the subspaces of mechanisms, self-stress states, resolvable, and non-resolvable forces. They also presented a method to distinguish between first-order infinitesimal mechanisms and higher order infinitesimal or finite mechanisms.

In this paper, the problem of determining the equilibrium positions for a passive tensegrity structure with elastic connecting ties is examined. Fig. 1 shows a typical five strut tensegrity structure with elastic connecting ties. The word passive is used here to mean that no active forces or moments act on the structure. The system is only passively maintained in equilibrium by internal spring forces. The problem is modeled as a constrained optimization problem and the solution is advanced using the Lagrange multiplier method. Moreover, the idea of rigidity as used here is the propensity of a structure to return to its equilibrium state if subjected to a small displacement away from the equilibrium configuration. The rigidity of the structure at different equilibrium configurations is studied by examining the positive definiteness of the Hessian matrix of the Lagrangian function.

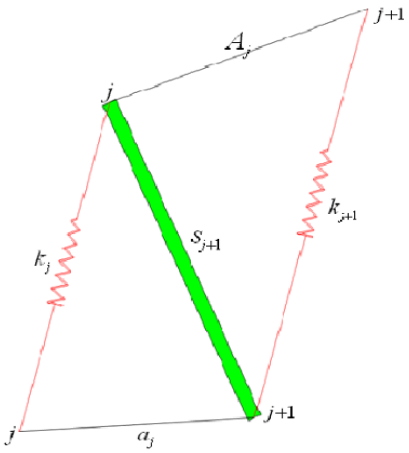


Fig. 2 Tensegrity elements.

2 Problem Formulation

Figure 2 shows two neighboring top nodes j, $j+1$ and the corresponding bottom nodes.

Let
$[X_j, Y_j, Z_j]^T$, $j = 1, \ldots, n$ be the coordinates of the top nodes
$[x_j, y_j, z_j]^T$, $j = 1, \ldots, n$ be the coordinates of the bottom nodes
A_j be the length of the non-elastic top tie connecting the nodes $j, j+1$
a_j be the length of the non-elastic bottom tie connecting the nodes $j, j+1$
k_j be the spring constant of the spring j connecting top and bottom nodes j
s_{j+1} be the length of the strut connecting top node j and bottom node $j+1$
$\mathbf{P}_j = [X_j, Y_j, Z_j]^T$ be the position vector of top node j
$\mathbf{p}_j = [x_j, y_j, z_j]^T$ be the position vector of bottom node j

The following assumptions are also made without loss of generality:

- The struts are massless.
- No external wrench is acting on the system.
- All ties are in tension at an equilibrium position.
- All springs have zero free lengths. This assumption is valid because zero-free length springs can be created by extending the actual springs beyond their equilibrium attachment points [16–18].

In addition to the above constraints, there are constraints imposed on the geometry of the problem due to the known lengths of the top and bottom ties as well as the strut lengths. These are termed geometric constraints for the purpose of this paper. Following the virtual work principle employed in [7] and the assumption that no

external wrench acts on the system, it can be inferred that the equilibrium configuration of the structure is such that the potential energy function is stationary. The following subsections details the formulation of the equilibrium configuration problem as an optimization problem.

2.1 Geometric Constraint

Connelly et al. [19] considered three connecting elements in their analysis of the rigidity of tensegrity frameworks. According to them; cables impose an upper bound constraint on the distance between the pair of points they connect, struts, with complementary behavior to cables, impose lower bound constraints, and bars maintain fixed lengths. However, here the idea of struts is the same as the bars used in [19] and springs are used in place of their ties. Springs impose no constraints on the distance between the nodes they connect but tend to keep them together as much as possible by preventing any attempt to separate them with an opposition which is dependent on their spring constant and the distance of separation.

From Fig. 2 these geometric constraints can be written as follows:

$$\begin{aligned} \|\mathbf{P}_{j+1} - \mathbf{P}_j\| &\leq A_j \\ \|\mathbf{p}_{j+1} - \mathbf{p}_j\| &\leq a_j \\ \|\mathbf{P}_j - \mathbf{p}_{j+1}\| &= s_{j+1} \end{aligned} \tag{1}$$

where $\|\cdot\|$ denotes the Euclidean norm operator, $\mathbf{P}_j$ and $\mathbf{P}_{j+1}$ refer to the points at the top of the tensegrity element shown in Fig. 1, and $\mathbf{p}_j$ and $\mathbf{p}_{j+1}$ refer to points at the bottom of the tenesegrity element shown in Fig. 1. Expanding (1) yields:

$$(X_{j+1} - X_j)^2 + (Y_{j+1} - Y_j)^2 + (Z_{j+1} - Z_j)^2 \leq A_j^2 \tag{2}$$

$$(x_{j+1} - x_j)^2 + (y_{j+1} - y_j)^2 + (z_{j+1} - z_j)^2 \leq a_j^2 \tag{3}$$

$$(X_j - x_{j+1})^2 + (Y_j - y_{j+1})^2 + (Z_j - z_{j+1})^2 = s_{j+1}^2 \tag{4}$$

2.2 Potential Energy

Since the struts are assumed massless, the effect of gravitational potential is neglected and the total potential energy becomes the total energy stored in the springs which is given by

$$u = \frac{1}{2} \sum_{j=1}^{n} k_j \|\mathbf{P}_j - \mathbf{p}_j\|^2 \tag{5}$$

where n is the number of top and bottom nodes

2.3 Problem Statement

The problem statement in optimization form is as follows:

Given: A_j, a_j, k_j

Find: $X_j, Y_j, Z_j, x_j, y_j, z_j$
such that:

$$u = \frac{1}{2}\sum_{j=1}^{n} k_j((X_j - x_j)^2 + (Y_j - y_j)^2 + (Z_j - z_j)^2)$$

is an extremum subject to:

$$\begin{aligned} g_j^1 &= (X_{j+1} - X_j)^2 + (Y_{j+1} - Y_j)^2 + (Z_{j+1} - Z_j)^2 - A_j^2 + M_j^2 = 0 \\ g_j^2 &= (x_{j+1} - x_j)^2 + (y_{j+1} - y_j)^2 + (z_{j+1} - z_j)^2 - a_j^2 + m_j^2 = 0 \\ g_j^3 &= (X_j - x_{j+1})^2 + (Y_j - y_{j+1})^2 + (Z_j - z_{j+1})^2 - s_{j+1}^2 = 0 \\ j &= 1, 2, \ldots, n \end{aligned}$$

where $M_j^2, m_j^2 \in \Re$ are slack variables. One advantage of this formulation, apart from being amendable to several optimization techniques, is that the need for the substitution of tan-half angle [20] is eliminated.

3 Solution Approach

The solution method employed is the Lagrange multiplier method. By introducing Lagrange multipliers λ_j^1, λ_j^2 and λ_j^3 , the weighted objective function becomes:

$$w = \sum_{j=1}^{n}\left[\frac{k_j}{2}((X_j - x_j)^2 + (Y_j - y_j)^2 + (Z_j - z_j)^2) + \lambda_j^1 g_j^1 + \lambda_j^2 g_j^2 + \lambda_j^3 g_j^3\right] \quad (6)$$

The stationary values of (6) occur when

$$\frac{\partial w}{\partial X_j} = \frac{\partial w}{\partial Y_j} = \frac{\partial w}{\partial Z_j} = \frac{\partial w}{\partial x_j} = \frac{\partial w}{\partial y_j} = \frac{\partial w}{\partial z_j} = \frac{\partial w}{\partial M_j} = \frac{\partial w}{\partial m_j} = g_j^1 = g_j^2 = g_j^3 = 0 \quad (7)$$

First, $\partial w/\partial M_j = \partial w/\partial m_j = 0$ implies that

$$\begin{aligned} 2\lambda_j^1 M_j &= 0 \\ 2\lambda_j^2 m_j &= 0 \end{aligned} \quad (8)$$

According to (8), either $M_j, m_j = 0$ in which case the constraints g_j^1 and g_j^2 given in (1) become strict equality constraints or $\lambda_j^1, \lambda_j^2 = 0$ in which case the constraints become degenerate. The former, i.e. $M_j, m_j = 0$, must in general be the case, except in special circumstances which will not be considered here.

The partials $\partial w/\partial X_j$, $\partial w/\partial Y_j$, and $\partial w/\partial Z_j$ may now be written as

$$\frac{\partial w}{\partial X_j} = \frac{\partial u}{\partial X_j} + \lambda_{j-1}^1 \frac{\partial g_{j-1}^1}{\partial X_j} + \lambda_j^1 \frac{\partial g_j^1}{\partial X_j} + \lambda_j^3 \frac{\partial g_j^3}{\partial X_j} = 0$$

$$\frac{\partial w}{\partial Y_j} = \frac{\partial u}{\partial Y_j} + \lambda_{j-1}^1 \frac{\partial g_{j-1}^1}{\partial Y_j} + \lambda_j^1 \frac{\partial g_j^1}{\partial Y_j} + \lambda_j^3 \frac{\partial g_j^3}{\partial Y_j} = 0$$

$$\frac{\partial w}{\partial Z_j} = \frac{\partial u}{\partial Z_j} + \lambda_{j-1}^1 \frac{\partial g_{j-1}^1}{\partial Z_j} + \lambda_j^1 \frac{\partial g_j^1}{\partial Z_j} + \lambda_j^3 \frac{\partial g_j^3}{\partial Z_j} = 0$$

which yields

$$\begin{aligned}
&k_j(X_j - x_j) + 2\lambda_{j-1}^1(X_j - X_{j-1}) - 2\lambda_j^1(X_{j+1} - X_j) + 2\lambda_j^3(X_j - x_{j+1}) = 0 \\
&k_j(Y_j - y_j) + 2\lambda_{j-1}^1(Y_j - Y_{j-1}) - 2\lambda_j^1(Y_{j+1} - Y_j) + 2\lambda_j^3(Y_j - y_{j+1}) = 0 \\
&k_j(Z_j - z_j) + 2\lambda_{j-1}^1(Z_j - Z_{j-1}) - 2\lambda_j^1(Z_{j+1} - Z_j) + 2\lambda_j^3(Z_j - z_{j+1}) = 0
\end{aligned} \tag{9}$$

Equation (9) can be expressed in matrix format as:

$$\Delta_j \begin{bmatrix} -2\lambda_{j-1}^1 \\ 2\lambda_j^1 \\ -2\lambda_j^3 \end{bmatrix} = k_j \Lambda_j \tag{10}$$

where

$$\Delta_j = \begin{bmatrix} X_j - X_{j-1} & X_{j+1} - X_j & X_j - x_{j+1} \\ Y_j - Y_{j-1} & Y_{j+1} - Y_j & Y_j - y_{j+1} \\ Z_j - Z_{j-1} & Z_{j+1} - Z_j & Z_j - z_{j+1} \end{bmatrix} \tag{11}$$

and

$$\Lambda_j = \begin{bmatrix} X_j - x_j \\ Y_j - y_j \\ Z_j - z_j \end{bmatrix} \tag{12}$$

Using Cramer's rule

$$\begin{bmatrix} -2\lambda_{j-1}^1 \\ 2\lambda_j^1 \\ -2\lambda_j^3 \end{bmatrix} = \frac{k_j}{|\Delta_j|} \begin{bmatrix} |\Delta_j^1| \\ |\Delta_j^2| \\ |\Delta_j^3| \end{bmatrix} \tag{13}$$

Δ_j^1, Δ_j^2 and Δ_j^3 are obtained from Δ_j by replacing columns 1, 2 and 3 by Λ_j respectively and $|A|$ is used to denote the determinant of a square matrix A. Increasing the index j of (13) by 1 gives

$$\begin{bmatrix} -2\lambda_j^1 \\ 2\lambda_{j+1}^1 \\ -2\lambda_{j+1}^3 \end{bmatrix} = \frac{k_{j+1}}{|\Delta_{j+1}|} \begin{bmatrix} -|\Delta_{j+1}^1| \\ |\Delta_{j+1}^2| \\ |\Delta_{j+1}^3| \end{bmatrix} \tag{14}$$

Comparing (13) and (14)

$$-\frac{k_{j+1}|\Delta_{j+1}^1|}{|\Delta_{j+1}|} = \frac{k_j|\Delta_j^2|}{|\Delta_j|}$$

or

$$k_j|\Delta_j^2||\Delta_{j+1}| + k_{j+1}|\Delta_{j+1}^1||\Delta_j| = 0 \tag{16}$$

$$j = 1,2,\ldots,n.$$

It can be shown that for a particular choice of coordinate system, $\mathbf{p}_j$ can be fully determined for cases where $n \leq 4$. Hence, (16) together with the geometric equality constraints (2) and (4) constitute a complete system of nonlinear polynomial equations whose solutions define all possible equilibrium configurations for the tensegrity structure. However, for cases where $\mathbf{p}_j$ is not fully determined, it is necessary to obtain additional equations to balance with the number of unknowns. This is easily obtained by following a similar argument as (9) through (16) for the subset

$$\frac{\partial w}{\partial x_j} = \frac{\partial w}{\partial y_j} = \frac{\partial w}{\partial z_j} = 0$$

of (7) where one obtains

$$k_j|\delta_j^3||\delta_{j+1}| - k_{j+1}|\delta_{j+1}^2||\delta_j| = 0 \tag{17}$$

$$j = 1,2,\ldots,n$$

where

$$\delta_j = \begin{bmatrix} X_{j-1}-x_j & x_j-x_{j-1} & x_{j+1}-x_j \\ Y_{j-1}-y_j & y_j-y_{j-1} & y_{j+1}-y_j \\ Z_{j-1}-z_j & z_j-z_{j-1} & z_{j+1}-z_j \end{bmatrix} \tag{18}$$

and δ_j^1, δ_j^2 and δ_j^3 are obtained from δ_j by replacing first, second and third columns by Λ_j respectively.

4 Numerical Example

Figure 3 shows a case where $n = 3$. The following numerical values are given, where lengths are in units of cm and spring stiffness values are in units of N/cm:

$a_1 = 10, a_2 = 12.3, a_3 = 15, A_1 = 11.5, A_2 = 9, A_3 = 10,$
$s_1 = 20, s_2 = 23, s_3 = 19.5,$
$k_1 = 3.8, k_2 = 3, k_3 = 4.3.$

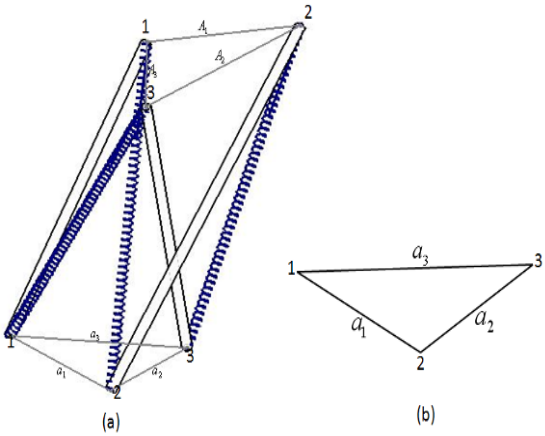


Fig. 3 Numerical example.

Here, since all the bottom nodes have zero z-coordinates, their position vectors are known. There are two possible configurations for the bottom nodes and the overall equilibrium configurations will be symmetric for the two cases. So picking either of them is sufficient for this analysis. Thus

$$\mathbf{p}_1 = \begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix}^T = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$$
$$\mathbf{p}_2 = \begin{bmatrix} x_2 & y_2 & z_2 \end{bmatrix}^T = \begin{bmatrix} a_1 & 0 & 0 \end{bmatrix}^T$$
$$\mathbf{p}_3 = \begin{bmatrix} x_3 & y_3 & z_3 \end{bmatrix}^T = \begin{bmatrix} a_3 \cos\alpha_1 & a_3 \sin\alpha_1 & 0 \end{bmatrix}^T$$

where α_1 is the angle between ties 1–2 and 1–3 at node 1. The unknown coordinates of the top nodes are written as: $\mathbf{P}_j = \begin{bmatrix} X_j & Y_j & Z_j \end{bmatrix}^T$, $j = 1,2,3$.

Thus for this case the matrix expression in (16) gives three equations in the nine unknowns (the coordinates of the top points). The geometric constraints in Eqs. (2) and (4) for the $n = 3$ case give another six equations in the nine unknowns. A homotopy continuation method was used to solve the set of equations and a total of 10 different equilibrium equations were obtained. A second order stability analysis was conducted on each of the ten cases to determine if each represented a stable or unstable equilibrium case. Seven of the configurations have indefinite Hessians, two of them are predominantly unstable with negative semidefinite Hessian, and only one is predominantly stable with positive semidefinite Hessian. The numerical values are not shown here due to length considerations.

5 Conclusions

The equilibrium analysis of spatial tensegrity structures with elastic connecting ties was considered. The solution was approached by writing a Lagrangian energy function for the constrained optimization model of the problem. Slack variable were introduced to accommodate inequality constraints imposed by the cables. It was shown that the equilibrium is achieved when all cables are fully stretched and the corresponding equilibrium equations were deduced using the Lagrange multiplier method. Next, the Continuation method was used to numerically evaluate all equilibrium configurations for the system. Lastly, a stability study of the various equilibrium configurations was carried out using the second and higher order conditions, although this work was not detailed here due to length considerations.

Acknowledgements The authors would like to gratefully acknowledge the support provided by the Department of Energy via the University Research Program in Robotics (URPR), grant number DE-FG04-86NE37967.

References

1. Tobie, R.S.: A report on an inquiry into the existence, formation and representation of tensile structures. Master of Industrial Design Thesis, Pratt Institute, New York (1976).
2. Yin, J.P.: An analysis for the design of self-deployable tensegrity and reinforced tensegrity prisms with elastic ties. Ph.D. Dissertation, University of Florida, Gainesville, FL (2000).
3. Edmondson, A., *A Fuller Explanation: The Synergetic Geometry of R. Buckminster Fuller*. Birkhäuser, Boston (1987).
4. Fuller, R., *Synergetics: The Geometry of Thinking*. MacMillan Publishing, New York (1975).
5. Kenner, H.: *Geodesic Math and How to Use It*. University of California Press, Berkeley (1976).
6. Stern, I.: Development of design equations for self-deployable N-strut tensegrity systems. M.S. Thesis, University of Florida, Gainesville, FL (1999).
7. Correa, J., Crane, C., Duffy, J.: Static analysis of tensegrity structures. *ASME Journal of Mechanical Design*, **127**(2), 257–268 (2005).
8. Roth, B., Whitely, W.: Tensegrity frameworks. *Transactions of the American Mathemetical Society*, **265**, 419–446 (1981).
9. Connelly, R.: Rigidity and energy. *Inventiones Mathematicae*, **66**, 11–33 (1982).
10. Pellegrino, S., Calladine, C.: Two step matrix analysis of prestresssed cable nets. In: *Proceedings of the Third International Conference of Space Structures* (1984).
11. Pellegrino, S.: Matrix analysis of statically and kinematically indeterminate frameworks. *International Journal of Solids and Structures*, **22**(4), 409–428 (1986).
12. Pellegrino, S.: Analysis of prestressed mechanisms. *International Journal of Solids and Structures*, **26**(12), 1329–1350 (1989).
13. Calladine, C.: First order infinitesimal mechanisms. *International Journal of Solids and Structures*, **27**(4), 505–515 (1991).
14. Calladine, C., Pellegrino, S.: Further remarks on first order infinitesimal mechanisms. *International Journal of Solids and Structures*, **29**, 2119–2122 (1992).
15. Tibert, A., Pellegrino, S.: Review of form-finding methods for tensegrity structures. *International Journal of Solids and Structures*, **18**(4), 209–223 (2003).

16. Arsenault, M., Gosselin, C.: Static balancing of tensegrity mechanisms. *ASME Journal of Mechanical Design*, **129**, 295–300 (2007).
17. Streit, D., Gilmore, B.: Perfect spring equilibrators for rotable bodies. *ASME Journal of Mechanisms, Transmissions, Automation in Design*, **111**(4), 451–458 (1989).
18. Herder, J.: Energy-free systems: Theory, conception and design of statically balanced spring mechanisms. Ph.D. Dissertation, Delft University of Technology, Delft, FL (2001).
19. Connelly, R., Whiteley, W.: Second order rigidity and prestress stability for tensegrity frameworks. *Discrete Mathematics*, **9**(3), 453–491 (1996).
20. Bayat, J., Crane, C.: Closed-form equilibrium analysis of planar tensegrity structures. In: *Proceedings of the ASME 2007 International Design Engineering Technical Conferences and Computers and Information Engineering Conference*, Las Vegas (2007).

Singularity Analysis of Lower-Mobility Parallel Robots with an Articulated Nacelle

Semaan Amine, Daniel Kanaan, Stéphane Caro and Philippe Wenger

Institut de Recherche en Communications et Cybernétique de Nantes, France; e-mail: {amine, kanaan, caro, wenger}@irccyn.ec-nantes.fr

Abstract. This paper presents a generic approach to analyze the singularity of robots with an articulated nacelle like the $H4$ robot. Using screw theory, the concept of equivalent twist graph is introduced in order to characterize the constraint wrenches and the actuation wrenches applied to the moving platform. Using Grassmann–Cayley Algebra, the geometric conditions associated with the dependency of six Plücker vectors of finite and infinite lines in the projective space $\mathbb{P}_3$ are reformulated in the superbracket decomposition in order to characterize geometrically the parallel singularities of the robot.

Key words: Grassmann–Cayley Algebra, superbracket, screw theory, parallel manipulators, singularities, twist graph, nacelle.

1 Introduction

Parallel singularities are critical configurations in which a parallel manipulator loses its stiffness and gains one or more degrees of freedom (DOF). They can be found using either numerical, analytical or geometrical methods. For Gough Stewart parallel manipulators, the rows of the inverse Jacobian matrix are Plücker coordinate vectors of six finite lines that are six actuation forces applied by the actuators to the moving platform. The parallel singularities of such manipulators that Merlet [1] analyzed using Grassmann line geometry occur when those lines become linearly dependent. Ben-Horin and Shoham analyzed the parallel singularities of 6-DOF parallel manipulators using Grassmann-Cayley Algebra (GCA) [2, 3]. Contrary to 6-DOF parallel manipulators, the legs of a lower-mobility parallel manipulator have a connectivity smaller than six and, in turn, each leg constrains partly the motion of the moving platform. When the legs lose their ability to constrain the moving platform, a so-called constraint singularity occurs [4]. Kanaan et al. [5] enlarged the application of GCA to lower-mobility manipulators, in which the rows of the 6×6 inverse Jacobian are not necessarily finite lines (zero pitch wrenches) but can be also infinite lines (infinite pitch wrenches). These wrenches, also known as governing lines, are actuation and constraint wrenches applied to the moving platform. Choi et al. [6] analyzed the singularities of the $H4$ robot using screw theory, while Wu et al. [7] analyzed them using line geometry.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_29, © Springer Science+Business Media B.V. 2010

In this paper, we present a methodology for the singularity analysis of robots with an articulated nacelle. This methodology is general and applied to the $H4$ robot. To deal with robots with complex architectures like the $H4$ robot, we introduce the concept of *twist graph*. Each leg of this robot contains a closed loop (the 4S parallel linkage) and can be replaced by a virtual serial chain having the same twist system as the leg in the equivalent twist graph of the robot. The twist graph of the $H4$ robot is obtained in order to characterize the constraint and actuation wrenches applied to its end effector. Then, the parallel singularity conditions of the robot are derived using its wrench diagram in the projective space $\mathbb{P}_3$ and the superbracket decomposition. Finally, some singularity configurations are illustrated geometrically.

2 Grassmann–Cayley Algebra in the Projective Space $\mathbb{P}_3$

The Grassmann–Cayley Algebra (GCA), also known as exterior algebra, was developed by Grassmann as a calculus for linear varieties operating on extensors with the operators *join* and *meet* associated with the union and intersection of vector spaces of extensors. The bracket is a determinant that satisfies special product relations called *sygzygies*. In the projective space $\mathbb{P}_3$, extensors could represent entities such as points, lines or planes. Points are represented with their homogeneous coordinates while lines and planes are represented with their Plücker coordinates. The notion of extensor makes it possible to work at the symbolic level, and therefore, to produce coordinate-free algebraic expressions for the geometric singularity conditions of spatial parallel manipulators. For further details on GCA the reader is referred to [2, 8].

3 Theory of Reciprocal Screws

Screw theory is suitable for the type synthesis and the study of the instantaneous motion of parallel manipulators. The principle of reciprocal screws was studied in [9–11] and then developed in [12–15]. The method deals with the reciprocity condition between two screw systems to characterize for each leg-chain, the constraint wrench system reciprocal to its twist system as well as the actuation wrench system obtained by locking actuators. In a non-singular configuration, constraint wrenches and actuation wrenches form a 6-system.

A *twist* is a screw representing the instantaneous motion of a rigid body, a *wrench* is a screw representing a system of forces and moments acting on a rigid body. Let ε denotes a twist and τ denotes a wrench. An infinite pitch twist ε_∞ represents a pure translation of a prismatic joint P and a zero pitch twist ε_0 represents a pure rotation of a revolute joint R. A pure force constrains the translation along its line direction and is represented by a zero pitch wrench τ_0. A pure moment constrains the rotation about its direction and is represented by an infinite pitch wrench τ_∞. A zero pitch screw represents the Plücker coordinate vector of a finite line in $\mathbb{P}_3$. An infinite pitch screw represents the Plücker coordinate vector of an infinite line in $\mathbb{P}_3$.

A *screw system* of order n ($0 \leq n \leq 6$), also called n-system, comprises all the screws that are linearly dependent on n given linearly independent screws. Any set of linearly independent screws within a n-system forms a basis of this system. A n-screw system can be replaced by another equivalent n-system by applying a linear transformation to the basis of the first one [16]. The twist system $\mathtt{T}^i$ and the wrench system $\mathtt{W}^i$ of a serial kinematic chain composed of f joints are given by:

$$\mathtt{T}^i = \bigoplus_{j=1}^{f} \mathtt{T}_j \quad , \quad \mathtt{W}^i = \bigcap_{j=1}^{f} \mathtt{W}_j \tag{1}$$

The twist system $\mathtt{T}$ and the wrench system $\mathtt{W}$ of a parallel kinematic chain composed of m serial chains are given by:

$$\mathtt{T} = \bigcap_{i=1}^{m} \mathtt{T}^i \quad , \quad \mathtt{W} = \bigoplus_{i=1}^{m} \mathtt{W}^i \tag{2}$$

The twist system $\mathtt{T}$ and the wrench system $\mathtt{W}$ of a given kinematic chain are reciprocal to each other: $\mathtt{T} = \mathtt{W}^{\perp}$ and $\mathtt{W} = \mathtt{T}^{\perp}$, the operator $\perp$ denoting the reciprocity of two screw systems. If t denotes the twist system order, then the wrench system order is $w = 6 - t$. Any twist in $\mathtt{T}$ is reciprocal to any wrench in $\mathtt{W}$ and vice versa [15]. Two screws are reciprocal to each other if their orthogonal product is equal to zero. Two zero pitch screws are reciprocal to each other if and only if their axes are coplanar. A zero pitch screw is reciprocal to an infinite pitch screw if their directions are orthogonal to each other. Two infinite pitch screws are always reciprocal to each other.

4 The *H*4 Constraint Analysis

The $H4$ robot shown in Fig. 1 belongs to a new family of 4-DOF parallel robots designed for high-speed pick and place operations [17, 18]. A kinematic graph of the $H4$ robot was given in [19]. The $H4$ robot is composed of four identical legs $l_i = \underline{\mathsf{R}}^i\text{–}(4\mathsf{S})^i$, ($i = 1, \ldots, 4$), attached to a common base (B) and linked to the end effector (E) by means of an articulated nacelle. The nacelle is composed of three bodies: (i) b_I, connecting l_1 and l_2 in parallel and giving a resulting chain l_{12}; (ii) b_{II} connecting l_3 and l_4 in parallel and giving a resulting chain l_{34}; and (iii) (E), linked to b_I and b_{II} with two revolute joints R_I and R_{II} respectively, and carrying the end effector of the robot. Finally, the $H4$ robot has two compound legs: $L_I = l_{12}\text{–}\mathsf{R}_I$ and $L_{II} = l_{34}\text{–}\mathsf{R}_{II}$.

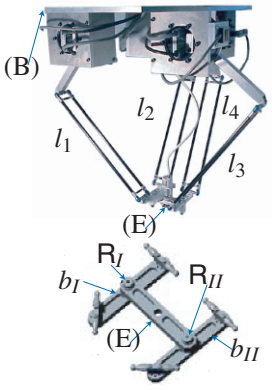


Fig. 1 Prototype of the $H4$ robot.

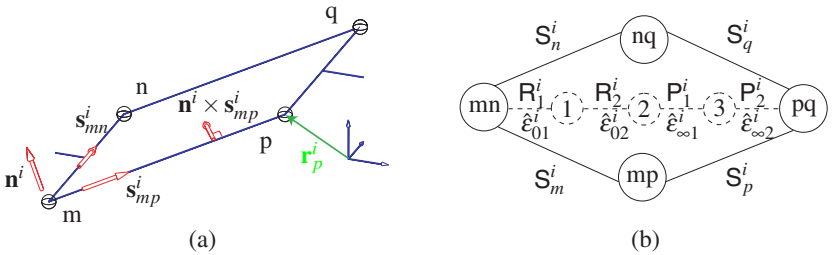


Fig. 2 (a) A $(4\mathsf{S})^i$ linkage; (b) Kinematic graph and twist graph of the $(4\mathsf{S})^i$ linkage.

4.1 Twist System and Wrench System of the 4S Parallel Linkage

A parallel linkage $(4\mathsf{S})^i$ of ith l_i is represented in Fig. 2a and its kinematic graph is shown in Fig. 2b. In this graph, a segment denotes a kinematic joint and a circle denotes a link. The $(4\mathsf{S})^i$ linkage is a parallel kinematic chain composed of two serial chains S^i_m–S^i_p and S^i_n–S^i_q, where S^i_m, S^i_n, S^i_p and S^i_q are four spherical joints centered at points m, n, p and q respectively. In a general configuration, a 4S linkage is not a parallelogram. However, the $H4$ robot is designed in such a way that the $(4\mathsf{S})^i$ is a parallelogram in all configurations [17], i.e., mn // pq and mp // nq. Note that for the ith leg l_i, $\mathbf{r}^i_p$ denotes the position vector of point p, $\mathbf{s}^i_{mp}$ denotes a unit vector directed along line mp and $\mathbf{n}^i$ denotes a unit vector normal to the plane of the parallelogram linkage.

The wrench system of S^i_m–S^i_p includes all τ_0 passing through m and p. It is a 1-system: $\mathtt{W}_{mp} = span(\hat{\tau}_{01})$ where[1] $\hat{\tau}_{01} = (\mathbf{s}^i_{mp}, \mathbf{r}^i_p \times \mathbf{s}^i_{mp})^T$. Similarly, the wrench system of S^i_n–S^i_q is: $\mathtt{W}_{nq} = span(\hat{\tau}_{02})$ where $\hat{\tau}_{02} = (\mathbf{s}^i_{nq}, \mathbf{r}^i_q \times \mathbf{s}^i_{nq})^T$. From Eq. (2), the wrench system of the $(4\mathsf{S})^i$ linkage is $\mathtt{W}_{4\mathsf{S}} = \mathtt{W}_{mp} \oplus \mathtt{W}_{nq} = span(\hat{\tau}_{01}, \hat{\tau}_{02})$. Since $\mathbf{s}^i_{mp}$ and $\mathbf{s}^i_{nq}$ are parallel, $\mathtt{W}_{4\mathsf{S}} = span(\hat{\tau}_{01}, \hat{\tau}_{\infty 1})$, where[2] $\hat{\tau}_{\infty 1} = \hat{\tau}_{02} - \hat{\tau}_{01} = (\mathbf{0}, \mathbf{s}^i_{mp} \times \mathbf{s}^i_{mn})^T$ [16]. $\mathtt{W}_{4\mathsf{S}}$ is a 2-system, its twist system $\mathtt{T}_{4\mathsf{S}} = \mathtt{W}^{\perp}_{4\mathsf{S}}$ is a 4-system spanned by: $\hat{\varepsilon}_{01} = (\mathbf{s}^i_{mp}, \mathbf{r}^i_p \times \mathbf{s}^i_{mp})^T$, $\hat{\varepsilon}_{02} = (\mathbf{s}^i_{mn}, \mathbf{r}^i_n \times \mathbf{s}^i_{mn})^T$, $\hat{\varepsilon}_{\infty 1} = (\mathbf{0}, \mathbf{n}^i)^T$ and $\hat{\varepsilon}_{\infty 2} = (\mathbf{0}, \mathbf{s}^i_{mp} \times \mathbf{n}^i)^T$.

The $(4\mathsf{S})^i$ linkage provides two independent translations and two independent rotations represented with the four independent twists $\hat{\varepsilon}_{01}$, $\hat{\varepsilon}_{02}$, $\hat{\varepsilon}_{\infty 1}$ and $\hat{\varepsilon}_{\infty 2}$ that span its twist system. Its twist graph represents the joints associated with the twists of a basis of its twist system. It is a virtual serial kinematic chain (Fig. 2b) composed of two virtual revolute joints R^i_1 and R^i_2 associated with $\hat{\varepsilon}_{01}$ and $\hat{\varepsilon}_{02}$ and two virtual prismatic joints P^i_1 and P^i_2 associated with $\hat{\varepsilon}_{\infty 1}$ and $\hat{\varepsilon}_{\infty 2}$. These virtual joints are represented with dashed lines. Links 1, 2 and 3 are virtual links represented with dashed circles.

[1] $\hat{\tau}$ and $\hat{\varepsilon}$ stand for unit wrench and unit twist, respectively.

[2] By applying a linear transformation.

4.2 Constraint Wrenches of the H4 Robot

Since the twist system of a $(4\mathsf{S})^i$ linkage is represented with a virtual chain composed of two revolute joints and two prismatic joints, a twist graph of the $H4$ robot is obtained as shown in Fig. 3. The twist system $\mathtt{T}_i$ of the ith leg $l_i = \underline{\mathsf{R}}^i$–$(4\mathsf{S})^i$ (Fig. 4a) is equivalent to the twist system of a serial chain $\underline{\mathsf{R}}^i$–R^i_1–R^i_2–P^i_1–P^i_2 spanned by: $\hat{\varepsilon}^i_{01} = (\mathbf{s}^i_1, \mathbf{r}^i_1 \times \mathbf{s}^i_1)^T$, $\hat{\varepsilon}^i_{02} = (\mathbf{s}^i_{mn}, \mathbf{r}^i_n \times \mathbf{s}^i_{mn})^T$, $\hat{\varepsilon}^i_{03} = (\mathbf{s}^i_{mp}, \mathbf{r}^i_p \times \mathbf{s}^i_{mp})^T$, $\hat{\varepsilon}^i_{\infty 1} = (\mathbf{0}, \mathbf{n}^i)^T$ and $\hat{\varepsilon}^i_{\infty 2} = (\mathbf{0}, \mathbf{s}^i_{mp} \times \mathbf{n}^i)^T$. Note that $\mathbf{s}^i_1$ denotes a unit vector along the direction of $\underline{\mathsf{R}}^i$ joint axis. For the ith leg l_i, $\mathbf{s}^i_1 \,//\, \mathbf{s}^i_{mn} \,//\, \mathbf{s}^i_{pq}$ [17]. The constraint wrench system $\mathtt{W}_i$ of l_i includes the wrenches that are reciprocal to all the twists in $\mathtt{T}_i$. Thus, the axis of a τ_0 in $\mathtt{W}_i$ is coplanar to the axes of $\hat{\varepsilon}^i_{01}$, $\hat{\varepsilon}^i_{02}$ and $\hat{\varepsilon}^i_{03}$ and orthogonal to the directions of $\hat{\varepsilon}^i_{\infty 1}$ and $\hat{\varepsilon}^i_{\infty 2}$. Therefore, $\mathtt{W}_i$ does not contain any τ_0. The direction of a τ_∞ in $\mathtt{W}_i$ is orthogonal to $\mathbf{s}^i_{mn}, \mathbf{s}^i_{mp}$ and $\mathbf{s}^i_1$, the corresponding direction is $\mathbf{s}^i_{mn} \times \mathbf{s}^i_{mp} = \mathbf{n}^i$. Hence, $\mathtt{W}_i$ is a 1-system spanned by $\hat{\tau}^i_\infty = (\mathbf{0}, \mathbf{n}^i)^T$.

The chain l_{12} being the connection of l_1 and l_2 in parallel, its wrench system $\mathtt{W}_{12} = \mathtt{W}_1 \oplus \mathtt{W}_2 = span(\hat{\tau}^1_\infty, \hat{\tau}^2_\infty)$. Thus, l_{12} provides translations in any direction and cannot rotate about an axis of direction $\mathbf{n}^1$ or $\mathbf{n}^2$. The direction of possible rotation is independent of these two directions. This rotation is represented with the twist $\hat{\varepsilon}_{012}$ whose axis is directed along $\mathbf{s}^{12}_m = \mathbf{n}^1 \times \mathbf{n}^2$. Therefore, $\mathtt{T}_{12} = \mathtt{W}^\perp_{12} = span(\hat{\varepsilon}_{\infty x}, \hat{\varepsilon}_{\infty y}, \hat{\varepsilon}_{\infty z}, \hat{\varepsilon}_{012})$ where $\hat{\varepsilon}_{\infty x}$, $\hat{\varepsilon}_{\infty y}$ and $\hat{\varepsilon}_{\infty z}$ are the infinite pitch twists associated with translations along directions $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$, respectively. Similarly, the twist system of leg l_{34} is $\mathtt{T}_{34} = \mathtt{W}^\perp_{34} = span(\hat{\varepsilon}_{\infty x}, \hat{\varepsilon}_{\infty y}, \hat{\varepsilon}_{\infty z}, \hat{\varepsilon}_{034})$ and the axis of $\hat{\varepsilon}_{034}$ is directed along $\mathbf{s}^{34}_m = \mathbf{n}^3 \times \mathbf{n}^4$.

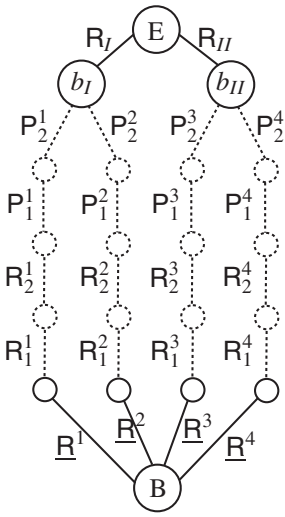


Fig. 3 Twist graph of the $H4$ robot.

The twist system of leg $L_I = l_{12}$–R_I is $\mathtt{T}_I = \mathtt{T}_{12} \oplus \mathtt{T}_{R_I}$. R_I is the rotation about axis Z_I (Fig. 4c) represented with the twist $\hat{\varepsilon}_{0Z_I} = (\mathbf{z}, \mathbf{r}_c \times \mathbf{z})^T$. Thus, $\mathtt{T}_I = span(\hat{\varepsilon}_{\infty x}, \hat{\varepsilon}_{\infty y}, \hat{\varepsilon}_{\infty z}, \hat{\varepsilon}_{012}, \hat{\varepsilon}_{0Z_I})$. Therefore, $\mathtt{W}_I = \mathtt{T}^\perp_I = span(\hat{\tau}_{\infty I})$ where $\hat{\tau}_{\infty I} = (\mathbf{0}, \mathbf{z} \times \mathbf{s}^{12}_m)^T$. Likewise, $\mathtt{W}_{II} = span(\hat{\tau}_{\infty II})$ where $\hat{\tau}_{\infty II} = (\mathbf{0}, \mathbf{z} \times \mathbf{s}^{34}_m)^T$. Legs L_I and L_{II} are mounted in parallel on the end effector of the $H4$ robot. Thus, its constraint wrench system is:

$$\mathtt{W}^c_{H4} = \mathtt{W}_I \oplus \mathtt{W}_{II} = span(\hat{\tau}_{\infty I}, \hat{\tau}_{\infty II}) \tag{3}$$

The end effector of the $H4$ robot is constrained by two pure moments: $\hat{\tau}_{\infty I} = (\mathbf{0}, \mathbf{z} \times \mathbf{s}^{12}_m)^T$ and $\hat{\tau}_{\infty II} = (\mathbf{0}, \mathbf{z} \times \mathbf{s}^{34}_m)^T$. It provides a Schönflies motion, i.e., three independent translations and one rotation about an axis of fixed direction $\mathbf{z}$.

4.3 Actuation Wrenches and Global Wrench System of the H4 Robot

The actuated joints of the $H4$ robot are the $\underline{\mathsf{R}}^i$ joints of legs l_i, $(i = 1, \ldots, 4)$. Legs l_1 and l_2 (respectively l_3 and l_4) are connected with R_I (respectively R_{II}). The actuation wrench system $\mathtt{W}_1^a$ of l_1 includes wrenches that are reciprocal to the twist of the R_I joint and all the twists in $\mathtt{T}_1$, except for the twist $\hat{\mathcal{E}}_{01}^1$ of $\underline{\mathsf{R}}^1$. Thus, $\mathtt{W}_1^a$ does not include any infinite pitch wrench. It is a 1-system spanned by $\hat{\tau}_0^1$ of axis parallel to $\mathbf{s}_{mp}^1$, intersecting axis Z_I and lying on the plane spanned by mn and mp. The actuation wrench of l_1 is, thus, $\hat{\tau}_0^1 = (\mathbf{s}_{uv}^1, \mathbf{r}_v^1 \times \mathbf{s}_{uv}^1)^T$ with $\mathbf{s}_{uv}^1 \mathbin{/\!/} \mathbf{s}_{mp}^1$. Similarly, the actuation wrenches of legs l_2, l_3 and l_4 are $\hat{\tau}_0^i = (\mathbf{s}_{uv}^i, \mathbf{r}_v^i \times \mathbf{s}_{uv}^i)^T$, $(i = 2, 3, 4)$ with $\mathbf{s}_{uv}^i \mathbin{/\!/} \mathbf{s}_{mp}^i$.

In a non-singular configuration, the four actuation wrenches $\hat{\tau}_0^i$ are independent and span the actuation wrench system of the $H4$ robot: $\mathtt{W}_{H4}^a = span(\hat{\tau}_0^i)$, $(i = 1, \ldots, 4)$. The global wrench system of the robot is:

$$\mathtt{W}_{H4} = \mathtt{W}_{H4}^c \oplus \mathtt{W}_{H4}^a = span(\hat{\tau}_{\infty I}, \hat{\tau}_{\infty II}, \hat{\tau}_0^1, \hat{\tau}_0^2, \hat{\tau}_0^3, \hat{\tau}_0^4) \tag{4}$$

The legs of the $H4$ robot apply two constraint moments and four actuation forces to its end effector. Its global wrench system is a 6-system. A parallel singularity occurs when the wrenches in the 6-system become linearly dependent and span a k-system with $k < 6$.

5 Singularity Analysis of the $H4$ Robot

5.1 Wrench Diagram of the H4 Robot in $\mathbb{P}_3$

A basis of the global wrench system $\mathtt{W}_{H4}$ is composed of two constraint moments $\hat{\tau}_{\infty I}$ and $\hat{\tau}_{\infty II}$, and four actuation forces $\hat{\tau}_0^i = (\mathbf{s}_{uv}^i, \mathbf{r}_v^i \times \mathbf{s}_{uv}^i)^T$, $(i = 1, \ldots, 4)$. Those wrenches are represented by two infinite lines and four finite lines in $\mathbb{P}_3$. To obtain the six extensors of the $H4$ superbracket, we have to select twelve projective points on the six projective lines, i.e., two points on each one. The extensor of an infinite line is represented by two distinct infinite points. The extensor of a finite line can be represented by either two distinct finite points or one finite point and one infinite point, since any finite line has one point at infinity corresponding to its direction.

We know that $\hat{\tau}_0^1$ and $\hat{\tau}_0^2$ intersect axis Z_I. Likewise, $\hat{\tau}_0^3$ and $\hat{\tau}_0^4$ intersect axis Z_{II} (Fig. 4). Let $\mathtt{a}$ (respectively $\mathtt{c}$) denote the intersection point of $\hat{\tau}_0^1$ (respectively $\hat{\tau}_0^2$) and Z_I and let $\mathtt{e}$ (respectively $\mathtt{g}$) denote the intersection point of $\hat{\tau}_0^3$ (respectively $\hat{\tau}_0^4$) and Z_{II}. Besides, Z_I and Z_{II} are parallel, i.e., $\mathtt{ac}$ and $\mathtt{eg}$ are parallel lines. They intersect in the infinite plane Π_∞ at point $\underline{\mathtt{j}} = (\mathbf{z}, 0)^T$, which corresponds to the $\mathbf{z}$ direction. Note that an underlined letter stands for an infinite point. The finite line representing $\hat{\tau}_0^1 = (\mathbf{s}_{uv}^1, \mathbf{r}_v^1 \times \mathbf{s}_{uv}^1)^T$ can be defined by any two points on this line.

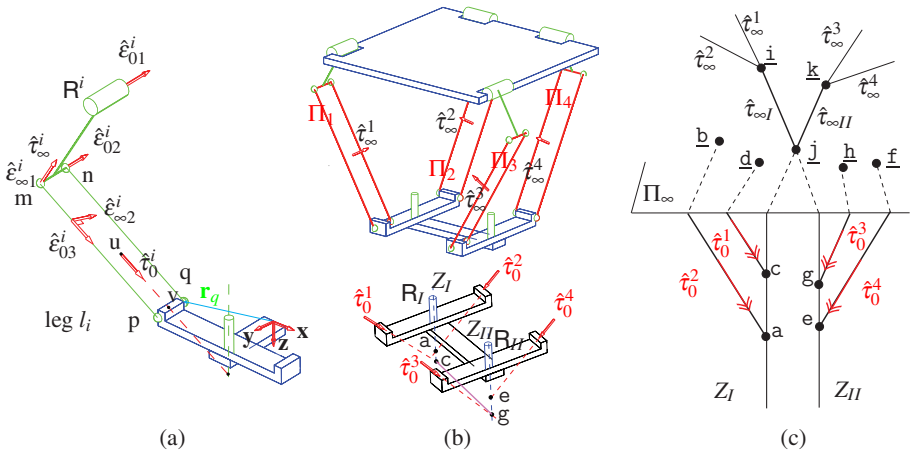


Fig. 4 (a) Twists of leg l_i; (b) Constraint and actuation wrenches; (c) Wrench diagram in $\mathbb{P}_3$ of the $H4$ robot.

Therefore, let $\hat{\tau}_0^1 = \mathtt{a}\underline{\mathtt{b}}$ with $\underline{\mathtt{b}} = (\mathbf{s}_{uv}^1, 0)^T$ the infinite point of $\hat{\tau}_0^1$ representing its direction. Likewise, $\hat{\tau}_0^2 = \mathtt{c}\underline{\mathtt{d}}$ with $\underline{\mathtt{d}} = (\mathbf{s}_{uv}^2, 0)^T$, $\hat{\tau}_0^3 = \mathtt{e}\underline{\mathtt{f}}$ with $\underline{\mathtt{f}} = (\mathbf{s}_{uv}^3, 0)^T$ and $\hat{\tau}_0^4 = \mathtt{g}\underline{\mathtt{h}}$ with $\underline{\mathtt{h}} = (\mathbf{s}_{uv}^4, 0)^T$.

The infinite line $\hat{\tau}_\infty^i = (\mathbf{0}, \mathbf{n}^i)^T$, $(i = 1, \ldots, 4)$ is the projection on Π_∞ of a finite plane Π_i of normal $\mathbf{n}^i$. The intersection line of planes Π_1 and Π_2 is a finite line orthogonal to $\mathbf{n}^1$ and $\mathbf{n}^2$. The infinite point of this line is expressed as: $\underline{\mathtt{i}} = (\mathbf{n}^1 \times \mathbf{n}^2, 0)^T = (\mathbf{s}_m^{12}, 0)^T$ and corresponds to the intersection point of $\hat{\tau}_\infty^1$ and $\hat{\tau}_\infty^2$. In the same vein, the intersection point of $\hat{\tau}_\infty^3$ and $\hat{\tau}_\infty^4$ is expressed as: $\underline{\mathtt{k}} = (\mathbf{n}^3 \times \mathbf{n}^4, 0)^T = (\mathbf{s}_m^{34}, 0)^T$.

Let us consider the constraint moment $\hat{\tau}_{\infty I} = (\mathbf{0}, \mathbf{z} \times \mathbf{s}_m^{12})^T$. The vector $\mathbf{z} \times \mathbf{s}_m^{12}$ is normal to any finite plane spanned by the two vectors $\mathbf{z}$ and $\mathbf{s}_m^{12}$. The infinite point of $\mathbf{s}_m^{12}$ is $\underline{\mathtt{i}}$ and the infinite point of $\mathbf{z}$ is $\underline{\mathtt{j}}$. Therefore, $\hat{\tau}_{\infty I} = \underline{\mathtt{ij}}$. Likewise, $\hat{\tau}_{\infty II} = \underline{\mathtt{kj}}$. We have selected the twelve points of the $H4$ superbracket. The wrench diagram of the $H4$ robot is represented in Fig. 4c.

5.2 *Superbracket of the H4 Robot*

The rows of the inverse Jacobian matrix of a parallel manipulator are Plücker coordinates of six lines in $\mathbb{P}_3$. The superjoin of these six vectors in $\mathbb{P}_5$ corresponds to the determinant of their six Plücker coordinate vectors up to a scalar multiple, which is the superbracket in GCA $\Lambda(V^{(2)})$ [8]. Thus, a singularity occurs when these six Plücker coordinate vectors are dependent, which is equivalent to a superbracket equal to zero.

In [20], the theory of projective invariants has been used to decompose the superbracket into an expression having brackets involving 12 points selected on the axes

of these lines. The expression of the $H4$ superbracket is $[\underline{\mathtt{a}}\underline{\mathtt{b}}\,\mathtt{c}\underline{\mathtt{d}}\,\mathtt{e}\underline{\mathtt{f}}\,\underline{\mathtt{g}}\underline{\mathtt{h}}\,\underline{\mathtt{i}}\,\underline{\mathtt{j}}\,\underline{\mathtt{k}}\,\underline{\mathtt{j}}]$, it corresponds to twelve points selected in the robot wrench diagram. This expression can be developed into a linear combination (*sum*) of 24 bracket monomials [2, 20], each one being the product of three brackets of four projective points. The bracket of any four coplanar projective points is equal to zero. Hence, 19 amongst the 24 monomials are null. As detailed in [21], the simplified expression of this superbracket is:

$$\begin{aligned}[\underline{\mathtt{a}}\underline{\mathtt{b}}\,\mathtt{c}\underline{\mathtt{d}}\,\mathtt{e}\underline{\mathtt{f}}\,\underline{\mathtt{g}}\underline{\mathtt{h}}\,\underline{\mathtt{i}}\,\underline{\mathtt{j}}\,\underline{\mathtt{k}}\,\underline{\mathtt{j}}] &= [\underline{\mathtt{g}}\,\underline{\mathtt{i}}\,\underline{\mathtt{k}}\,\underline{\mathtt{j}}]\left([\underline{\mathtt{a}}\,\underline{\mathtt{b}}\,\underline{\mathtt{d}}\,\underline{\mathtt{f}}][\mathtt{c}\,\underline{\mathtt{g}}\,\underline{\mathtt{h}}\,\underline{\mathtt{j}}] - [\underline{\mathtt{a}}\,\underline{\mathtt{b}}\,\underline{\mathtt{d}}\,\underline{\mathtt{h}}][\mathtt{c}\,\underline{\mathtt{g}}\,\underline{\mathtt{f}}\,\underline{\mathtt{j}}]\right) \\ &= [\underline{\mathtt{g}}\,\underline{\mathtt{i}}\,\underline{\mathtt{k}}\,\underline{\mathtt{j}}][\underline{\mathtt{a}}\,\underline{\mathtt{b}}\,\underline{\mathtt{d}}\,\dot{\underline{\mathtt{f}}}][\mathtt{c}\,\underline{\mathtt{g}}\,\dot{\underline{\mathtt{h}}}\,\underline{\mathtt{j}}] \end{aligned} \tag{5}$$

where the dotted letters denote the permuted elements as mentioned in [8].

5.3 Geometric Conditions for the H4 Robot Singularities

From Eq. (5), a parallel singularity occurs when:

1. $[\underline{\mathtt{g}}\,\underline{\mathtt{i}}\,\underline{\mathtt{k}}\,\underline{\mathtt{j}}] = 0 \Leftrightarrow [\underline{\mathtt{i}}\,\underline{\mathtt{k}}\,\underline{\mathtt{j}}] = 0 \Leftrightarrow \underline{\mathtt{i}}$, $\underline{\mathtt{j}}$ and $\underline{\mathtt{k}}$ belong to the same projective line. This condition is expressed in vector form as follows:

$$(\mathbf{s}_m^{12} \times \mathbf{s}_m^{34}) \bullet \mathbf{z} = 0 \tag{6}$$

where $\mathbf{s}_m^{12} = \mathbf{n}^1 \times \mathbf{n}^2$ and $\mathbf{s}_m^{34} = \mathbf{n}^3 \times \mathbf{n}^4$. This condition corresponds to the constraint singularities of the robot that occur when the legs lose their ability to constrain the motion of the end effector, which gains one or several DOFs. By solving Eq. (6), we obtain the different cases for this type of singularity as follows:

   (a) $\mathbf{s}_m^{12} \times \mathbf{s}_m^{34} = 0$: $\mathbf{s}_m^{12}$ and $\mathbf{s}_m^{34}$ are parallel, which happens when the intersection line of two amongst the four planes Π_i, $(i = 1, \ldots 4)$, is parallel to the intersection line of the two other planes. For example, when $\mathbf{n}^1 \,//\, \mathbf{n}^3$ and $\mathbf{n}^2 \,//\, \mathbf{n}^4$ as shown in Fig. 5a.
   (b) $\mathbf{s}_m^{12} \,//\, \mathbf{z}$, i.e., when the two planes Π_1 and Π_2 are vertical, their normal vectors $\mathbf{n}^1$ and $\mathbf{n}^2$ are in the horizontal plane and $\mathbf{s}_m^{12} = \mathbf{n}^1 \times \mathbf{n}^2$ is parallel to $\mathbf{z}$. A similar case happens when $\mathbf{s}_m^{34} \,//\, \mathbf{z}$;
   (c) $\mathbf{n}^1 \,//\, \mathbf{n}^2$, i.e., the two planes Π_1 and Π_2 are parallel, and as a consequence $\mathbf{s}_m^{12} = 0$. A similar case happens when $\mathbf{n}^3 \,//\, \mathbf{n}^4$;
   (d) $\mathbf{s}_m^{12}$, $\mathbf{s}_m^{34}$ and $\mathbf{z}$ are coplanar but not parallel to each other.
   (e) $\mathbf{s}_m^{12} \,//\, \mathbf{s}_m^{34} \,//\, \mathbf{z}$, i.e., when the four planes Π_i, $(i = 1, \ldots 4)$, are vertical as shown in Fig. 5b.

   In cases (a), (b), (c) and (d), the two constraint moments $\hat{\tau}_\infty^I$ and $\hat{\tau}_\infty^{II}$ are identical $(\mathbf{z} \times \mathbf{s}_m^{12} \,//\, \mathbf{z} \times \mathbf{s}_m^{34})$ or one of these moments is null. The constraint wrench system of the robot becomes a 1-system and its twist system a 5-system, and as a result,

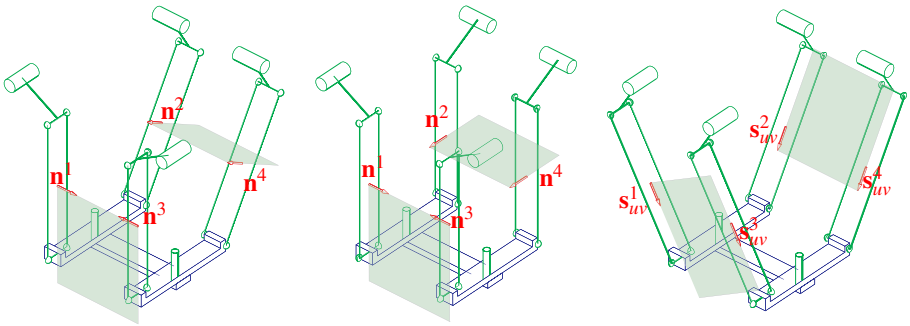


Fig. 5 Some singular configurations of the $H4$ robot.

the manipulator gains one DOF. In case (e), the two constraints moments $\hat{\tau}^I_\infty$ and $\hat{\tau}^{II}_\infty$ are null and the robot gains two DOFs.

2. $[\mathtt{a}\underline{\mathtt{bd}}\underline{\mathtt{f}}][\mathtt{c}\mathtt{g}\underline{\mathtt{h}}\underline{\mathtt{j}}] = 0 \Leftrightarrow \left(\mathtt{a}\underline{\mathtt{bd}} \wedge \mathtt{c}\mathtt{g}\underline{\mathtt{j}}\right) \wedge \underline{\mathtt{fh}} = 0 \Leftrightarrow$ the projective line $\underline{\mathtt{fh}}$ intersects with the intersection line of planes $\mathtt{a}\underline{\mathtt{bd}}$ and $\mathtt{c}\mathtt{g}\underline{\mathtt{j}}$. This condition is expressed in vector form as follows:

$$\left([\mathbf{s}^1_{uv} \times \mathbf{s}^2_{uv}] \times [(\mathbf{r}_g - \mathbf{r}_c) \times \mathbf{z}]\right) \bullet (\mathbf{s}^3_{uv} \times \mathbf{s}^4_{uv}) = 0 \tag{7}$$

This condition occurs when the legs cannot control the linear velocity of the end effector. The different cases for this condition can be established by solving Eq. (7). For example, when $\mathbf{s}^1_{uv} /\!/ \mathbf{s}^2_{uv}$ and $\mathbf{s}^3_{uv} /\!/ \mathbf{s}^4_{uv}$ (Fig. 5c).

6 Conclusions

In this paper, a general methodology was proposed to analyze the singularities of parallel manipulators with an articulated nacelle. The methodology consists of two main steps. First, the new concept of twist graph is used to simplify the constraint analysis. This graph is obtained with the theory of reciprocal screws. Then, a wrench diagram is obtained in order to derive a simplified expression of the superbracket decomposition. This expression is analyzed to provide geometric conditions for singular configurations. The methodology was explained through the singularity analysis of the $H4$ robot. It can be also applied to analyze the singularities of other complex robots such as the Par4[3] [22].

[3] The commercial name of the Par4 is Quattro (http://www.adept.com/).

References

1. J.P. Merlet. Singular configurations of parallel manipulators and Grassmann geometry. *The International Journal of Robotics Research*, **8**, 45–56 (1989).
2. P. Ben-Horin and P. Shoham. Singularity condition of six-degree-of-freedom three-legged parallel robots based on Grassmann–Cayley Algebra. *IEEE Transactions on Robotics*, **22**, 577–590 (2006).
3. P. Ben-Horin and P. Shoham. Application of Grassmann–Cayley Algebra to geometrical interpretation of parallel robot singularities. *International Journal of Robotics Research*, **1**, 127–141 (2009).
4. S.A. Joshi and L.W. Tsai. Jacobian analysis of limited-DOF parallel manipulators. *Transactions of the ASME*, **124** (2002).
5. D. Kanaan, P. Wenger, S. Caro, and D. Chablat. Singularity analysis of lower-mobility parallel manipulators using Grassmann–Cayley Algebra. *IEEE Transactions on Robotics*, **25**, 995–1004 (2009).
6. H.B. Choi, A. Konno, and M. Uchiyama. Singularity analysis of a novel 4-DOFs parallel robot H4 by using screw theory. In *Proceedings of DETC'03 ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Chicago, Illinois, USA, September 2–6 (2003).
7. J. Wu, Z. Yin, and Y. Xiong. Singularity analysis of a novel 4-DOF parallel manipulator H4. *The International Journal of Advanced Manufacturing Technology*, **29**, 794–802 (2006).
8. N.L. White. Grassmann–Cayley Algebra and robotics applications. In *Handbook of Geometric Computing*, Part VIII, pp. 629–656 (2005).
9. R.S. Ball. *A Treatise on the Theory of Screws*. Cambridge University Press, Cambridge (1900).
10. K.J. Waldron. The constraint analysis of mechanisms. *Journal of Mechanisms*, **1**, 101–114 (1966).
11. K.H. Hunt. *Kinematic Geometry of Mechanisms*. Clarendon Press, Oxford (1978).
12. K.H. Hunt. Structural kinematics of in. parallel actuated robot arms. *ASME J. of Mech. Transm. and Automn. in Design*, **105**, 705–712 (1983).
13. M. Mohamed and J. Duffy. A direct determination of the instantaneous kinematics of fully parallel robot manipulators. *ASME J. of Mechanisms, Transmissions and Automation in Design*, **107**(2), 226–229 (1985).
14. D. Zlatanov, I. Bonev, and C. Gosselin. Constraint singularities of parallel mechanisms. In *IEEE ICRA'02*, Washington DC, Vol. 1, pp. 496–502 (2002).
15. W. Kong and C.M. Gosselin. *Type Synthesis of Parallel Mechanisms*. Springer (2007).
16. T.S. Zhao, J.S. Dai, and Z. Huang. Geometric analysis of overconstrainted parallel manipulators with three and four degrees of freedom. *JSME International Journal*, **4**(3), 730–740 (2002).
17. F. Pierrot and O. Company. H4: A new family of 4-DOF parallel robots. In *Proc. of the IEEE / ASME Int. Conf. on Advanced Intelligent Mechatronics*, Atlanta, Georgia, USA, September, pp. 508–513 (1999).
18. F. Pierrot, F. Marquet, O Company, and T. Gil. H4 parallel robot: Modeling, design and preliminary experiments. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3256–3261 (2001).
19. G. Gogu. *Structural Synthesis of Parallel Robots*, Part 1: methodology edition, Solid Mechanics and Its Applications, Vol. 149, Springer (2008).
20. T. McMillan and N.L. White The dotted straightening algorithm *Journal of Symbolic Computation*, **11**, 471–482 (1991).
21. S. Amine, D. Kanaan, S. Caro, and P. Wenger. Singularity analysis of the H4 robot using Grassmann–Cayley Algebra. Internal Report IRCCyN 2010, http://www.irccyn.ec-nantes.fr/∼caro/Publications/ARK2010/IntReport.pdf.
22. V. Nabat, M. de la Rodriguez, O. Company, S. Krut, and F. Pierrot. Par4: Very high speed parallel robot for pick-and-place. In *Proceedings Intelligent Robots and Systems, IEEE/RSJ International Conference*, August, pp. 553–558 (2005).

Human Motion Reconstruction and Synthesis of Human Skills

Emel Demircan[1], Thor Besier[2], Samir Menon[1] and Oussama Khatib[1]

[1]*Artificial Intelligence Lab.,* [2]*Human Performance Lab.,*
Stanford University, Stanford, CA 94305, U.S.A.
e-mail: {emeld, besier, smenon, khatib}@stanford.edu

Abstract. Reconstructing human motion dynamics in real-time is a challenging problem since it requires accurate motion sensing, subject specific models, and efficient reconstruction algorithms. A promising approach is to construct accurate human models, and control them to behave the same way the subject does. Here, we demonstrate that the whole-body control approach can efficiently reconstruct a subject's motion dynamics in real world task-space when given a scaled model and marker based motion capture data. We scaled a biomechanically realistic musculoskeletal model to a subject, captured motion with suitably placed markers, and used an operational space controller to directly track the motion of the markers with the model. Our controller tracked the positions, velocities, and accelerations of many markers in parallel by assigning them to tasks with different priority levels based on how free their parent limbs were. We executed lower priority marker tracking tasks in the successive null spaces of the higher priority tasks to resolve their interdependencies. The controller accurately reproduced the subject's full body dynamics while executing a throwing motion in near real time. Its reconstruction closely matched the marker data, and its performance was consistent for the entire motion. Our findings suggest that the direct marker tracking approach is an attractive tool to reconstruct and synthesize the dynamic motion of humans and other complex articulated body systems in a computationally efficient manner.

Key words: Motion reconstruction, marker space control, musculoskeletal model, human motion synthesis.

1 Introduction

The reconstruction of human motions is important to researchers who wish to understand the motor strategies skilled humans employ, predict clinical treatment outcomes, or synthesize actor movement in virtual environments. Understanding human motor control involves studying the principles used to optimize movement, and its improvement requires finding changes which make it more optimal. Predicting clinical outcomes of specific biomechanics operations requires developing detailed subject customized models to predict changes in the dynamics as parameters vary. Finally, synthesizing motion involves mapping a subject's motion to a model, over-

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_30,

coming differences in scale, and possibly modifying and mixing motions as they are executed. Motion reconstruction's many objectives make it a challenging task.

Existing reconstruction techniques are application specific, use approximate human models, and usually focus on animating virtual characters to execute tracked motions. The inverse approach estimates the subject's joint angles by imposing motion capture data's spatial and timing constraints to a model and obtains the kinematics or dynamics in joint space [1–7]. However, estimating joint space from the real world task space translates motion sensing errors into unnatural joint constraints which are further amplified by model imperfections. In addition, inverting motion capture data is computationally expensive. Task space reconstruction [8, 9] overcomes these difficulties by controlling a model to track task space motion capture data directly and obviates the inversion to joint space. Controlling realistic musculoskeletal models in task space to reconstruct motion is challenging due to the many degrees of freedom, their novel singular configurations, and computational efficiency constraints.

In this paper, we applied the task space reconstruction approach to track multiple markers with a detailed subject-customized biomechanical model. To ensure biomechanical detail, we developed a controllable musculoskeletal model based on existing biomechanical models of the upper and lower body [10, 11]. We identified a marker set that constrains the model sufficiently to make its motions match the subject's. We chose a task space marker control hierarchy which tracks markers on the root and leaves of the model's branching structure with the highest priority, and simultaneously tracks intermediate constrained markers in successive null spaces. The higher priority tasks track the motions of the end effectors and pelvis, and the lower priority tasks ensure that the motion of the remaining limbs is consistent with their marker trajectory constraints. Finally, we executed our reconstruction algorithms in our simulation and control framework [12] in near real time.

2 Musculoskeletal Motion Reconstruction

For a given desired task, all motion patterns such as body segment location and orientation, balance, posture, collision avoidance [13, 14] need to be specified and controlled in a logical order. They also need to be consistent with physio-mechanical constraints including joint range of motion, singularity avoidance, and muscular effort minimization [9]. To solve these problems, we extend the task-space control framework to marker space, where the marker trajectories are tracked with an accurate musculoskeletal model. The model is scaled to the human subject and is simulated in real-time. The reconstruction process starts with real-time data acquisition using optical marker-based motion capture, and motion data filtering. The musculoskeletal model is then scaled and used to directly track the marker trajectories to obtain the motion dynamics which may be analyzed later.

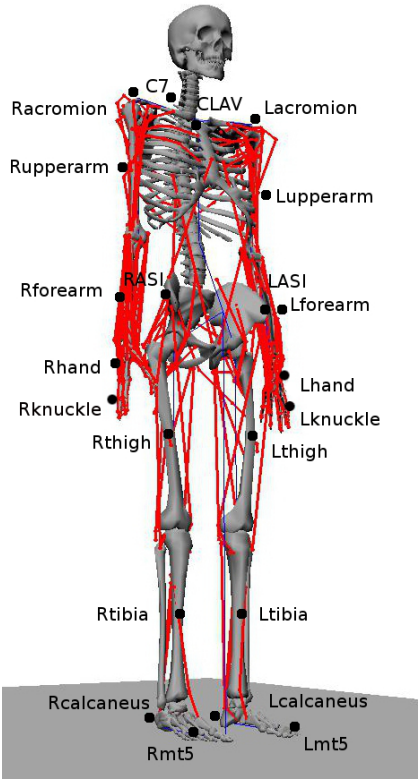


Fig. 1 The musculoskeletal model scaled and used as the basis for the human motion simulation and reconstruction in our task-level dynamic framework. The 22 markers tracked by our motion reconstruction controller are labeled on the model.

2.1 Experimental Procedure and Musculoskeletal Model

Experiments were conducted using an eight-camera Vicon motion capture system (OMG plc, Oxford UK). A 25-year old healthy left-handed female athlete performed maximum velocity (left-hand) throws of a tennis ball. The motion of the subject was captured at a rate of 120Hz. Following the experiment, the collected position data was processed in Vicon Nexus Software. The raw marker data were filtered using a 15Hz low pass 4th order Butterworth filter.

The musculoskeletal model used in this work combines existing upper [11] and lower [10] body models. The upper body's kinematics contain 15 degrees of freedom which represent the shoulder, elbow, forearm, wrist, and hand. The lower body's kinematics contain 17 degrees of freedom which represent the hip, knee, ankle, subtalar, and metatarsophalangeal joints. The arms-torso, torso-pelvis, and

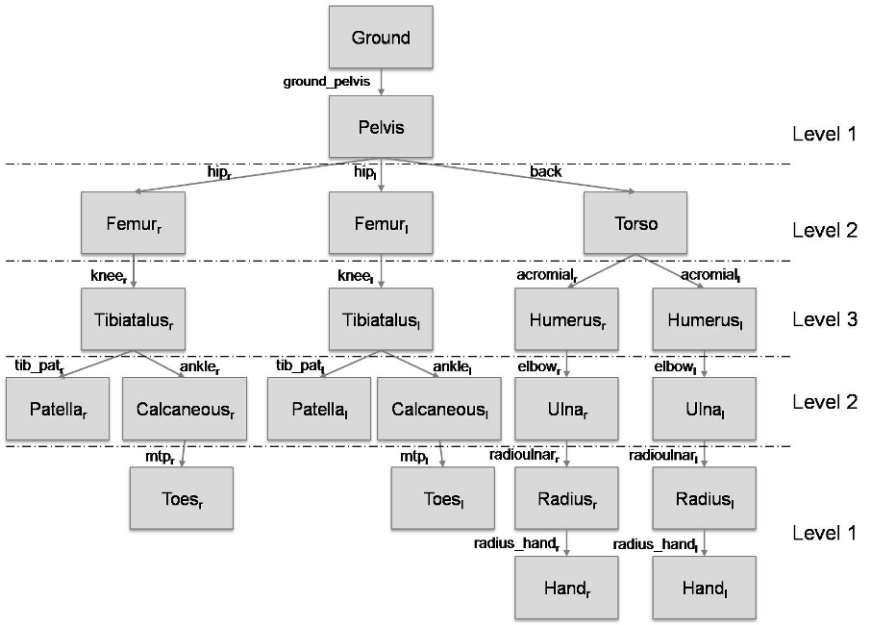


Fig. 2 The topology of the skeletal model. Each block represents a body segment which is connected to its parent body by the corresponding custom joint. The subscripts *r* and *l* denote the right and left body parts, respectively. The levels correspond to Table 1.

pelvis-leg joints are represented by ball-and-socket joints. The remaining joints are revolute.

The generic human model consists of 20 joints and has 32 degrees of freedom. It was scaled based on body segment mass-center locations [15] to match the anthropometry of the subject. Figure 1 illustrates the scaled musculoskeletal model used in our control and simulation framework and Fig. 2 shows the body segments of the model, each connected to its parent body via the corresponding custom joint. For example, the right ulna is connected to its parent body, right humerus via the right elbow custom joint.

2.2 Control Framework

2.2.1 Marker Space Control Formulation

The marker space formulation is constructed by applying an operational space controller [16] to track marker trajectories. The formulation begins with the joint space dynamics of the robot

$$A(q)\ddot{q} + b(q,\dot{q}) + g(q) = \Gamma \tag{1}$$

where q is the vector of n generalized coordinates of the articulated system, A is the $n \times n$ kinetic energy matrix, b is the vector of centrifugal and Coriolis generalized forces, g is the vector of gravity forces, and Γ is the vector of generalized control forces.

Task dynamic behavior is obtained by projecting (1) into the space associated with the task, which can be done with the following operation

$$\overline{J}_t^T [A\ddot{q} + b + g = \Gamma] \Longrightarrow \Lambda_t \ddot{x}_t + \mu_t + p_t = \overline{J}_t^T \Gamma \tag{2}$$

Here, $\overline{J}_t^T$ is the dynamically-consistent generalized inverse of J_t, the Jacobian of the task, Λ_t is the $m \times m$ kinetic energy matrix associated with the task, and μ_t and p_t are the associated centrifugal/Coriolis and gravity force vectors.

In the operational space framework, the task behavior is divided into a set of independent task points, and the torque component for the task is determined in a manner that compensates for the dynamics in task space. For a task behavior, x_t, with decoupled dynamics and unit inertial properties $\ddot{x}_t = F_t^*$, this torque is given by the force transformation

$$\Gamma_{\text{task}} = J_t^T F_t \tag{3}$$

where J_t is the Jacobian of the task and F_t is the operational space force. This operational space control is given by

$$F_t = \Lambda_t F_t^* + \mu_t + p_t \tag{4}$$

where F_t^* is the desired force associated with the task.

In the application to marker space, the task is defined in terms of the markers position coordinates describing the motion capture. The marker space control structure is established as

$$F_{m_i} = \Lambda_{m_i} F_{m_i}^* + \mu_{m_i} + p_{m_i} \tag{5}$$

Here, $F_{m_i}^*$ is the control force associated with i^{th} marker task, and is defined by

$$F_{m_i}^* = \ddot{x}_{m_{i_d}} - k_v(\dot{x}_{m_i} - \dot{x}_{m_{i_d}}) - k_p(x_{m_i} - x_{m_{i_d}}) \tag{6}$$

where $x_{m_{i_d}}$, $\dot{x}_{m_{i_d}}$, and $\ddot{x}_{m_{i_d}}$ denote the desired position, velocity, and acceleration, respectively, associated with the marker tracking task. k_p and k_v are the position and velocity gains. Thus, equation (5) represents the control structure for the trajectory tracking in marker space.

However, the coordinates associated with the positions of markers placed on the articulated body are not all independent. In order to address this dependency, we start by selecting an independent set m_1 of markers and a task, x_{m_1}, associated with this set. The control of the additional marker task is achieved by projecting the associated control in the null space of the Jacobian matrix associated with x_{m_1}.

Dynamic consistency between marker-set tasks is achieved by recursive projections of the associated control torques in the higher priority task null space [17]. For a marker set m_i, this is achieved by the dynamically consistent Jacobian $J_{m_i|m_{i-1}|\cdots|m_1}$

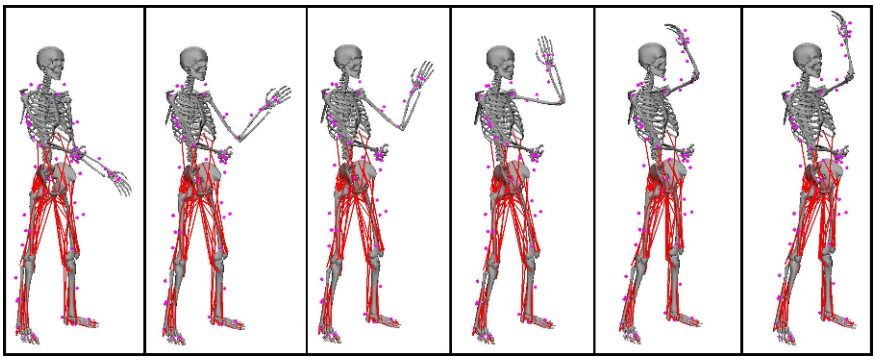


Fig. 3 A sequence of the reconstructed left hand throwing motion. Note that the dots correspond to the experimental markers attached to the subject.

defined as

$$J_{m_i|m_{i-1}|\cdots|m_1} = J_{m_i} N_{m_{i-1}} \cdots N_{m_1} \tag{7}$$

where N_{m_i} is the null space associated with the x_{m_i} marker-set task. For n marker-set tasks, the corresponding control torque vector is

$$\Gamma = J_{m_1}^T F_{m_1} + J_{m_2|m_1}^T F_{m_2|m_1} + \cdots + J_{m_n|m_{n-1}|\cdots|m_1}^T F_{m_n|m_{n-1}|\cdots|m_1} \tag{8}$$

2.2.2 Human Motion Control Hierarchy

The implementation of the marker space control formulation, described in Section 2.2.1, to the human model requires building a hierarchy of independent marker sets. Our approach for assigning markers to these task-sets is based on the observation that two markers can be controlled independently if they are separated by three degrees of freedom that span the space of motion. This principle is applied to the human model following its natural tree-like branching structure. The first level in the task hierarchy is formed by markers placed on the pelvis, its root, and on the hands and feet, its leaves. The following levels are constructed with markers placed on the intermediate links through assignments consistent with the above principle. Additional tasks, such as postures and dissipative forces, are included in the lowest level of the hierarchy.

Table 1 shows the resulting assignment for the human model. Note that the pelvis segment includes both RASI and LASI while the torso segment incorporates C7, CLAV, Lacromion, and Racromion markers that are illustrated in Fig. 1. The independent marker sets are then tracked through the entire movement sequence using this prioritization order. The remaining redundancy is labeled as the posture space of the marker tasks, containing all possible motions that do not affect marker tasks performance. The direct marker control framework allows us to synthesize

Table 1 The hierarchy of the controlled tasks in the marker space. See also Figs. 1 and 2.

Level 1	Level 2	Level 3	Level 4
Pelvis	Torso	Lupperarm	*Posture*
Lhand	Lthigh	Rupperarm	*Additional Behaviors*
Rhand	Rthigh	Ltibia	
Lmt5	Lforearm	Rtibia	
Rmt5	Rforearm		
LKnuckle	Lcalcaneus		
RKnuckle	Rcalcaneus		

any additional behavior by projecting its control into the marker task null-space and establishing a new priority.

3 Results and Real-Time Simulation

The motion reconstruction algorithm presented in Section 2.2.1 was tested on a sequence of human throwing motion described in Section 2.1 (Fig. 3). The reconstruction was executed by controlling the tasks in three-level marker space (Table 1) formed by independent sets of 22 experimental marker trajectories (see Fig. 1 and Table 1).

Desired and reconstructed trajectories for the throwing (left) hand were recorded during the simulation. Figure 4 illustrates the configurations of the throwing hand together with the desired and reconstructed hand trajectories. Figure 5 shows the reconstructed marker trajectories in time. Trajectory components (x, y and z) of the desired and reconstructed motions were given for both throwing left hand and left knuckle.

The results demonstrate the effectiveness of the reconstruction algorithm by tracking the trajectories with little error (0–4 cm). Our principal error source is the scapular elevation and depression of the left shoulder, which are not taken into account in the current human model. The error could be reduced by incorporating the movement of the scapula into the model. Overall, the results show that fast dynamic motions can be effectively reconstructed in near real-time.

4 Conclusions

We described our direct marker space control framework for reconstructing human motions by tracking captured marker trajectories with a simulated musculoskeletal model. The reconstruction was performed by successive projections into the null spaces of all tasks that are above it in the hierarchy formed in marker space. A control hierarchy which allows whole-body motion reconstruction was established

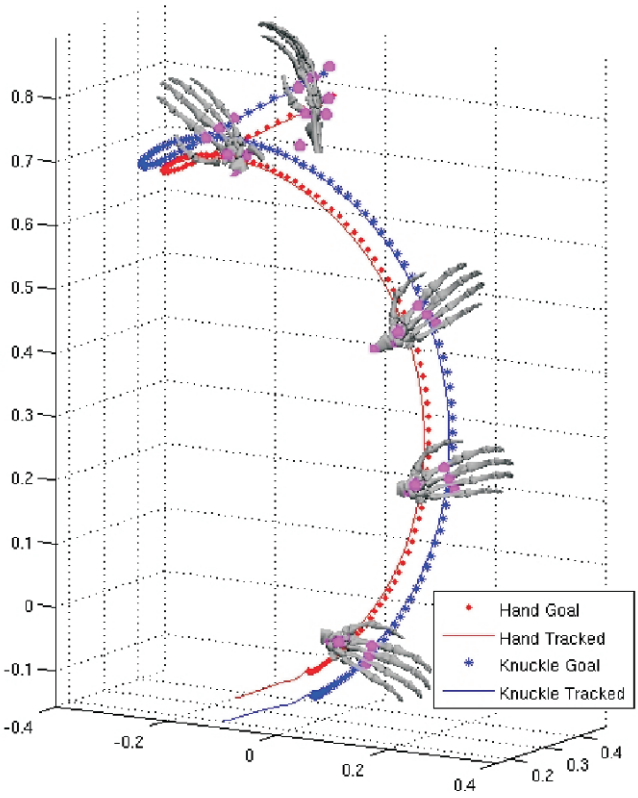


Fig. 4 Marker trajectories in task space: The left hand and knuckle marker trajectories demonstrate the effectiveness of the task space motion reconstruction algorithm. The markers are tracked while executing a left hand throwing motion. The tracked motion closely follows the recorder marker positions. Placing multiple markers on links (some hidden for clarity) enables the controller to track the position and orientation of the hand well.

and tested on a sequence of human throwing motions. Our framework provides an efficient way to map motion patterns to accurate musculoskeletal models without the need for inverse kinematics computations. It also runs in real-time. Our foremost limitation is modeling inaccuracy. For instance, the elevation and depression of the scapula, which enables the shoulder to translate, is not included in the model. The missing scapula movement limits the freedom of the shoulder in fast motions, and its absence dramatically degrades reconstruction near the limits of the arm's workspace.

Our marker space reconstruction methodology provides the full motion dynamics by operating in marker space and automatically resolving the kinematic constraints of the markers. The framework has been used to analyze high performance human motion such as that of athletes and martial art masters [9]. If we control a subset of the limbs, our controller will predict optimal motions for the remaining

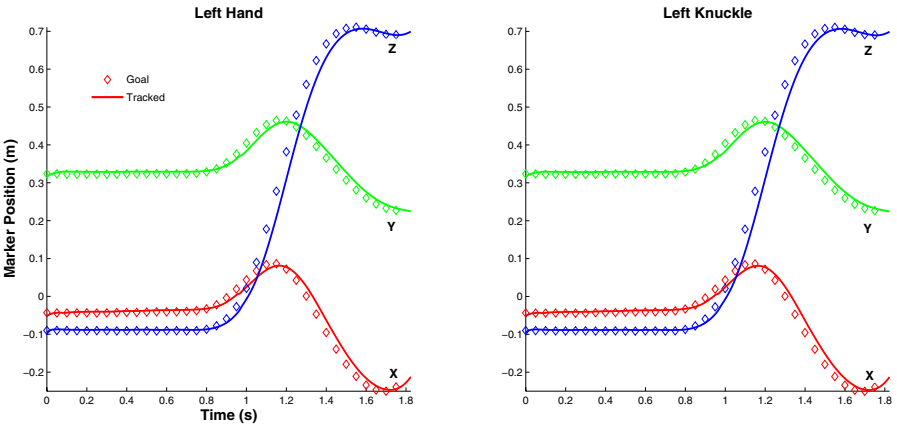


Fig. 5 Marker trajectories in time: The left hand and knuckle marker trajectories are tracked with little error, even when the motion is fast. While straight line motion is tracked with very low error (1 cm), fast dynamic motions introduce small error overshoots (2–4 cm) since the system is mildly underdamped.

limbs which can then be used to train subjects. Novel gaits that minimize the knee adduction moment can be predicted by subject-specific musculoskeletal modeling and trained using multi-modal feedback. Similarly, external knee loading that may lead to a non-contact anterior cruciate ligament (ACL) injury during running and cutting maneuvers can be estimated [18, 19], and new altered motion patterns with decreased loads on the knee joint can be modeled in this framework.

Acknowledgements The financial supports of the Simbios National Center for Biomedical Computing Grant (http://simbios.stanford.edu/, NIH GM072970) and Honda Company are gratefully acknowledged. The authors would like to thank Rebecca Shultz and Amy Silder for their assistance with the motion capture experiments. Many thanks to Roland Philippsen for his help during the preparation of the paper.

References

1. Lee J., Shin S.Y.: A hierarchical approach to interactive motion editing for human-like figures. In: *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*. ACM Press/Addison Wesley Publishing, pp. 39–48 (1999).
2. Choi, K., Ko, H.: On-line motion retargetting. In: *Proceedings Seventh Pacific Conference on Computer Graphics and Applications*. IEEE Computer Society, p. 32 (1999).
3. Savenko, A., Clapworthy, G.: Using motion analysis techniques for motion retargetting. In: *Proceedings Sixth International Conference on Information Visualization*, Vol. IV. IEEE Computer Society Press, p. 110 (2002).

4. Nakamura, Y., Yamane, K., Suzuki, I., Fujita, Y.: Dynamic computation of musculo-skeletal human model based on efficient algorithm for closed kinematic chains. In: *Proceedings of the 2nd International Symposium on Adaptive Motion of Animals and Machines*. Springer, New York (2003).
5. Grochow, K., Martin, S.L., Hertzmann, A., Popovic, Z.: Style-based inverse kinematics. In: *ACM Transactions on Graphics (TOG)*, Proceedings of the 2004 SIGGRAPH Conference, **23**(3), 522–531 (2004).
6. Nakamura, Y., Yamane, K., Suzuki, I., Fujita, Y.: Somatosensory computation for man-machine interface from motion capture data and musculoskeletal human model. *IEEE Transactions on Robotics*, **21**(1), 58–66 (2005).
7. Dariush, B., Gienger, M., Jian, B., Goerick, C., and Fujimura, K.: Whole body humanoid control from human motion descriptors. In: *IEEE Int. Conf. on Robotics and Automation*, pp. 2677–2684 (2008).
8. Demircan, E., Sentis, L., De Sapio, V., Khatib, O.: Human motion reconstruction by direct control of marker trajectories. In: Lenarčič, J., Wenger, P. (Eds.), *Advances in Robot Kinematics: Analysis and Design*. Springer, Dordrecht, pp. 263–272 (2008).
9. Khatib, O., Demircan, E., DeSapio, V., Sentis, L., Besier, T., Delp, S.: Robotics-based synthesis of human motion. *Journal of Physiology, Paris*, **103**, 211–219 (2009).
10. Delp, S.L., Loan, J.P., Hoy, M.G., Zajac, F.E., Topp, E.L., Rosen, J.M.: An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *IEEE Transactions on Biomedical Engineering* **37**, 757–767 (1990).
11. Holzbaur, K.R.S., Murray, W.M., Delp, S.L: A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control. *Ann. of Biomed. Eng.* **33**, 829–840 (2005).
12. Khatib, O., Brock, O., Chang, K., Conti, F., Ruspini, D., Sentis, L.: Robotics and interactive simulation. *Communications of the ACM*, **45**(3), 46–51 (2002).
13. Khatib, O., Sentis, L., Park, J., Warren, J.: Whole-body dynamic behavior and control of human-like robots. *International Journal of Humanoid Robotics*, **1**(1), 29–43 (2004).
14. Sentis, L., Khatib, O.: Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. *International Journal of Humanoid Robotics*, **2**(4), 505–518 (2005).
15. Dempster, W.T.: Space requirements of the seated operator. WADC Technical Report, Wright-Patterson Air Force Base, OH, pp. 55–159 (1955).
16. Khatib, O.: A unified approach for motion and force control of robot manipulators: The operational space formulation. *International Journal of Robotics Research*, **3**(1), 43-53 (1987).
17. Khatib, O.: Inertial properties in robotic manipulation: An object level framework. *International Journal of Robotics Research*, **14**(1), 19–36 (1995).
18. Besier, T.F., Lloyd, D.G., Ackland, T.R., Cochrane, J.L.: External loading of the knee joint during running and cutting manoeuvres. *Medicine in Science and Sports and Exercise*, **33**(7), 1168–1175 (2001).
19. Besier, T.F., Lloyd, D.G., Ackland, T.R., Cochrane, J.L.: Anticipatory effects on knee joint loading during running and cutting manoeuvres. *Medicine in Science and Sports and Exercise*, **33**(7), 1176–1181 (2001).

Overconstrained Mechanisms with Radially Reciprocating Motion

Guowu Wei and Jian S. Dai

Department of Mechanical Engineering, King's College London Strand, London WC2R 2LS, UK; e-mail: {guowu.wei, jian.dai}@kcl.ac.uk

Abstract. Inspired by the study of polyhedral and spheroidal linkages that perform radial motions, this paper uses the reciprocating motion of the PRRP chain to arrange a pentahedron and a cube and presents four overconstrained mechanisms with radially reciprocating motion. Characterized by the radial motion, the mechanisms developed in this paper may have potential use in robotic grasping and space exploration.

Key words: Radially reciprocating motion, overconstrained mechanisms, PRRP-chain, kinematics, robotic grasping.

1 Introduction

Linkages of radial motion have drawn great interest from kinematicians and mechanical engineers. In the past two decades, polyhedral and spheroidal linkages that perform radial motions aroused interests and had been applied to various fields in the customer products, architectural applications and space technologies. A typical representation is Wohlhart's [1] overconstrained spatial mechanisms as the turning tower, screwing tower and folding octoid. The study evolved into four new structures [2] of spheroid mechanisms including 'star-cube' which was then investigated by Wei and Dai [3]. The mobility of these overconstrained mechanisms can be modelled by examining order of the screw system in a similar relationship to a class of plane symmetric five and six bars that were investigated by Waldron [4] in 1969. In 1998, Dai and Rees Jones [5] proposed a family of foldable/erectable mechanisms in radial motion stemming from the study of origami cartons and reconfigurable packaging [6]. In 2001, Agrawal et al. [7] developed a single-mobility structure of a lattice of expanding polyhedral units. In 2004, Dai et al. [8] investigated a radially expandable and contractible sphere by applying the screw system analysis [9]. In 2006, Kusakari [10] presented a linkage coined 'radial trees'. In 2007, You [11] investigated the motion structure of the Hoberman flight ring and identified the motion characteristics that could be used to develop new structures.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_31,

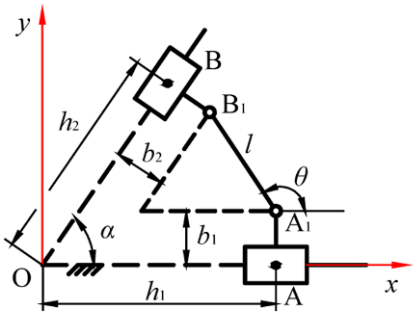


Fig. 1 A PRRP chain and a spatial 2-PRRP mechanism and its constraint graph.

Inspired by the above development, it is found in this paper that by properly arranging PRRP chains in the polyhedrons and removing the frames, new overconstrained mechanisms can be identified. These mechanisms are characterized by their radially reciprocating motions and may have great potential applications in many consumer products and in grasping, pressing machines, and antenna.

2 Characteristics of the PRRP Chain

Figure 1a gives a general PRRP chain. As is well known, this simple linkage can provide a motion such that when slider A moves towards point O, slider B moves away from point O and vice versa. This creates the reciprocating motion. To carry out the kinematic analysis, coordinate frame $\{O;x,y,z\}$ is established with origin fixed at point O and x-axis collinear with the center of prismatic joint A. l is the distance between two revolute joints A_1 and B_1, α is the angle between the central lines of the two prismatic joints. AA_1 is perpendicular to OA and b_1 indicates the distance between A and A_1. Similarly, BB_1 is perpendicular to OB and b_2 denotes the distance between B and B_1. Thus, from Fig. 1 kinematics of the PRRP chain can be derived as follows:

$$h_2 \sin\alpha - b_1 - b_2 \cos\alpha = l \sin\theta \tag{1}$$

$$h_2 \cos\alpha + b_2 \sin\alpha - h_1 = l \cos\theta \tag{2}$$

Taking squares of both sides of Eq. (1) and similar to Eq. (2) and adding the two resultant equations, the following quadratic equation is given with respect to h_1

$$h_1^2 + Bh_1 + C = 0 \tag{3}$$

where, $B = -2(h_2 \cos\alpha + b_2 \sin\alpha)$ and $C = h_2^2 + b_1^2 + b_2^2 - l^2 - 2b_1 h_2 \sin\alpha + 2b_1 b_2 \cos\alpha$.

Thus, it follows that

$$h_1 = -B \pm \sqrt{B^2 - 4C} \tag{4}$$

For a given input h_2, there are two possible values of the sliding variable h_1. They are, geometrically, the intersection of a circle of radius l centered at B_1 with a line through A_1 parallel to the x-axis. This kinematic property contributes to development of the overconstrained mechanisms with radially reciprocating motion proposed in the following sections.

Further, taking the time derivative of Eqs. (1) and (2) and combining the equations yield

$$\dot{h}_1 \cos\theta = \dot{h}_2 \cos(\theta - \alpha) \tag{5}$$

The above equation indicates two singular configurations for the PRRP chain. One is when $\theta = \pi/2 + \alpha$ when prismatic joint B is the actuated joint and the other is when $\theta = \pi/2$ when prismatic joint A is the actuated joint.

3 Construction of the Radially Reciprocating Motion Mechanism

The above characteristics of the PRRP chain of reciprocating motion can be used to construct the polyhedral and spheroidal mechanisms that have radially reciprocating motion (RRM). The principle of generating such linkages can be summarized by the following three steps.

Step 1: choose an appropriate polyhedron.

Step 2: place the PRRP chains with guides of two P joints in the polyhedron and align the guides with the edges or diagonals of the polyhedron such that an n-PRRP-combined chain is formed.

Step 3: remove the frame of the n-PRRP-combined chain, this actually removes P joints of the PRRP chains, to construct an overconstrained linkage that performs radially reciprocating motion.

It should be noted that in step 3, if an overconstrained linkage of mobility one cannot be constructed directly, kinematic property of the PRRP chain presented in Eq. (4) can be used such that relevant vertexes and corresponding links are added to the uncompleted linkage to form an overconstrained linkage of mobility one. This approach can be illustrated by an 8-PRRP-evolved mechanism and a 12-PRRP-evolved mechanism that are constructed by placing PRRP chains into a pentahedron and a cube following the aforementioned method and steps.

As shown in Fig. 2a, in a pentahedron OABCD with angle α between two adjacent edges a first PRRP chain is arranged in such a way that guides of the two prismatic joints collinear with edges E1 and E2. Further a second PRRP chain is introduced into the pentahedron with its two prismatic joints being aligned with

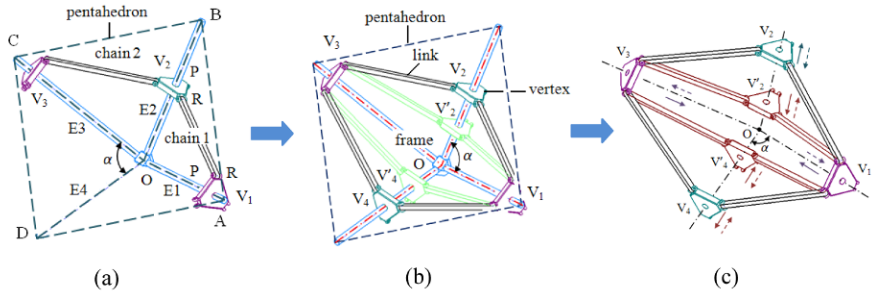


Fig. 2 Construction of type I 8-PRRP-evolved radially reciprocating motion mechanism.

edges E2 and E3. If the sliders (vertexes) of the two prismatic joints of chain 1 and chain 2 located at edge E2 are merged into one vertex as V_2, a 2-PRRP-combined chain can be formed in Fig. 2a. Taking the same procedure, and introducing two more PRRP chains into the pentahedron, a 4-PRRP-combined chain with a frame can be generated in Fig. 2b. Constructing in this way, it is evident that the 4-PRRP-combined chain can perform a radially reciprocating motion such that when vertexes V_1 and V_3 move outwards center O, vertexes V_2 and V_4 will move towards center O, and vice versa. This chain contains only triangle-shaped vertexes and it is termed as type I 4-PRRP-combined chain. It should be noted that in this work, *n*-PRRP-combined chains containing only triangle-shaped vertexes are termed as type I *N*-PRRP-combined chains, those containing square-shaped are termed as type II chains, and those containing pentadon-shaped as type III chains, and so on.

Now we try to remove the frame of the type I 4-PRRP-combined chain to see if an overconstrained mechanism can be found. Removing the frame, it is found that a normal eight-bar mechanism of mobility two is obtained. This indicates that removing frame does not directly result in an overconstrained mechanism of mobility one. As aforementioned, in order to construct an overconstrained mechanism from the type I 4-PRRP-combined chain, kinematic property of the PRRP chain revealed in Eq. (4) needs to be considered. Equation (4) reveals that for every one position of V_1 and V_3, vertexes V_2 and V_4 may occupy two positions as shown in Fig. 2b. Thus, adding two more vertexes V_2' and V_4' and corresponding links to the 4-PRRP-combined chain and removing the frame, an 8-PRRP-evloved mechanism can be generated in Fig. 2c as an overconstrained mechanism of mobility one with radially reciprocating motion. This results in such a way that when vertexes V_1 and V_3 move towards point O, vertexes V_2 and V_4 move away from point O accompanied with vertexes V_2' and V_4' moving towards point O and vice versa. The motion property of this mechanism enables it to have potential applications for robotic grasping and other consumer products.

Further, based on the principle that one cube comprises six identical pentahedrons, taking vertex angle $\alpha = 70.53^\circ$ and placing six type I 4-PRRP-combined chains into a cube by aligning their edges and merging their vertexes at center point O of the cube, a 12-PRRP-combined chain with a frame can be generated in Fig. 3a.

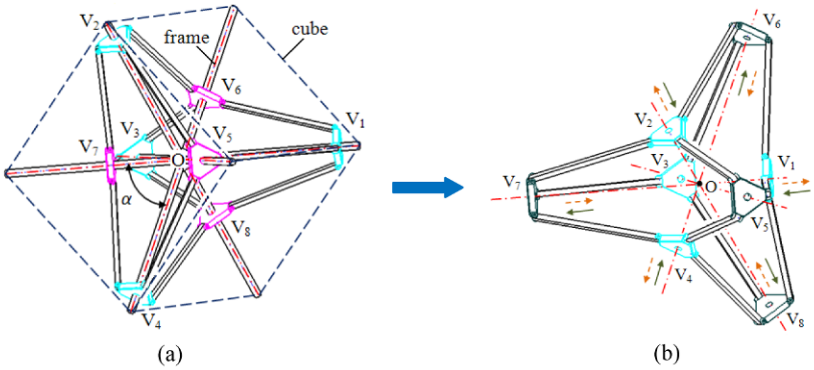


Fig. 3 Construction of a 12-PRRP-evolved radially reciprocating motion mechanism.

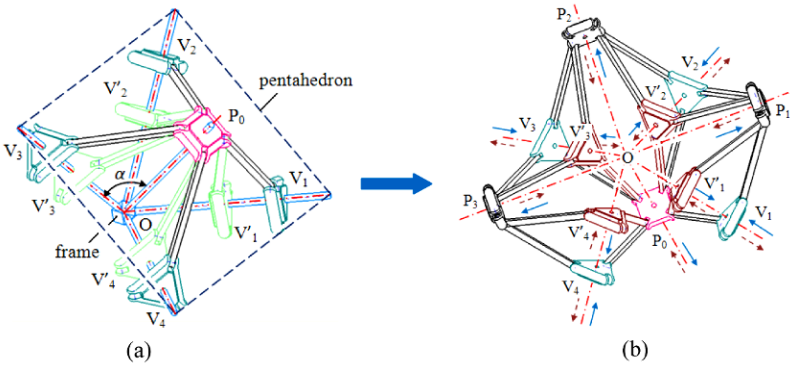


Fig. 4 Construction of a 16-PRRP-evolved radially reciprocating motion mechanism.

Removing the frame of the 12-PRRP-combined chain, a 12-PRRP-evolved mechanism is constructed. This is an overconstrained mechanism of mobility one that performs radially reciprocating motion with vertexes V_1, V_2, V_3 and V_4 moving towards central point O and vertexes V_4, V_5, V_6 and V_7 moving away from point O and vice versa. However, it should point out that this 12-PRRP-evolved mechanism is not a new mechanism. Using a different method, Wohlhart [2] has developed the same mechanism and the geometry and kinematics of this mechanism have been studied by Wei and Dai [3].

4 The New Radially Reciprocating Motion Mechanisms

Making use of the method presented in Section 3, two new overconstrained mechanisms are proposed below.

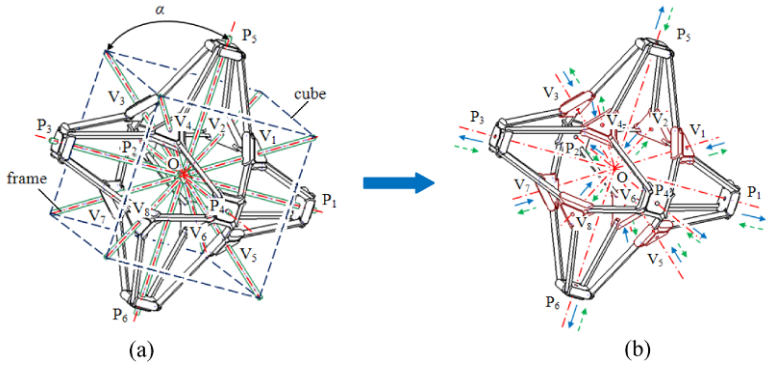


Fig. 5 Construction of a 24-PRRP-evolved radially reciprocating motion mechanism.

In the type I 4-PRRP-combined chain in Fig. 2b, another prismatic joint is introduced in such a way that the axis of the prismatic joint is perpendicular to the base of the pentahedron and passes through central point O. Further introducing a square-shaped vertex P located at this new prismatic joint and rearranging the PRRP chains in a way as indicated in Fig. 4a, a further 4-PRRP-combined chain is formed. Since this 4-PRRP-combined chain contains a square-shaped vertex, it is a type II 4-PRRP-combined chain. Adding four additional vertexes V_1', V_2', V_3' and V_4' and corresponding links and removing the frame, a new mechanism is expected to be generated. However, looking at Fig. 4a, it is evident that removing the frame does not result in a mechanism because in this case links V_1, V_2, V_3 and V_4, and V_1', V_2', V_3' and V_4' are not connected. In order to form a mechanism, in addition to adding of vertexes V_1', V_2', V_3' and V_4', three square-shaped vertexes P_1, P_2 and P_3 and corresponding links are further added to the type-II 4-PRRP-combined chain that a 16-PRRP-evolved overconstrained mechanism with mobility of one is constructed in Fig. 4b. The mechanism is a butterfly-shaped overconstrained mechanism with radially reciprocating mechanism in which when vertexes P, P_1, P_2 and P_3 move away from central point O, vertexes V_1, V_2, V_3, V_4 move towards O and vertexes V_1', V_2', V_3' and V_4' away from O and vice versa. It should point out that in the mechanism, the angle α between the two prismatic joints in a single PRRP chain is 54.74° Considering the kinematic property of this mechanism, it can be used as a multi-functional grasping robot.

Similar to construction of the 12-PRRP-evolved mechanism, integrating six of the type II 4-PRRP-combined chains into a cube by aligning edges and merging their vertexes at central point O of the cube, a 24-PRRP-combined chain with a frame is developed in Fig. 5a.

Further, by removing the frame of the 24-PRRP-combined chain, a 24-PRRP-evolved overconstrained mechanism is obtained in Fig. 5b. This is a mechanism of mobility one and can produce radially reciprocating motion in such a way that vertexes P_1, P_2, P_3, P_4, P_5 and P_6 move outwards point O, vertexes V_1, V_2, V_3, V_4, V_5, V_6, V_7 and V_8 move towards point O and vice versa.

The 24-PRRP-evolved mechanism is a new overconstrained mechanism based on pure revolute joints that performs radially reciprocating motion. The kinematics of this mechanism can be studied similar to that of the 12-PRRP-evolved mechanism in [3]. Since vertexes P_1, P_2, P_3, P_4, P_5 and P_6 are symmetrically distributed around a sphere centered at point O, once the position of one of the these vertexes $\boldsymbol{p}_i$ is given, the positions of the other vertexes $\boldsymbol{p}_j$ can be calculated as

$$p_j = {}^j_i\mathbf{R}^T p_i \quad (i \neq j) \tag{6}$$

where ${}^j_i\mathbf{R}^T$ denotes the transpose of the rotation matrix describing coordinate frame of $\boldsymbol{p}_j$ relative to coordinate frame of $\boldsymbol{p}_i$.

Similarly, as vertexes V_1, V_2, V_3, V_4, V_5, V_6, V_7 and V_8 are symmetrically arranged around the same spherical, once the position of one of these vertexes $\boldsymbol{p}_{Vk}$ are obtained, the position of the other vertexes $\boldsymbol{p}_{Vl}$ can be derived by Eq. (6). Further, since the mechanism is evolved from the PRRP chain, the magnitudes of positions of vertexes P_1, P_2, P_3, P_4, P_5 and P_6, and positions of vertexes V_1, V_2, V_3, V_4, V_5, V_6, V_7 and V_8 must comply with Eq. (4). In addition, the magnitudes of the velocities of vertexes P_1, P_2, P_3, P_4, P_5 and $P_6$and that of vertexes V_1, V_2, V_3, V_4, V_5, V_6, V_7 and V_8 must satisfy

$$|\dot{\boldsymbol{p}}_i| \cos\theta = |\dot{\boldsymbol{p}}_{Vk}| \cos(\theta - \alpha) \tag{7}$$

where θ is the rotation angle of the link in the mechanism, α is the angle as illustrated in Fig. 5a, and $i = 1,2,\ldots,6$ and $k = 1,2,\ldots,8$. With the above equation, singularity of the mechanism can be revealed.

From the above construction and analysis of the new overconstrained mechanisms obtained, one can find that these mechanisms are characterized by their radially reciprocating motion. This is interesting motion that the mechanisms presented herein can be applied to whereever radial motion is required. Further, from Eq. (4), by considering the kinematic property of a PRRP chain, adding eight vertexes V'_1, V'_2, V'_3, V'_4, V'_5, V'_6, V'_7 and V'_8 and corresponding links to the mechanism to the 12-PRRP-evolved mechanism, a manipulator with eight end-effectors driven by only one actuator can be obtained. And so can be done to the 24-PRRP-evolved mechanism to obtain another manipulator with multiple end-effectors.

5 Conclusions

From investigation of the polyhedral and spheroidal mechanisms, choosing a pentaheraon and a cube as a frame base and utilizing the kinematic property of a PRRP chain, a set of *n*-PRRP-evolved overconstrained mechanisms that perform radially reciprocating motion were obtained in this paper. Using the method proposed in this paper, a series of overconstrained mechanisms with radially reciprocating motion can be designed by selecting proper polyhedrons. The analysis of the *n*-PRRP-

evolved mechanisms will have potential use for manipulators with multiple end-effectors. The *n*-PRRP-evolved radially reciprocating mechanism proposed in the paper may have potential applications in consumer products such as manipulators, pressing machines, and antenna for space exploration.

References

1. Wohlhart, K.: Heureka octahedron and brussels folding cube as special cases of the turing tower. In *Proceedings of the 6th International Symposium on Teoria si Practica Mecanismelor*, Vol. 2, pp. 303–311 (1993).
2. Wohlhart, K.: New overconstrained spheroidal linkages. In *Proceedings of the 9th World Congress on the Theory of Machines and Mechanisms*, Milano, Vol. 1, pp. 149–153 (1995).
3. Wei, G.W., Ding, X.L., and Dai, J.S.: Geometry and kinematic analysis of the Hoberman switch-pitch ball and its variant. In *Proceeding of Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, DETC2009-87329 (2009).
4. Waldron, K.J.: Symmetric overconstrained linkages. *Transactions of the ASME: Journal of Engineering for Industry*, **91**(1), 158–164 (1969).
5. Dai, J.S. and Rees Jones, J.: Mobility in metamorphic mechanisms of foldable/erectable kinds. *Transactions of ASME: Journal of Mechanical Design*, **121**(3), 375–382 (1999).
6. Zhang, L. and Dai, J.S.: Reconfiguration of spatial metamorphic mechanisms. *Transactions of the ASME: Journal of Mechanisms and Robotics*, **1**(1), 011012_1-8 (2009).
7. Agrawal, S.K., Kumar, S., Yim, M., and Suh, J.W.: Polyhedral single degree-of-freedom expanding structures. In *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 4, pp. 3338–3343 (2001).
8. Dai, J.S., Li, D., Zhang, Q., and Jin, G.: Mobility analysis of a complex structured ball based on mechanism decomposition and equivalent screw system analysis. *Mechanism and Machine Theory*, **39**(4), 445–458 (2004).
9. Dai, J.S. and Rees Jones, J.: Null space construction using cofactors from a screw algebra context. *Proc. Royal Society, Lond. A: Mathematical, Physical and Engineering Sciences*, **458**(2024), 1845–1866 (2002).
10. KusaKari, Y.: On Reconfiguring radial trees. *IEICE Trans. Fundamentals*, Vol. E89-A, **5**, 1207–1214 (2006).
11. You, Z.: Motion structures extend their reach. *Materials Today*, **10**(12), 52–57 (2007).

Control of Bipedal Turning While Running

A.D. Perkins and K.J. Waldron

Stanford University, Stanford, CA, USA; e-mail: alexp2@stanford.edu

Abstract. There remain unsolved control problems for running bipedal robots. In this work, dynamic principles are used to develop a set of heuristics for executing running turns. A specific system is then considered, with 17 DOF, 7 of which are active, a low inertia torso, and mechanical coupling between the knee and ankle joints. For this system, the heuristics are distilled into control laws governing the active joint torques. These control laws are implemented in simulation to achieve stable and smooth turning while running.

Key words: Running, turning, control, biped, robot.

1 Introduction

As the robotics revolution continues into the 21st century, legged robots are poised to enter service in roles ranging from military transports to in-home servants to search and rescue aids. In order to do this, they must be able to maneuver quickly. To date, however, no robotics systems have been created which are both fast and robust. Some, like the 230 g, 160 mm SPRAWL [1], maneuver well, but are too small to be practically useful. Others, like Raibert's biped [2], are quite dynamic, capable of runs, jumps, and aerial maneuvers, but require very specific system parameters that render them ineffective for general applications. Still others, like Boston Dynamics' BigDog [3] and Honda's ASIMO [4] are large and robust, but aren't designed for high speed locomotion.

High speed locomotion comprises many maneuvers. Perhaps the most fundamental is steady-state running, which has been studied extensively, including notable work with Raibert's biped [2], ASIMO [4], HRP [5], and RABBIT [6]. Beyond steady-state running, other useful dynamic maneuvers include accelerating and decelerating, turning, and running jumps. These maneuvers have been studied for application to the quadruped KOLT [7], but remain under-examined for bipedal systems. We have previously considered accelerating and decelerating [8]; here, we will explore turning.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_32,

The motivation for accelerating, by which we mean both changing speed and changing direction, comes in the form of practical obstacles. Being constrained to a fixed speed and direction is not incredibly useful. Eventually, it will become necessary to change speed, which we have discussed in [8], or direction, which we discuss here. Indeed, it is difficult to imagine a realistic task which does not require changing direction. Thus, the aim of this work is to develop a control strategy for turning while running in bipedal systems.

Before proceeding, it is important to define some terminology. Consider a body-fixed frame with its origin at the midpoint of the line connecting the hip centers. Rotation about the lateral, longitudinal, and vertical axes are called "pitch," "roll," and "yaw" respectively. The angle of a leg segment in the pitch plane is called "swing" and the angle of a leg segment in the roll plane is called "spread". "Foot-strike" refers to the instant at which the foot first comes into contact with the ground and "toe-off" refers to the instant at which the foot completely leaves contact with the ground. The "system heading" refers to the orientation of the projection of the longitudinal axis in the horizontal plane; i.e. the direction the system is facing. The "path heading" refers to the direction of the line tangent to the motion of the midpoint of the line connecting the hip centers (which is the origin of the machine frame); i.e. the direction the system is moving. Thus, if the robot were running sideways, the system heading and path heading would be at right angles to one another. In order to effectively turn, it is necessary to alter both the system heading and the path heading identically, or said another way, to alter the path heading and keep the system heading and path heading aligned.

The remainder of this work is organized as follows: Section 2 formulates a heuristic control method, Section 3 describes a specific system with 17 DOF, including 3-link articulated legs and a low inertia torso, Section 4 refines the heuristics into specialized control laws for this system, Section 5 presents simulated results of this system executing a variety of turns, and Section 6 concludes.

2 Heuristics

One approach to controlling legged robotics is heuristic control, in which a set of heuristics, or rules, are developed to govern the desired maneuver. These can then be translated into joint-specific control laws [2]. Heuristic control is less sensitive to model inaccuracies than model-based control, does not require large training sets of data like machine learning techniques, and is more robust to disturbances than central pattern generators [8].

For heuristic control, very little needs to be known about the system *a priori*. It is assumed here that the system being controlled holds to a few loose requirements: it is bipedal and symmetric about the sagittal plane, it has articulated hips connecting the torso to the legs, and the hips and the knees are fully actuated.

In order to develop the heuristics that form the top level of the hierarchy of our controller, the dynamic principles that govern a running biped must be considered.

The first consideration is the attitude of the torso. It is desirable to minimize the pitch impulse on the torso to avoid violent, and potentially damaging, pitching during contact. Thus, the net impulse in the sagittal plane should pass through the torso COM. In legged systems, the ground reaction force has been shown to act on a line from the point of ground contact through the hip [9]. Thus, ideally the COM of the torso should be kept on this line. To do this, however, requires constantly repositioning the torso during ground contact. As a simpler alternative, we propose:

TRN.1 *Keep the torso upright.*

As previously mentioned, it is desired to keep the system heading and path heading aligned. As the leg thrusts are applied to the hip axis and not directly to the torso, it is the hip axis that will yaw. For a system with the ability to turn its torso with respect to the hip about the vertical axis, it is thus necessary to add a heuristic to prevent just the path turning:

TRN.2 *Keep the torso aligned with the hip axis in the horizontal plane.*

In legged mechanical systems, the ground reaction forces during impact can be as much as 3-5 times the weight of the system [10]. The shock associated with such high impulses can be damaging to the delicate electronic systems required to sense and control the system. Thus, it is vital that the jarring effects of the impact be minimized. This leads to:

TRN.3 *Land in a pose removed from the singularity of the leg.*

If the inertia of the torso is very high, as it was in Raibert's biped, then the torso can be used to posture the leg prior to foot-strike [2]. If the inertia of the torso is low, then any attempt to posture the legs will result in the immediate and severe tilting of the torso while hardly moving the legs at all [11]. Since it is desirable for these heuristics be applicable to a wide range of systems, it is conservative to assume a low inertia torso. This leads to the third heuristics:

TRN.4 *Position the leg that will next strike using the other leg.*

As observed in [8], forward acceleration is a paired process; that is, it must be done over the course of at least two successive strides. The first stride applies a thrust to one hip, which in turn induces a yaw rate in the hip axis. This must then be countered by applying a similar thrust during the next stride. A first attempt at turning can therefore be made by accelerating with one foot while decelerating with the other, which would cause a turn in the direction of the decelerating foot (that is, if the right foot is decelerating, the biped will turn right). In a way, this is how most robotic walkers turn, shuffling one foot backwards (deceleraing) and the other foot forward (accelerating) [4, 5]. However, when the initial acceleration is applied, a yaw rate is induced only in the system, while the path heading remains unchanged. Thus, the first stance should do more than cause a forward acceleration; it should cause a lateral acceleration as well, directed toward the center of the turn, which will cause both the system and path to begin turning. This leads to the sequence shown in Fig. 1. It is worth noting that the ordering of the steps in this sequence is arbitrary,

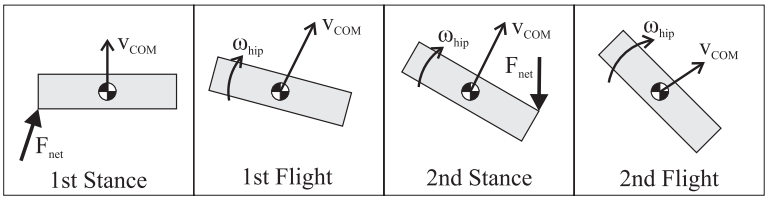


Fig. 1 Appropriate force application for inducing a yaw rate. During the first (accelerating) stance, the net force F_{net} is pointed forward and toward the inside of the turn, while during the second (decelerating) stance, the net force is pointed rearward and toward the inside of the turn.

meaning that a turn in either direction can commence on either foot, thus eliminating the need for anticipation. The sequence in Fig. 1 leads to the next two heuristics, which govern both the forward and lateral foot placement at foot-strike:

TRN.5 *Swing the outside leg closer to vertical at foot-strike, and swing the inside leg farther in front of the hip at foot-strike.*

TRN.6 *Spread both legs farther to the outside of the turn at foot-strike.*

The final heuristic comes from the fact that all of the previous heuristics, especially TRN.5 and TRN.6, combine to *induce* a turn. Thus, some consideration must be made toward *reducing* the turn back to a straight run. In order to reduce a turn in one direction, it is sufficient to induce a turn in the opposite direction of equal magnitude. Thus:

TRN.7 *To cease turning in one direction, perform a turn in the opposite direction of equal magnitude.*

3 Model

As mentioned previously, in order to convert the heuristics into realizable control laws, the specific system for implementation must be considered. For this work, we consider a system with 17 DOF, shown in Fig. 2. This system has 17 DOF: 6 DOF for position and orientation of the medial-lateral hip axis, 1 DOF for rotation at the pelvis of the HAT about the superior-inferior axis, and 5 DOF in each leg – 2 DOF articulated hips, 1 DOF articulated knees, and 2 DOF articulated ankles. Of these, the pelvis, hip, and knee joints are actively actuated, giving 7 actuated DOF.

There are two main aspects of this system which merit special consideration. First, the model has mass and inertia values to match a mechanical system constructed of commonly available materials. For example, the legs are constructed out of aluminum square tubing with 3.2 mm wall thickness and 25 mm width. The full list of parameters is given in Table 1. The important result is that the torso has relatively low inertia compared to the legs, meaning that the torso cannot, in fact, be used to position the legs during flight.

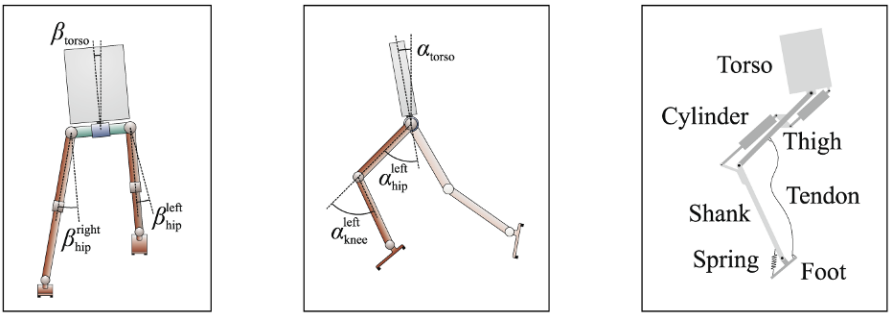


Fig. 2 The articulated bipedal system under consideration, with 17 DOF (7 actuated), showing: (left) the front view (forward motion out of the page); (center) the side view (forward motion to the left); and (right) the actuator layout. As shown, the torso roll angle $\beta_{\text{torso}} > 0$, the hip spread angles $\beta_{\text{hip}}^{\text{right}} > 0$ and $\beta_{\text{hip}}^{\text{left}} < 0$, the torso pitch angle $\alpha_{\text{torso}} > 0$, the left hip swing angle $\alpha_{\text{hip}}^{\text{left}} < 0$, and the left knee swing angle $\alpha_{\text{knee}}^{\text{left}} < 0$.

Table 1 Model parameters for the second simulated system, which are taken from commonly available mechanical materials.

Link	Length (m)	Mass (kg)	Inertia (kg m^2) (lateral, longitudinal, vertical)
HAT	0.2	12	2, 2, 2
Quad	0.45	5.6	1.14, 1.14, 0.11
Shank	0.45	1.1	0.22, 0.22, 0.02
Foot	0.21	0.45	0.01, 0.01, 0.01

The second aspect of this system which merits special consideration is the actuation of the ankle. As noted in [11], the ankle plays a crucial role in running, so control is necessary. However, placing an actuator at the ankle further increases the leg inertia, which in turn increases the energetic cost of running [12]. Biologic systems avoid this by using biarticular muscle/tendon groups, which provide mechanical coupling between multiple joints. Thus, an inelastic tendon is used to couple the ankle and knee joints of the system, eliminating the need for an active actuator at the ankle.

4 Control Laws

Given this system, the heuristics developed in Section 2 can now be converted into specialized control laws. Given a desired path heading $\gamma_{\text{path}}^{\text{des}}$ and current path heading $\gamma_{\text{path}}^{\text{act}}$, Eqs. (1–2), which give expressions for $\alpha_{\text{hip}}^{\text{fs,in}}$ and $\alpha_{\text{hip}}^{\text{fs,out}}$, the target hip swing angles at foot-strike for the inside and outside legs, are obtained by combining TRN.5 with proportional control.

$$\alpha_{\text{hip}}^{\text{fs,in}} = \tilde{\alpha}_{\text{hip}}^{\text{fs}} \left(1 - k_{SW}^{\text{in}}(\gamma_{\text{path}}^{\text{act}} - \gamma_{\text{path}}^{\text{des}})\right) \tag{1}$$

$$\alpha_{\text{hip}}^{\text{fs,out}} = \tilde{\alpha}_{\text{hip}}^{\text{fs}} \left(1 + k_{SW}^{\text{out}}(\gamma_{\text{path}}^{\text{act}} - \gamma_{\text{path}}^{\text{des}})\right) \tag{2}$$

where k_{SW}^{in} and k_{SW}^{out} are constants and $\tilde{\alpha}_{\text{hip}}^{\text{fs}}$ is the nominal hip swing angle at foot-strike for steady-state running. Similarly, TRN.6 yields Eqs. (3–4), which give expressions for $\beta_{\text{hip}}^{\text{fs,in}}$ and $\beta_{\text{hip}}^{\text{fs,out}}$, the target hip spread angles at foot-strike for the inside and outside legs.

$$\beta_{\text{hip}}^{\text{fs,in}} = \tilde{\beta}_{\text{hip}}^{\text{fs}} \left(1 + k_{SP}^{\text{in}}(\gamma_{\text{path}}^{\text{act}} - \gamma_{\text{path}}^{\text{des}})\right) \tag{3}$$

$$\beta_{\text{hip}}^{\text{fs,out}} = \tilde{\beta}_{\text{hip}}^{\text{fs}} \left(1 - k_{SP}^{\text{in}}(\gamma_{\text{path}}^{\text{act}} - \gamma_{\text{path}}^{\text{des}})\right) \tag{4}$$

where k_{SP}^{in} and k_{SP}^{out} are constants and $\tilde{\beta}_{\text{hip}}^{\text{fs}}$ is the nominal hip spread angle at foot-strike for steady-state running. The difference in sign between Eq.(3) and Eq.(4) stems from the definition of the spread angle β_{hip} as being positive under abduction. Thus, to move both feet toward the outside of the turn, the inside foot needs to be abducted further, while the outside foot needs to be adducted.

The individual joint torques can be derived from these control laws and the remaining heuristics using any of a variety of techniques; for the results presented here, simple proportional-derivative (PD) control is used. For example, TRN.2 can be used to specifiy the pelvic torque about the vertical axis $\tau_{\gamma,\text{pelvis}}$ as seen in Eq.(5).

$$\tau_{\gamma,\text{pelvis}} = \text{PD}\left(\gamma_{\text{torso}} - \gamma_{\text{path}}^{\text{act}}\right) \tag{5}$$

where γ_{torso} is the heading of the torso.

5 Results

In order to test the heuristics and control laws developed above, a dynamic simulation of the system in Fig. 2 is created using the method outlined in [13]. The symbolic manipulator AutoLev is used to create the equations of motion of the system, and these equations are imported into a dynamic simulator in C++ which uses a variable step Runge–Kutta–Merson algorithm for numerical integration (see [14] for details).

For this simulation, two key assumptions were made. First, it was assumed that the friction between the ground and the foot was sufficient at all times to prevent slipping. This assumption is validated by the fact that the highest required coefficient of friction for the results presented here was $\mu_{\text{max}} < 1.8$, while observed coefficients of static friction between a rubber foot sole and steel ground plate fall in the range of 3–10 [15]. The other main assumption is that all collisions are inelastic. Collisions are modeled using the generalized impulse-momentum method [16]. The decision

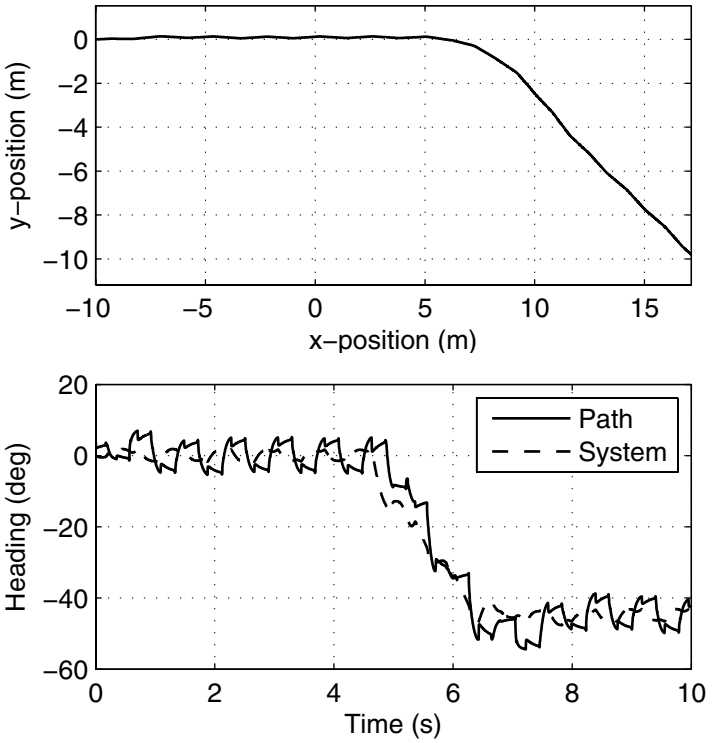


Fig. 3 Top: System path in the horizontal plane. Bottom: Path heading (solid line) and system heading (dashed line). The system starts out running along a 0°, heading at 3 m/s. It then turns 45° to the right and resumes running straight along this new heading.

to model collisions as inelastic is based on the observation that, in biologic systems performing dynamic gaits, the kinetic energy in the foot is deliberately sacrificed to maintain inelastic collisions [17].

Figure 3 shows the system running at 3 m/s while executing a 45° turn at 22 deg/s. The top plot demonstrates that the turn is successfully and smoothly executed. The bottom plot reveals that the system and path headings both turn to the new heading. Interestingly, the system executes the turn slightly ahead of the path. This indicates that the lateral force generated by TRN.6 (and subsequently Eqs. (3–4)) is a little too high relative to the forward force generated by TRN.5. Despite this, the system neatly executes the commanded turn.

6 Conclusion

A set of heuristics based on observations of physical principles was developed for turning while running. These heuristics are simple and easily applied to bipedal robots. These heuristics were then converted into simple proportional control laws to determine relevant parameters, which were fed into proportional-derivative control laws to determine appropriate joint torques. Using these control laws, stable turning was achieved.

References

1. McClung, A., Cutkosky, M., and Cham, J.: Rapid maneuvering of a biologically inspired hexapedal robot. In *ASME Intl. Mechanical Engineering Congress and Expo*, Anaheim, CA, November (2004).
2. Raibert, M.: *Legged Robots that Balance*. MIT Press, Cambridge, MA (1986).
3. Playter, R., Buehler, M., and Raibert, M.: BigDog. In *Unmanned Systems Technology VIII*, pp. 20–26, Orlando, FL (2006).
4. Hirose, M. and Ogawa, K.: Honda humanoid robots development. *Phil. Trans. of the Royal Society A*, **365**(1850), 11–19 (2007).
5. Kajita, S., Nagasaki, T., Kaneko, K., and Hirukawa, H.: ZMP-based biped running control. *IEEE Robotics and Automation Magazine*, **14**(2), 61–72 (2007).
6. Chevallereau, C., Abba, G., Aoustin, Y., Plestan, F., Westervelt, E., Canudas de Wit, C., and Grizzle, J.: RABBIT: A testbed for advanced control theory. *IEEE Control Systems Magazine*, **23**(5), 57–79 (2003).
7. Palmer, L. and Orin, D.: Attitude control of a quadruped trot while turning. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, October, pp. 5743–5749 (2006).
8. Perkins, A. and Waldron, K.: Heuristic control of bipedal running: Steady-state and accelerated. In *Intl. Conf. on Climbing and Walking Robots*, Istanbul, Turkey, June, pp. 575–582 (2009).
9. Alexander, R.: Energy-saving mechanisms in walking and running. *J. Exp. Biol.*, **160**, 55–69 (1991).
10. Hutchinson, J.: Biomechanical modeling and sensitivity analysis of bipedal running ability. I. Extant taxa. *J. Morphology*, **262**, 421–440 (2004).
11. Perkins, A. and Waldron, K.: The effects of ankle stiffness on articulated planar hopping. In *CISM-IFToMM Symposium on Robot Design, Dynamics, and Control*, Tokyo, Japan, July, pp. 521–527 (2008).
12. Abdallah, M.: Mechanics motivated control and design of biped running. PhD Dissertation, Stanford University, Stanford, CA (2007).
13. Perkins, A., Abdallah, M., Mitiguy, P., and Waldron, K.: A unified method for simulating multi-body systems subject to stick-slip friction and intermittent contact. In *IEEE Intl. Conf. on Intelligent Robotics and Systems*, Nice, France, September, pp. 2311–2316 (2008).
14. Lance, G.: *Numerical Methods for High-Speed Computers*. Iliffe & Sons, London (1960).
15. Castelli, V.: Friction. In *Marks' Standard Handbook for Mechanical Engineers*. McGraw Hill, New York (1996).
16. Kane, T. and Levinson, D.: *Dynamics: Theory and Applications*. McGraw Hill, New York (1985).
17. Waldron, K., Estremera, J., Csonka, P., and Singh, S.: Thinking about bounding and galloping using simple models. In *Intl. Conf. on Climbing and Walking Robots*, Singapore, September, pp. 445–453 (2008).

Geometrico-Static Analysis of Under-Constrained Cable-Driven Parallel Robots

M. Carricato[1] and J.-P. Merlet[2]

[1]*DIEM – Department of Mechanical Engineering, University of Bologna, Italy;*
e-mail: marco.carricato@mail.ing.unibo.it
[2]*INRIA Sophia Antipolis – Méditerranée, Sophia Antipolis, France;*
e-mail: jean-pierre.merlet@sophia.inria.fr

Abstract. This paper studies the kinematics and statics of cable-driven parallel robots with less than six cables, in crane configuration. A geometrico-static model is provided, together with a general procedure aimed at effectively solving, in analytical form, the inverse and direct position problems. The stability of equilibrium is assessed within the framework of a constrained optimization problem, for which a purely algebraic formulation is provided. A spatial robot with three cables is studied as an application example.

Key words: Under-constrained cable-driven parallel robots, geometric analysis, static analysis.

1 Introduction

Cable-driven parallel robots (CDPRs) are ordinarily referred to as fully-constrained or under-constrained, depending on whether all six degrees of freedom (dofs) of the moving platform are controlled or not [1]. It is well known that, in the general case, fully-constrained CDPRs require at least seven cables, and only six in crane configuration, namely when gravity acts as an additional cable. The distinction between the two aforementioned categories of robots is somewhat fictitious, since a theoretically fully-constrained CDPR is, in considerable parts of its workspace, actually under-constrained, namely when a full restraint would require a negative tension in one or more cables. Permanently under-constrained CDPRs with less than six cables are furthermore used in a number of applications, such as rescue operations, in which a limitation of dexterity is acceptable in order to decrease complexity, set-up time, likelihood of cable interference, etc.

The above considerations motivate a careful study of under-constrained robots. However, while fully-constrained CDPRs have been extensively investigated [2], few studies have been dedicated to under-constrained ones [3–5]. A major challenge in the study of these robots consists in the intrinsic coupling between kinematics and statics (or dynamics). Indeed, while in a fully-constrained CDPR the platform posture is determined in a purely geometric way by assigning the cable lengths (provided that all cables are under tension), in an under-constrained CDPR the pose

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_33,

depends on both cable lengths and equilibrium equations. Consequently, kinematics and statics must be dealt with simultaneously. Furthermore, as the platform posture depends on the applied load, it may change due to external disturbances. Hence, it is of great importance to investigate equilibrium stability [5, 6].

This paper presents a kinematic and static study of under-constrained nn-CDPRs, in crane configuration. An under-constrained nn-CDPR is controlled by n cables and it exhibits n distinct anchor points on both the base and the platform, with $n < 6$. A geometrico-static model is presented, together with a general procedure aimed at effectively solving, in analytical form, the inverse and direct position problems. These consist in determining the overall robot configuration (and cable tensions) once n configuration variables (e.g. platform posture coordinates or cable lengths) are given. The problem of equilibrium stability is formulated as a constrained optimization problem, and a purely algebraic method, which rules out the need of differentiation, is provided. The geometrico-static study of a general 33-CDPR is outlined as an application example.

2 Geometrico-Static Model

Let a mobile platform be connected to a fixed base by n cables, with $2 \leq n \leq 5$. A_i and B_i are, respectively, the anchor points of the ith cable on the frame and the platform, and $\mathbf{s}_i = B_i - A_i$ (Fig. 1(a)). The set $\mathcal{C}$ of theoretical geometrical constraints imposed on the platform comprises the relations

$$|\mathbf{s}_i| = \sqrt{\mathbf{s}_i \cdot \mathbf{s}_i} = \rho_i, \quad i = 1 \ldots n, \tag{1}$$

where ρ_i is the length of the ith-cable, which is assumed, for apparent practical reasons, strictly positive (so that, as a consequence, $B_i \not\equiv A_i$).

Since only n geometrical constraints are enforced, the platform preserves $6 - n$ degrees of freedom, with its posture being determined by equilibrium laws. If $Q\$_e$, with $Q > 0$, is an arbitrary external wrench acting on the platform (including inertia forces, in the case of dynamic conditions) and $(\tau_i/\rho_i)\,\$_i$ is the force exerted by the ith cable ($\$_e$ and $\$_i/\rho_i$ are assumed to be unit screws), then

$$\sum_{i=1}^{n} \frac{\tau_i}{\rho_i}\$_i + Q\$_e = \mathbf{0}, \tag{2}$$

with

$$\tau_i \geq 0, \quad i = 1 \ldots n. \tag{3}$$

Equations (1)–(2) amount to $6 + n$ scalar relations involving $6 + 2n$ variables, namely, the cable tensions and lengths, and the variables parameterizing the platform posture. In general, a finite set of system configurations may be determined if n of such variables are assigned.

In this paper, only *static* equilibrium is considered and $Q\$_e$ is assumed to be a *constant* force applied on a point G of the platform (e.g. the platform weight acting through its center of mass). Hence, Eqs. (1)-(2) are algebraic, or may be easily rendered so. If Eq. (2) is written as

$$\underbrace{\left[\$_1 \cdots \$_n \ \$_e\right]}_{\mathbf{M}} \begin{bmatrix} (\tau_1/\rho_1) \\ \vdots \\ (\tau_n/\rho_n) \\ Q \end{bmatrix} = \mathbf{0}, \tag{4}$$

$\mathbf{M}$ is a $6 \times (n+1)$ matrix only depending on the platform posture and equilibrium is possible only if

$$\operatorname{rank}(\mathbf{M}) \leq n, \tag{5}$$

namely, if the cables and the line of action of $\$_e$ span the same n-dimensional system of lines. Within the domain of rigid-body mechanics, the problem is *statically determinate* if the equality holds, *indeterminate* otherwise. In the former case, it is always possible to replace Eq. (4) with $6-n$ scalar relations that do not contain the unknowns τ_i, $i = 1 \ldots n$. In fact, the linear dependence of $\$_1, \ldots, \$_n$ and $\$_e$ is a purely geometrical condition. A most straightforward strategy consists in computing cable tensions by way of n linearly independent relations chosen within Eq. (4), then substituting them back into the remaining ones. The resulting equations, however, exhibit a remarkable complexity. A more convenient strategy consists in setting all $(n+1) \times (n+1)$ minors of $\mathbf{M}$ equal to zero, which amounts to $\binom{6}{n+1}$ scalar relations, among which $6-n$ linearly independent ones may be suitably chosen. By such an approach, the resulting equations are significantly simpler. Furthermore, since they do *not* comprise cable lengths, they lead to a partial decoupling of the system equations, with cable lengths only appearing in Eq. (1). Such an approach may also be applied when the problem is statically indeterminate.

Depending on the variables designated as input, one may tackle an inverse geometric problem (IGP), if n variables concerning the platform posture are assigned, or a direct one (DGP), if cable lengths are given. The IGP takes particular advantage of the partial decoupling of system equations, since the platform configuration may be computed by simply solving the $6-n$ relations emerging from Eq. (5). Cable lengths and tensions may be subsequently (and straightforwardly) computed by Eq. (1) and a suitable set of linear independent relations chosen within Eq. (4). The set of *admissible* solutions consists of all those for which cable tensions are non-negative (cf. Eq. (3)) and the platform equilibrium is stable (cf. Section 3). The DGP is remarkably more complex, since in this case the platform configuration must be determined by simultaneously solving both the $6-n$ relations emerging from Eq. (5) and the n relations in Eq. (1).

It must be said that Eq. (1) represents a set of *theoretical* constraints, since the *actual* constraint imposed by a generic cable consists in that

$$|\mathbf{s}_i| = \sqrt{\mathbf{s}_i \cdot \mathbf{s}_i} \leq \rho_i \,. \tag{6}$$

The above refinement causes no concern when the IGP is dealt with, for in this case the *theoretical* values of cable lengths are conveniently computed by Eq. (1), after the platform posture has been established. Conversely, when the DGP is tackled, cable lengths are assigned as inputs, and *a priori* nothing ensures that *all* cables are called upon to sustain the load. Indeed, if a subset $\mathcal{W}$ of cable indexes exists such that $\text{card}(\mathcal{W}) < n$ and $\$_e \in \text{span}\{\$_j,\ j \in \mathcal{W}\}$, equilibrium configurations possibly exist such that $|\mathbf{s}_k| < \rho_k$, for all $k \notin \mathcal{W}$, and thus $\tau_k = 0$. These are legitimate solutions of the problem at hand. It follows that the overall solution set is obtained by solving the DGP for *all possible* constraint sets $\{|\mathbf{s}_j| = \rho_j,\ j \in \mathcal{W}\}$, with $\mathcal{W} \subseteq \{1 \ldots n\}$ and $\text{card}(\mathcal{W}) \leq n$, and by retaining, for each corresponding solution set, the solutions for which $|\mathbf{s}_k| < \rho_k,\ k \notin \mathcal{W}$. In general, this amounts to solving $\sum_{h=0}^{n-1} \binom{n}{n-h}$ DGPs.

A caveat is worth being mentioned. Equation (5) is only a *necessary* condition for equilibrium. In very special conditions, it may happen that equilibrium is not possible, in spite of Eq. (5) being fulfilled and irrespective of the sign of cable tensions. In particular, this occurs if $\mathbf{M}$ loses its full rank because a subset of its n first columns becomes linearly dependent, i.e. if the rank loss is 'concentrated' among the set of screws associated with the cable lines. In this case, the rank of the block[1] $\mathbf{M}_{1\ldots6,1\ldots n}$ is at most equal to $n-1$ and Eq. (2) may be satisfied only if $\text{rank}(\mathbf{M}) \leq n-1$. Cases like the one described here, however, are sufficiently unlikely to occur not to be, in practice, of particular concern. Nonetheless, a check of the rank of $\mathbf{M}_{1\ldots6,1\ldots n}$ is advisable before attempting to solve for cable tensions.

Throughout the text, the following notation is adopted (Fig. 1(a)). $Oxyz$ is a Cartesian coordinate frame fixed to the base, with $\mathbf{i}$, $\mathbf{j}$ and $\mathbf{k}$ being unit vectors along the coordinate axes. $Gx'y'z'$ is a Cartesian frame attached to the platform. $\mathbf{e}$ is a unit vector directed as $\$_e$, $\mathbf{x} = G - O$, $\mathbf{a}_i = A_i - O$, $\mathbf{r}_i = B_i - G$, $\mathbf{s}_i = B_i - A_i = \mathbf{x} + \mathbf{r}_i - \mathbf{a}_i$, $\mathbf{u}_i = (A_i - B_i)/\rho_i = -\mathbf{s}_i/\rho_i$ and $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, with $i, j = 1 \ldots n$, $i \neq j$. Without loss of generality, O is chosen to coincide with A_1 (so that $\mathbf{a}_1 = \mathbf{0}$) and $\mathbf{k} = \mathbf{e}$. If $\mathbf{b}_i$ is the projection of $B_i - G$ on $Gx'y'z'$, $\mathbf{\Phi}$ is the array grouping the variables parameterizing the platform orientation with respect to the fixed frame and $\mathbf{R}(\mathbf{\Phi})$ is the corresponding rotation matrix, then $\mathbf{r}_i = \mathbf{R}(\mathbf{\Phi})\,\mathbf{b}_i$. The platform posture is described by the array $\mathbf{X} = (\mathbf{x}; \mathbf{\Phi})$, with the components of $\mathbf{x}$ in $Oxyz$ being denoted, for the sake of brevity, as x, y and z. If O is chosen as the reduction pole of moments, $\$_i$ and $\$_e$ may be respectively expressed, in axis coordinates, as $\$_i = [\mathbf{s}_i;\ \mathbf{a}_i \times \mathbf{s}_i]$ and $\$_e = [\mathbf{e};\ \mathbf{x} \times \mathbf{e}]$. Accordingly, $\mathbf{M}$ becomes

$$\mathbf{M} = \begin{bmatrix} \mathbf{x}+\mathbf{r}_1 & \cdots & \mathbf{x}+\mathbf{r}_i-\mathbf{a}_i & \cdots & \mathbf{e} \\ \mathbf{0} & \cdots & \mathbf{a}_i \times (\mathbf{x}+\mathbf{r}_i) & \cdots & \mathbf{x} \times \mathbf{e} \end{bmatrix}, \tag{7}$$

or, equivalently, after subtracting the first column from the columns 2 through n,

$$\mathbf{M}' = \begin{bmatrix} \mathbf{x}+\mathbf{r}_1 & \cdots & \mathbf{r}_{i1}-\mathbf{a}_i & \cdots & \mathbf{e} \\ \mathbf{0} & \cdots & \mathbf{a}_i \times (\mathbf{x}+\mathbf{r}_i) & \cdots & \mathbf{x} \times \mathbf{e} \end{bmatrix}. \tag{8}$$

[1] The notation $\mathbf{M}_{hij,klm}$ denotes the block matrix obtained from rows h, i and j, and columns k, l and m of $\mathbf{M}$. When all columns of $\mathbf{M}$ are used, the corresponding subscripts are omitted.

3 The Stability of Equilibrium

Let an equilibrium configuration $(\bar{\mathbf{X}}, \bar{\rho}_1 \dots \bar{\rho}_m)$ be considered, with m being the number of *active* constraints (i.e. the number of cables contributing to supporting the platform). By a convenient reordering of indexes, taut cables may be assumed to be the first m, with $m \leq n$. Since the platform preserves $6-m$ dofs, it may displace under the effect of a change in the external force acting on it, while cable lengths remain unvaried (for the sake of simplicity, it is assumed that the number of cables in tension does not change because of the perturbation, which is reasonable, but not necessarily true). The problem of assessing equilibrium stability is, thus, in order. In particular, G may generally move within a closed region of $\mathbb{R}^3$ (in some cases, a surface or a curve). If g is the frontier of this region, the equilibrium is stable any time the potential energy U associated with the external force $Q\$_e$, namely $-Q\mathbf{e}\cdot\mathbf{x}$, is at a minimum on g. Loosely speaking, the platform is at rest in all points $\bar{G}$ of g in which the variety tangent to g is perpendicular to $\mathbf{e}$, with the equilibrium being stable if and only if a neighborhood $W_{\bar{G}}$ of $\bar{G}$ exists such that $(P-\bar{G})\cdot\mathbf{e} < 0$, for all $P \in (g \cap W_{\bar{G}})$. In such a condition, when the platform displaces under the effect of a perturbation, the original configuration is restored if the perturbation ceases. Figure 1(b) helps to depict this concept. The figure shows the locus g of the positions that G may assume for an exemplifying 22-CDPR, under the constraints (1) and with $m = n = 2$. If the platform is thought of as the coupler of a four-bar linkage whose grounded links are the cables (with assigned lengths), g is the coupler curve of G, namely a bicursal sextic. The stationary configurations of G are the points of g in which the tangent line is perpendicular to $\mathbf{e}$, with U being at a minimum in $\bar{G}_2$ and $\bar{G}_4$. These are the stable equilibrium poses (of course, since cable tensions must be negative in the configurations lying above the base, and positive otherwise, $\bar{G}_4$ is *de facto* the only feasible configuration for the example at hand). Finding the minima of a constrained function is a classical issue in optimization theory. An efficient algorithmic formalization is presented in the following.

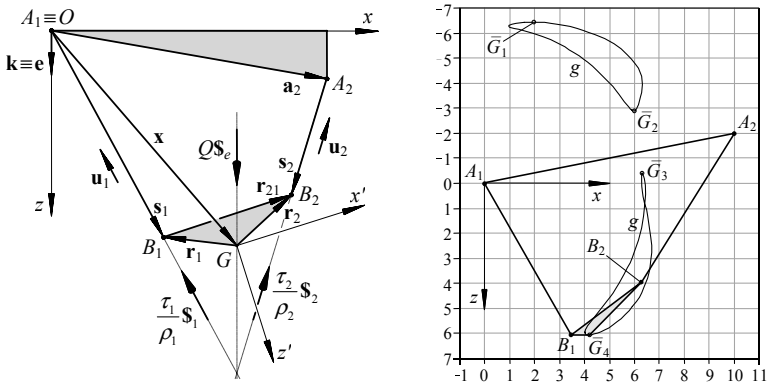
At equilibrium, the variation of the *global* potential energy of the platform due to a virtual displacement of it must be zero. Such a variation is the opposite of the work carried out by all forces acting on the platform, namely

$$\delta L = -\sum_{i=1}^{m} \tau_i \mathbf{u}_i \cdot \delta B_i - Q\mathbf{e}\cdot\delta G = 0. \tag{9}$$

If $\delta\mathbf{x}$ and $\delta\boldsymbol{\Theta}$ are, respectively, the virtual displacement of G and the virtual rotation of the platform, then $\delta G = \delta\mathbf{x}$ and $\delta B_i = \delta\mathbf{s}_i = \delta\mathbf{x} + \delta\boldsymbol{\Theta}\times\mathbf{r}_i$, so that

$$\delta L = -\left(\sum_{i=1}^{m}\tau_i\mathbf{u}_i + Q\mathbf{e}\right)\cdot\delta\mathbf{x} - \left(\sum_{i=1}^{m}\tau_i\mathbf{r}_i\times\mathbf{u}_i\right)\cdot\delta\boldsymbol{\Theta} = \mathbf{f}\cdot\delta\mathbf{x} + \mathbf{m}\cdot\delta\boldsymbol{\Theta} = 0. \tag{10}$$

Equation (10), from which $\mathbf{f}$ and $\mathbf{m}$ are inferred to be zero, is clearly equivalent to Eq. (2), by letting $n = m$. Since, for $\rho_i = \bar{\rho}_i$, $\delta\,(|\mathbf{s}_i| - \rho_i) = \delta|\mathbf{s}_i|$ and



(a) (b)

Fig. 1 22-CDPR: (a) general model; (b) equilibrium configurations for the case $\mathbf{a}_2 = (10, 0, -2)$, $\mathbf{b}_1 = (-0.5, 0, -0.5)$, $\mathbf{b}_2 = (3, 0, 0)$, and $\rho_1 = \rho_2 = 7$.

$$\delta|\mathbf{s}_i| = \frac{\mathbf{s}_i \cdot \delta\mathbf{s}_i}{\rho_i} = \frac{\mathbf{s}_i \cdot \delta\mathbf{x} + \mathbf{r}_i \times \mathbf{s}_i \cdot \delta\boldsymbol{\Theta}}{\rho_i} = -\left(\mathbf{u}_i \cdot \delta\mathbf{x} + \mathbf{r}_i \times \mathbf{u}_i \cdot \delta\boldsymbol{\Theta}\right), \tag{11}$$

δL may be written as

$$\delta L = -Q\mathbf{e} \cdot \delta\mathbf{x} + \sum_{i=1}^{m} \tau_i \delta\left(|\mathbf{s}_i| - \rho_i\right), \tag{12}$$

i.e. as the virtual variation of the Lagrange function

$$L = -Q\mathbf{e} \cdot \mathbf{x} + \sum_{i=1}^{m} \tau_i \left(|\mathbf{s}_i| - \rho_i\right), \tag{13}$$

with Lagrange multipliers coinciding with the cable tensions, namely, with the forces necessary to impose the geometrical constraints [7].[2] Such an observation is useful, since it allows the stability characteristics of the equilibrium to be assessed by evaluating the definiteness of the reduced Hessian $\mathbf{H}_r$ of L, i.e. the Hessian of L taken with respect to the configuration variables, further projected on the tangent space of the constraints $\mathcal{C}$ [7]. An algebraic expression of $\mathbf{H}_r$ is derived hereafter.

The second-order variation of δL is given by

[2] Equation (9) plus the relations $\{\tau_i > 0, |\mathbf{s}_i| = \rho_i\}$ for $i = 1 \ldots m$ and $\{\tau_i = 0, |\mathbf{s}_i| < \rho_i\}$ for $i = m \ldots n$ are equivalent to the Karush–Kuhn–Tucker conditions for the minimization of L under the constraints (6), provided that $\$_1, \ldots, \$_m$ are linearly independent.

$$\delta^2 L = -Q\mathbf{e} \cdot \delta^2 \mathbf{x} + \sum_{i=1}^{m} \tau_i \frac{\delta \mathbf{s}_i \cdot \delta \mathbf{s}_i}{\rho_i} + \sum_{i=1}^{m} \tau_i \frac{\mathbf{s}_i \cdot \delta^2 \mathbf{s}_i}{\rho_i}, \tag{14}$$

with $\delta^2 \mathbf{s}_i = \delta^2 \mathbf{x} + \delta^2 \boldsymbol{\Theta} \times \mathbf{r}_i + \delta \boldsymbol{\Theta} \times (\delta \boldsymbol{\Theta} \times \mathbf{r}_i)$. Enforcing $\mathbf{f} = \mathbf{m} = \mathbf{0}$ in Eq. (14) yields

$$\delta^2 L = \sum_{i=1}^{m} \frac{\tau_i}{\rho_i} \{\delta \mathbf{x} \cdot \delta \mathbf{x} - 2\delta \mathbf{x} \cdot (\mathbf{r}_i \times \delta \boldsymbol{\Theta}) - (\mathbf{r}_i \times \delta \boldsymbol{\Theta}) \cdot [(\mathbf{x} - \mathbf{a}_i) \times \delta \boldsymbol{\Theta}]\} \tag{15}$$

or, in matrix notation,

$$\delta^2 L = \sum_{i=1}^{m} \frac{\tau_i}{\rho_i} [\delta \mathbf{x}^T \delta \mathbf{x} - 2\delta \mathbf{x}^T \tilde{\mathbf{r}}_i \delta \boldsymbol{\Theta} + \delta \boldsymbol{\Theta}^T \tilde{\mathbf{r}}_i \left(\tilde{\mathbf{x}} - \tilde{\mathbf{a}}_i\right) \delta \boldsymbol{\Theta}], \tag{16}$$

where $\tilde{\mathbf{n}}$ denotes, for a generic vector $\mathbf{n}$, the skew-symmetric matrix expressing the operator $\mathbf{n}\times$. $\delta^2 L$ is a bilinear form in the twist space of the platform. If the platform virtual displacement is expressed, in ray coordinates, as $\delta \mathbf{t} = [\delta \mathbf{x}; \delta \boldsymbol{\Theta}]$, and $\mathbf{I}_3$ denotes the 3×3 identity matrix, the symmetric matrix associated with this form is

$$\mathbf{H}_p = \sum_{i=1}^{m} \frac{\tau_i}{\rho_i} \begin{bmatrix} \mathbf{I}_3 & -\tilde{\mathbf{r}}_i \\ \tilde{\mathbf{r}}_i & \frac{1}{2}\left(\tilde{\mathbf{r}}_i \tilde{\mathbf{x}} - \tilde{\mathbf{r}}_i \tilde{\mathbf{a}}_i + \tilde{\mathbf{x}} \tilde{\mathbf{r}}_i - \tilde{\mathbf{a}}_i \tilde{\mathbf{r}}_i\right) \end{bmatrix}, \tag{17}$$

which represents the pseudo-Hessian of L ($\mathbf{H}_p$ is not a true and proper Hessian, since $\delta \boldsymbol{\Theta}$ is not generally integrable).

The tangent space of $\mathcal{C}$ is obtained by setting Eq. (11) equal to zero for all values of i. In matrix notation, this amounts to

$$\mathbf{J}_p \delta \mathbf{t} = \begin{bmatrix} \mathbf{s}_1^T & (\mathbf{r}_1 \times \mathbf{s}_1)^T \\ \vdots & \vdots \\ \mathbf{s}_m^T & (\mathbf{r}_m \times \mathbf{s}_m)^T \end{bmatrix} \begin{bmatrix} \delta \mathbf{x} \\ \delta \boldsymbol{\Theta} \end{bmatrix} = \mathbf{0}, \tag{18}$$

where the ith row of $\mathbf{J}_p$ coincides with $\$_i$, expressed in axis coordinates and assuming G as the moment pole. $\mathbf{J}_p$ is the pseudo-Jacobian of the constraint equations.

If $\mathbf{N}_p$ is any $6 \times (6 - m)$ matrix whose columns generate the null space of $\mathbf{J}_p$, the reduced Hessian of $\mathcal{C}$ is the following $(6 - m) \times (6 - m)$ matrix:

$$\mathbf{H}_r = \mathbf{N}_p^T \mathbf{H}_p \mathbf{N}_p. \tag{19}$$

A sufficient condition for the equilibrium to be stable consists in $\mathbf{H}_r$ being positive definite.

If the method described above is applied to the example portrayed in Fig. 1(b), results that agree with those expected are obtained. The equilibrium configurations

are the *real* solutions of the DGP of the robot,[3] i.e. $\bar{G}_1 = (1.94, -6.43)$, $\bar{G}_2 = (5.98, -2.89)$, $\bar{G}_3 = (6.31, -0.42)$ and $\bar{G}_4 = (4.18, 6.07)$. Since the problem is planar, $\mathbf{H}_p$ and $\mathbf{J}_p$ are, respectively, 3×3 and 2×3 matrices, so that the reduced Hessian is a scalar. H_r is positive in $\bar{G}_2$ and $\bar{G}_4$ and negative in $\bar{G}_1$ and $\bar{G}_3$, namely

$$H_r|_{\bar{G}_1} = -29589, \;\; H_r|_{\bar{G}_2} = 18709, \;\; H_r|_{\bar{G}_3} = -22875, \;\; H_r|_{\bar{G}_4} = 61650. \quad (20)$$

If $\boldsymbol{\tau} = (\tau_1, \tau_2)$, corresponding cable tensions are

$$\boldsymbol{\tau}_{\bar{G}_1} = -(8.6, 2.3), \;\; \boldsymbol{\tau}_{\bar{G}_2} = -(24.2, 22.7), \;\; \boldsymbol{\tau}_{\bar{G}_3} = (22.1, 24.5), \;\; \boldsymbol{\tau}_{\bar{G}_4} = (6.0, 5.6). \quad (21)$$

4 Application Example: The 33-CDPR

Due to space limitations, only a brief outline of the IGP and the DGP of the 33-CDPR is sketched hereafter. Technical details and relevant discussions about empty and nonzero-dimensional solution sets will be provided in future papers.

When $n = 3$, Eq. (5) is satisfied and $\text{rank}(\mathbf{M}) = 3$ only if $\$_1$, $\$_2$, $\$_3$ and $\$_e$ belong to the same tridimensional subspace of lines [8]. Letting all 4×4 minors of $\mathbf{M}'$ vanish leads to 15 polynomial equations in $\mathbf{x}$ and $\boldsymbol{\Phi}$ of the form $p_j = 0$. We look for the variety V of the ideal generated by such equations. If three configuration variables are known (as in the IGP), any three p_j, say p_l, p_h, p_k, may be chosen and a corresponding (generally zero-dimensional) variety V_{lhk} is obtained. V is the intersection of the five varieties that may be generated this way. Clearly, a primary objective of the solving algorithm consists in limiting the number of varieties to be computed to the lowest possible value, possibly to just one.

It is useful to observe that letting $B_i \equiv A_i$ causes the ith column of $\mathbf{M}$ to vanish (since $\mathbf{s}_i = \mathbf{a}_i \times \mathbf{s}_i = \mathbf{0}$) and, hence, it causes all 4×4 minors of $\mathbf{M}$ (and thus of $\mathbf{M}'$) to be zero. It follows that a configuration for which $B_i \equiv A_i$ always belongs to V: we call it a *trivial* solution and we need to discard it (cf. Section 2). This observation is particularly important for the IGP with assigned orientation. In this case, in fact, it is always possible to displace the platform (with a given orientation) so as to superimpose B_i onto A_i. Consequently, all varieties V_{lhk} necessarily contain the trivial solutions corresponding to $B_i \equiv A_i$, namely $\bar{\mathbf{x}}_i = \mathbf{a}_i - \mathbf{r}_i$, $i = 1 \ldots 3$.

Inverse geometric problem. When the orientation is assigned, all vectors $\mathbf{r}_i$, $i = 1 \ldots 3$, are known. If the equations

$$p_1 := \det \mathbf{M}'_{1236} = 0, \quad p_2 := \det \mathbf{M}'_{1235} = 0, \quad p_3 := \det \mathbf{M}'_{1234} = 0 \quad (22)$$

are considered, it may be proven that $V \equiv V_{123}$. Such equations comprise the lowest-degree polynomials among all minors of $\mathbf{M}'$. In particular, p_1 is quadratic

[3] The analytical solution of the DGP of the general 22-CDPR will be reported in a future paper. As the class of a generic coupler curve is 12 [8], there are 12 lines tangent to it passing through the point at infinity perpendicular to **e**, so that the DGP admits up to 12 complex solutions [5].

in x and y, whereas p_2 and p_3 are quadratic in x, y and z. By eliminating z and y from Eq. (22), a 4th-degree polynomial equation in x may be obtained, i.e. $p_{123} = 0$. Since three roots of p_{123} necessarily correspond to trivial solutions, the fourth root is real and it may be computed by Vieta's formulas in closed form. The problem admits, thus, a single solution. Of course, it is admissible only if the corresponding cable tensions are nonnegative and the equilibrium is stable.

When the position $\mathbf{x}$ is assigned, $\mathbf{r}_1$, $\mathbf{r}_2$ and $\mathbf{r}_3$ are unknown. If the Rodrigues parameters e_1, e_2 and e_3 are chosen to describe the platform orientation, the relations in Eq. (22) assume a particularly favorable structure. Indeed, after letting $\mathbf{r}_i = \mathbf{R}\mathbf{b}_i$, $i = 1 \dots 3$, p_1, p_2 and p_3 become quartic polynomials in e_1, e_2 and e_3. The minors of $\mathbf{M}'$ other than $\mathbf{M}'_{1234}$, $\mathbf{M}'_{1235}$ and $\mathbf{M}'_{1236}$ yield, instead, sextic equations in the Rodrigues parameters. Another useful quartic is obtained as follows. By setting the minors $\mathbf{M}'_{j456}$, $j = 1 \dots 3$, equal to zero, one has that $(\mathbf{x} + \mathbf{r}_1) \det \mathbf{M}'_{456,234} = \mathbf{0}$. The variety defined by the above equation comprises the trivial solution $\mathbf{x} = -\mathbf{r}_1$ and the set of all configurations for which

$$p_8 := \det \mathbf{M}'_{456,234} = 0. \tag{23}$$

Equation (23) is, indeed, of degree four in e_1, e_2 and e_3 (and it is quadratic in x, y, z). It is well known that three polynomial equations of the same total degree always admit a Sylvester-type resultant free from extraneous polynomial factors [9]. For the case of three quartics, such a resultant is a univariate 64th-degree polynomial in one of the unknowns, say e_3. However, if an attempt is made compute the resultant of p_1, p_2 and p_3, it appears to be identically nought. Conversely, if any one among p_1, p_2 and p_3 is replaced by p_8, the expected 64th-degree polynomial is obtained. The problem admits, thus, at the most 64 solutions.

Direct geometric problem. When cable lengths are assigned, the platform configuration has to be determined. Equation (5) provides up to 15 (non-independent) polynomial equations in the platform posture variables. Among them, Eqs. (22) and (23) are of degree four in e_1, e_2 and e_3 and degree two in x, y, z.

Equation (1) provides three further relations in the platform posture variables. In particular, one may conveniently consider

$$q_1 := |\mathbf{s}_1|^2 - \rho_1^2 = 0, \tag{24a}$$

$$q_2 := |\mathbf{s}_2|^2 - \rho_2^2 - |\mathbf{s}_1|^2 + \rho_1^2 = 0, \tag{24b}$$

$$q_3 := |\mathbf{s}_3|^2 - \rho_3^2 - |\mathbf{s}_1|^2 + \rho_1^2 = 0, \tag{24c}$$

which, after clearing the denominator $1 + e_1^2 + e_2^2 + e_3^2$, are quadratic in e_1, e_2 and e_3. q_1 is also quadratic in the elements of $\mathbf{x}$, whereas q_2 and q_3 are linear in these variables.

The point-to-point distance relations in Eq. (24) represent the typical constraints governing the forward kinematics of parallel manipulators with telescoping legs connected to the base and the platform by ball-and-socket joints. In particular, the DGP of the general Gough–Stewart manipulator depends on six equations of this sort, one of which is equivalent to Eq. (24a) and five more to Eqs. (24b)-(24c).

This problem is known to be very difficult and it has attracted the interest of researchers for several years [10, 11]. The DGP of the 33-CDPR appears to be even more complex. In fact, three equations analogous to Eqs. (24b)–(24c), namely *linear* in the components of $\mathbf{x}$ and *quadratic* in the components of $\mathbf{\Phi}$, are replaced by relationships that are, at least, *quadratic* in the components of $\mathbf{x}$ and *quartic* in the components of $\mathbf{\Phi}$ (cf. Eqs. (22) and (23)). Possible simplifications may, indeed, arise from the fact that some power-products of x, y and z are actually missing in the equations emerging from Eq. (5). The redundancy of such equations may also play a role. The problem appears to be a daunting task and it has not been solved yet.

5 Conclusions

This paper studied the kinematics and statics of under-constrained cable-driven parallel robots with less than six cables, in crane configuration. In these robots, kinematics and statics are intrinsically coupled and they must be dealt with simultaneously. This poses major challenges.

A geometrico-static model was presented, together with an original and general procedure aimed at effectively solving the inverse and direct position problems in analytical form. A spatial robot with three cables was considered as a case study, in order to show the feasibility of the presented approach. It was shown that the position problems that arise gain remarkable complexity with respect to those of analogous rigid-link robots, such as the Gough–Stewart manipulator. The inverse analysis may lead up to 64 solutions, when the platform position is assigned and the its orientation and the cable lengths must be determined. The direct problem, with the platform posture being unknown and the cable lengths being given, appears to be a much more difficult task and it will be the subject of future research.

A purely algebraic method, based on a constrained optimization formulation, was provided for the assessment of equilibrium stability. The method proposed in [5] differs from the one presented here in that it determines the stability of equilibrium by looking at the Hessian of an *unconstrained* potential, explicitly expressed as a function of a number of independent coordinates equal to the number of tensioned cables. Such a mapping is, generally, very difficult to obtain (indeed, Michael et al. [5] apply important simplifications on the geometry of the robot) and extensive differential symbolic computation is needed. The advantage of the method described here consists in that it relies on a reduced Hessian of which a purely algebraic formulation is provided, and it may be very simply applied to the most general cases, with no need to perform any differentiation.

References

1. Ming, A., Higuchi, T.: Study on multiple degree of freedom positioning mechanisms using wires – Part 1. *Int. J. Jpn. S. Prec. Eng.*, **28**(2), 131–138 (1994).
2. Hiller, M., Fang, S., Mielczarek, S., Verhoeven, R., Franitza, D.: Design, analysis and realization of tendon-based parallel manipulators. *Mech. Mach. Theory*, **40**(4), 429–445 (2005).
3. Maier, T., Woernle, C.: Inverse kinematics for an underconstrained cable suspension manipulator. In: Lenarčič, J., Husty, M.L. (eds.), *Advances in Robot Kinematics: Analysis and Control*. Kluwer Academic Publishers, Dordrecht, pp. 97–104 (1998).
4. Yamamoto, M., Mohri, A.: Inverse kinematic analysis for incompletely restrained parallel wire mechanisms. In: *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Takamatsu, Japan, pp. 504–509 (2000).
5. Michael, N., Kim, S., Fink, J., Kumar, V.: Kinematics and statics of cooperative multi-robot aerial manipulation with cables. In: *ASME Int. Design Engineering Technical Conf.*, Paper No. DETC2009-87677, San Diego, USA (2009).
6. Bosscher, P., Ebert-Uphoff, I.: A stability measure for underconstrained cable-driven robots. In: *IEEE Int. Conf. on Robotics and Automation*, New Orleans, USA, pp. 4943–4949 (2004).
7. Pagani, C.D., Salsa, S.: *Analisi Matematica – Vol. 2*. Masson, Milan (1993) [in Italian].
8. Hunt, K.H.: *Kinematic Geometry of Mechanisms*. Clarendon Press, Oxford (1978).
9. Salmon, G.: *Lessons Introductory to the Modern Higher Algebra*. Hodges, Figgis, and Co., Dublin (1885).
10. Merlet, J.-P.: *Parallel Robots*. Springer, Dordrecht (2006).
11. Sommese, A.J., Wampler, C.W.: *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science*. World Scientific Publishing, Singapore (2005).

The Inverse Kinematics of 3-D Towing

Qimi Jiang and Vijay Kumar

University of Pennsylvania, USA; e-mail: {qimi, kumar}@seas.upenn.edu

Abstract. This work addresses the kinematics of 3-D towing where multiple aerial robots are used to cooperatively transport a payload using cables. In 3-D towing, it is important to determine the relative positions of the robots for the desired position and orientation of the payload. This is actually the inverse kinematics problem of the 3-D towing. In the case with three robots, when the tensions are given, the inverse kinematics problem has finite number of solutions. In order to obtain all possible solutions, an efficient analytic algorithm based on dialytic elimination is presented. An example is used for illustration.

Key words: Inverse kinematics, 3-D towing, static equilibrium, dialytic elimination.

1 Introduction

Aerial towed-cable-body systems have been widely used in emergency response, industrial, and military applications for object transport to environments inaccessible by other means [1–3]. For instance, in civil engineering, cranes or flying cranes are usually used for payload transport. In rescue missions, individuals are lifted by helicopters from dangerous situations, and in humanitarian missions, food and medical supplies have to be carried by helicopters to remote communities.

Because of so many potential applications, aerial towed-cable-body systems have attracted a lot of attention [1–4]. For instance, some researchers developed the control model for this kind of systems [2, 3]. However, for a typical towing with one single cable, the controllability of the payload is quite limited [4].

In order to improve the controllability, the manipulation and transportation with three aerial robots permitting six-dimensional pose control was studied in [5]. Also, the kinematics of such a system was analyzed in [5–7] based on screw theory and potential energy minimization. For the inverse kinematics problem, the feasible workspace was determined and represented using a normalized body-fixed coordinates. However, such an abstract representation is very difficult to visualize and inconvenient to use in practice. Also, the information about how the load is distributed among the cables was neglected.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_34,

This work addresses the inverse kinematics problem of 3-D towing with three robots: Given the desired position and orientation of the payload, what are the possible positions of the robots during 3-D towing? Section 2 formulates the static equilibrium condition. Section 3 formulates the inverse kinematics problem. Section 4 presents an efficient analytic algorithm based on dialytic elimination to solve the inverse kinematics problem. Section 5 uses an example to demonstrate the proposed algorithm. Finally, conclusions are drawn in Section 6.

2 The Static Equilibrium Condition

In order to formulate the inverse kinematics problem, the static equilibrium condition should be given first. Figure 1 shows the 3-D towing with n robots. Suppose that the position of robot Q_i in the reference frame is $\mathbf{q}_i = [x_{qi}, y_{qi}, z_{qi}]^T$. The positions of the attachment point P_i in the reference and body-fixed frames are respectively $\mathbf{p}_i = [x_{pi}, y_{pi}, z_{pi}]^T$ and $\tilde{\mathbf{p}}_i = [\tilde{x}_{pi}, \tilde{y}_{pi}, \tilde{z}_{pi}]^T$. The positions of the center of mass of the payload in the reference and body-fixed frames are respectively $\mathbf{r} = [x_c, y_c, z_c]^T$ and $\tilde{\mathbf{r}} = [\tilde{x}_c, \tilde{y}_c, \tilde{z}_c]^T$. The position and orientation of the payload can be represented by the position of the center of mass of the payload in the reference frame $\mathbf{r} = [x_c, y_c, z_c]^T$ and the rotation matrix $\mathbf{R}$ determined by three rotation angles ϕ, θ and ψ. Referring to Fig. 1, when $\mathbf{r}$ and $\mathbf{R}$ are given, the position of the attachment point P_i in the reference frame can be given as $\mathbf{p}_1 = \mathbf{r} - \mathbf{R}\tilde{\mathbf{r}}$, $\mathbf{p}_2 = \mathbf{p}_1 + \mathbf{R}\tilde{\mathbf{p}}_2$ and $\mathbf{p}_3 = \mathbf{p}_1 + \mathbf{R}\tilde{\mathbf{p}}_3$. If the length of cable i is given by l_i, the unit wrench $\mathbf{w}_i$ of cable i with respect to the origin O of the reference frame can be given as

$$\mathbf{w}_i = \frac{1}{l_i} \begin{bmatrix} \mathbf{q}_i - \mathbf{p}_i \\ \mathbf{p}_i \times \mathbf{q}_i \end{bmatrix}. \tag{1}$$

The wrench $\mathbf{G}$ caused by the weight of the payload with respect to the origin O is

$$\mathbf{G} = -mg \begin{bmatrix} \mathbf{e}_3 \\ \mathbf{r} \times \mathbf{e}_3 \end{bmatrix}, \tag{2}$$

where m is the mass of the payload and g is the acceleration due to gravity.

Hence, the static equilibrium condition can be given as

$$[\mathbf{w}_1 \ \ \mathbf{w}_2 \ \ \dots \ \ \mathbf{w}_n] \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} + \mathbf{G} = 0. \tag{3}$$

where λ_i $(i = 1, 2, \dots, n)$ is the tension of cable i.

Also, the geometric constraints are given as

$$\|\mathbf{q}_i - \mathbf{p}_i\| = l_i. \tag{4}$$

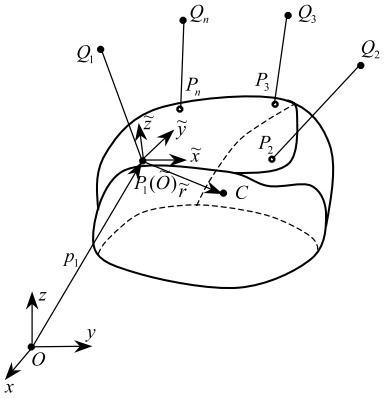


Fig. 1 3-D towing with multiple robots. Q_i represents the position of robot i and P_i represents the attachment point of cable i to the suspended object.

3 The Inverse Kinematics Problem

For the 3-D towing with three robots, the inverse kinematics problem can be formulated as follows: *Given the desired position and orientation of the payload (*$\mathbf{r},\mathbf{R}$*), find the positions of the robots (*$\mathbf{q}_i$*,* $i=1,2,3$*) that satisfy Eqs. (3) and (4).*

For a given set of tensions λ_i $(i=1,2,3)$, there are 9 unknowns $(x_{qi},y_{qi},z_{qi},i=1,2,3)$ in 9 equations given by Eqs. (3) and (4). Hence, finite number of solutions should be found for the inverse kinematics problem.

From Eq. (3), one gets

$$\begin{cases} s_1x_1+s_2x_2+s_3x_3=0, \\ s_1y_1+s_2y_2+s_3y_3=0, \\ s_1z_1+s_2z_2+s_3z_3=mg, \\ -s_6y_1+s_5z_1-s_9y_2+s_8z_2-s_{12}y_3+s_{11}z_3=t_1, \\ s_6x_1-s_4z_1+s_9x_2-s_7z_2+s_{12}x_3-s_{10}z_3=t_2, \\ -s_5x_1+s_4y_1-s_8x_2+s_7y_2-s_{11}x_3+s_{10}y_3=0. \end{cases} \tag{5}$$

From Eq. (4), one gets

$$\begin{cases} x_1^2+y_1^2+z_1^2=l_1^2, \\ x_2^2+y_2^2+z_2^2=l_2^2, \\ x_3^2+y_3^2+z_3^2=l_3^2. \end{cases} \tag{6}$$

where coefficients $s_1,s_2,\ldots,s_{12}$, t_1,t_2 are constants or functions of λ_i $(i=1,2,3)$. And $x_i=x_{qi}-x_{pi}$, $y_i=y_{qi}-y_{pi}$ and $z_i=z_{qi}-z_{pi}$ $(i=1,2,3)$. Hence, the problem becomes how to solve the 9 unknowns $(x_i,y_i,z_i,i=1,2,3)$ using the 9 equations given by Eqs. (5) and (6). When $x_i,y_i,z_i(i=1,2,3)$ are available, the position coordinates x_{qi}, y_{qi}, z_{qi} ($i=1,2,3$) of the robots can be obtained.

As the six equations in Eq. (5) are linear independent in z_1,y_2,z_2,x_3,y_3,z_3, one gets

$$\begin{cases} z_1 = T_1 x_1 + T_2 y_1 + T_3 x_2 + T_4, \\ y_2 = T_5 x_1 + T_6 y_1 + T_7 x_2, \\ z_2 = T_8 x_1 + T_9 y_1 + T_{10} x_2 + T_{11}, \\ x_3 = -(s_1 x_1 + s_2 x_2)/s_3, \\ y_3 = T_{12} x_1 + T_{13} y_1 + T_{14} x_2 \\ z_3 = T_{15} x_1 + T_{16} y_1 + T_{17} x_2 + T_{18}, \end{cases} \tag{7}$$

where coefficients $T_1, T_2, \ldots, T_{18}$ are constants or functions of λ_i $(i = 1,2,3)$. Then, substituting Eq. (7) into Eq. (6), one gets

$$\begin{cases} a_1 x_1^2 + b_1 y_1^2 + c_1 x_2^2 + d_1 x_1 y_1 + e_1 y_1 x_2 + f_1 x_2 x_1 + g_1 x_1 + h_1 y_1 + i_1 x_2 + j_1 = 0, \\ a_2 x_1^2 + b_2 y_1^2 + c_2 x_2^2 + d_2 x_1 y_1 + e_2 y_1 x_2 + f_2 x_2 x_1 + g_2 x_1 + h_2 y_1 + i_2 x_2 + j_2 = 0, \\ a_3 x_1^2 + b_3 y_1^2 + c_3 x_2^2 + d_3 x_1 y_1 + e_3 y_1 x_2 + f_3 x_2 x_1 + g_3 x_1 + h_3 y_1 + i_3 x_2 + j_3 = 0. \end{cases} \tag{8}$$

where coefficients $a_i, b_i, \ldots, j_i$ $(i = 1,2,3)$ are constants or functions of λ_i $(i = 1,2,3)$.

The polynomial system given by Eq. (8) has 3 quadratic equations. The total degree of this polynomial system is 8. According to Bezout's theorem, this system has at most 8 isolated solutions in the complex Euclidean space.

4 An Analytic Algorithm Based on Dialytic Elimination

In [8], Roth presented an dialytic elimination approach to eliminate two unknowns without increasing the power products and without increasing the degree of the system. Here, this approach is used to solve Eq. (8).

Suppose that x_2 is suppressed, Eq. (8) can be written as

$$a_i x_1^2 + b_i y_1^2 + d_i x_1 y_1 + k_i x_1 + u_i y_1 + v_i = 0 \qquad (i = 1,2,3). \tag{9}$$

where $k_i = f_i x_2 + g_i$, $u_i = e_i x_2 + h_i$ and $v_i = c_i x_2^2 + i_i x_2 + j_i$.

Rewriting Eq. (9) into the form with homogeneous coordinates by substituting $x_1 = X/T$, $y_1 = Y/T$ and then multiplying by T^2, one gets

$$a_i X^2 + b_i Y^2 + d_i XY + k_i XT + u_i YT + v_i T^2 = 0 \quad (i = 1,2,3). \tag{10}$$

The Jacobian matrix of Eq. (10) with respect to X, Y and T will be

$$\mathbf{J} = \begin{bmatrix} 2a_1 X + d_1 Y + k_1 T & 2b_1 Y + d_1 X + u_1 T & k_1 X + u_1 Y + 2v_1 T \\ 2a_2 X + d_2 Y + k_2 T & 2b_2 Y + d_2 X + u_2 T & k_2 X + u_2 Y + 2v_2 T \\ 2a_3 X + d_3 Y + k_3 T & 2b_3 Y + d_3 X + u_3 T & k_3 X + u_3 Y + 2v_3 T \end{bmatrix}. \tag{11}$$

The determinant of the above Jacobian matrix is

$$|\mathbf{J}| = AX^3 + BX^2 Y + CX^2 T + DXY^2 + EXT^2 + FXYT + GY^3 + HY^2 T + IYT^2 + KT^3, \tag{12}$$

where the coefficients $A, B, \ldots, K$ are functions of x_2.

Setting the derivatives of Eq. (12) with respect to the homogeneous coordinates to zero, three new equations can be obtained:

$$\begin{cases} \frac{\partial |\mathbf{J}|}{\partial X} = 3AX^2 + 2BXY + 2CXT + DY^2 + ET^2 + FYT = 0, \\ \frac{\partial |\mathbf{J}|}{\partial Y} = BX^2 + 2DXY + FXT + 3GY^2 + 2HYT + IT^2 = 0, \\ \frac{\partial |\mathbf{J}|}{\partial T} = CX^2 + 2EXT + FXY + HY^2 + 2IYT + 3JT^2 = 0. \end{cases} \tag{13}$$

The six equations in eqs.(10) and (13) can be written in the following matrix form:

$$\mathbf{MX} = 0, \tag{14}$$

where $\mathbf{X} = [X^2, Y^2, XY, XT, YT, T^2]^T$. To make $\mathbf{X}$ be not a zero vector, the determinant of matrix $\mathbf{M}$ should be zero. This determinant is an eight-degree polynomial in x_2. Hence, x_2 can be easily solved with Matlab. For every real root x_2 substituted into Eq. (14), a solution of x_1 and y_1 can be obtained by solving a linear system. When x_1, y_1 and x_2 are available, the other six unknowns (z_1, y_2, z_2, x_3, y_3, z_3) can be calculated with Eq. (7). Then, the position coordinates x_{qi}, y_{qi} and z_{qi} ($i = 1, 2, 3$) can be obtained.

5 An Example

In theory, the tension of every cable should be in a range from 0 to the payload capacity λ_{imax} of the robot, i.e., $\lambda_i \in [0, \lambda_{imax}]$. In the case with three robots, it seems that any point in the cuboid with side length λ_{imax} ($i = 1, 2, 3$) as shown in Fig. 2 is a work point. However, this is not true, because the tensions of the three cables should satisfy $\sum_{i=1}^{3} \lambda_i \geq mg$. This condition shows that the possible work point for the tensions should be in the region above the plane ABC with $\sum_{i=1}^{3} \lambda_i = mg$. However, even in this region, not any point can be used as a work point. For instance, any point on the boundary $FGHIJKF$ cannot be a work point if a desired orientation is needed.

The contribution ration of robot Q_i can be defined as $c_{ri} = \lambda_i / \lambda_{imax}$. During towing, all robots should cooperate. This means that every robot's contribution λ_i should be proportional to its payload capacity λ_{imax}. If the payload capacities λ_{imax} of the robots are the same, their contributions λ_i should be the same. If the payload capacities λ_{imax} of the robots are not the same, their contribution rations c_{ri} should be the same. Otherwise, the robots are not cooperative. To illustrate this point, suppose that the payload is $10N$ and the payload capacity λ_{imax} of every robot is $8N$. If the tension of cable 1 in the towing is $8N$, the tension of cable 2 is $3N$ and the tension of cable 3 is only $0.1N$, we cannot say that this is cooperative towing. Obviously, robots 2 and 3 do not cooperate in this case.

When $c_{r1} = c_{r2} = c_{r3} = c_r$, a corresponding line $\overline{DE}$ can be obtained which is inside the cuboid as shown in Fig. 2. D lies in the the plane ABC and represents

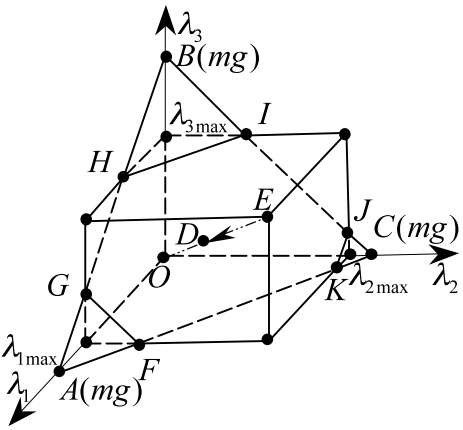
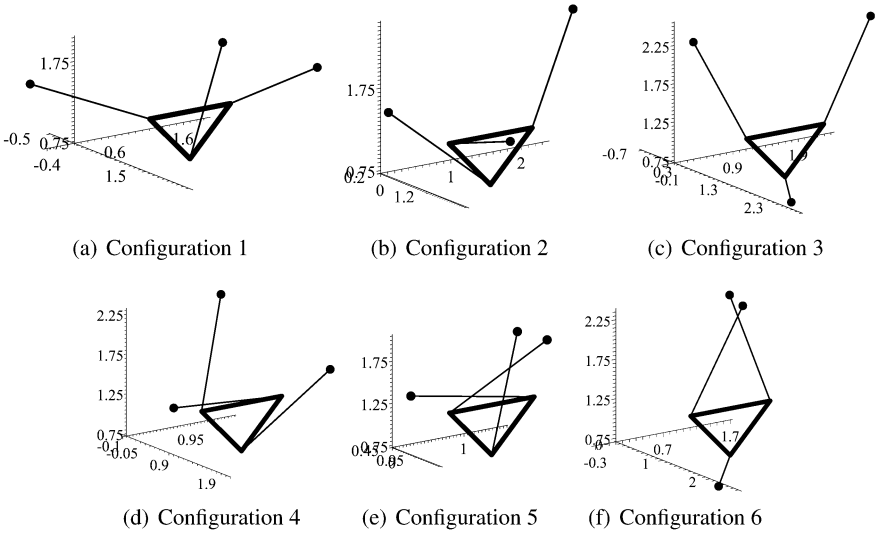


Fig. 2 Possible workspace of the tensions.



(a) Configuration 1 (b) Configuration 2 (c) Configuration 3

(d) Configuration 4 (e) Configuration 5 (f) Configuration 6

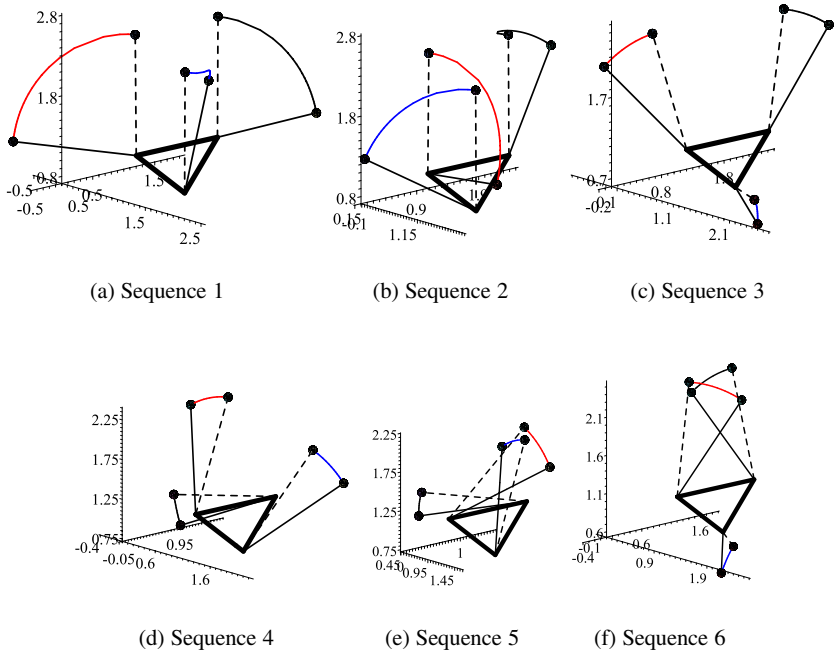
Fig. 3 The six configurations with $c_r = 0.8$ for the desired position and orientation of the payload.

the point with the minimal contribution ration, which should be the most interesting point. E represents the point with the maximal contribution ration, $c_r = 1$. When the contribution ration c_r is given, the tension of every cable will be available, i.e., $\lambda_i = c_r \lambda_{imax}$.

Suppose that the positions of P_1, P_2, P_3 and the center of mass of the payload in the body-fixed frame are $\tilde{\mathbf{p}}_1 = [0,0,0]^T$, $\tilde{\mathbf{p}}_2 = [1,0,0]^T$, $\tilde{\mathbf{p}}_3 = [0.5,\sqrt{3}/2,0]^T$ and $\tilde{\mathbf{r}} = [0.5,\sqrt{3}/6,0]^T$ respectively. The lengths of three cables are $l_1 = l_2 = l_3 = 1.5m$. The weight of the payload is $mg = 25N$. The payload capacities of three robots are $\lambda_{imax} = 20N$ $(i = 1,2,3)$. If the desired position and orientation of the payload

Table 1 The six solutions with $c_r = 0.8$ for the desired position and orientation of the payload.

No.	x_{q1}	y_{q1}	z_{q1}	x_{q2}	y_{q2}	z_{q2}	x_{q3}	y_{q3}	z_{q3}
1	-0.430	-0.347	1.424	1.045	1.447	1.996	2.385	1.900	1.924
2	1.907	0.646	1.399	0.277	0.052	1.466	0.816	2.302	2.479
3	-0.644	0.743	2.021	2.697	-0.091	0.849	0.946	2.348	2.473
4	-0.072	1.445	2.247	2.105	1.532	1.801	0.967	0.024	1.296
5	1.351	1.526	1.968	0.946	1.395	1.992	0.703	0.079	1.384
6	0.359	1.632	2.244	2.570	-0.271	0.787	0.071	1.638	2.312



(a) Sequence 1 (b) Sequence 2 (c) Sequence 3

(d) Sequence 4 (e) Sequence 5 (f) Sequence 6

Fig. 4 The six sequences of configurations for the desired position and orientation of the payload.

are given by $\mathbf{r} = [1,1,1]^T$ and $\phi = 25^\circ, \theta = 15^\circ, \psi = -5^\circ$, for a given contribution ration $c_r = 0.8$, six solutions for the inverse kinematics problem are obtained and listed in Table 1, which corresponds to six configurations as shown in Fig. 3.

Suppose that c_r varies from 1 which corresponds to point E in Fig. 2 and gradually decreases with a small step size, say $\Delta c_r = 0.05$, along with the line ED in Fig. 2, a sequence of every solution can be found. These sequences are shown in Fig. 4 in which sequences 1 and 2 correspond to a range of $c_r \in [0.417, 1]$ and sequences 3 to 6 correspond to a range of $c_r \in [0.675, 1]$. In other words, when the contribution ration c_r is less than 0.675, the inverse kinematics problem for this case has only two solutions. The minimal contribution ration $c_r = 0.417$ can be ob-

tained with a numerical iteration algorithm by reducing the step size at a satisfactory precision. When c_r reaches 0.417, the tension in every cable is exactly one-third of the weight of the payload. Also, the three cables lie in a vertical position as shown by the dashed lines in Figs. 4(a) and 4(b).

6 Conclusions

In 3-D towing with multiple aerial robots, it is important to determine the relative positions of the robots for a desired position and orientation of the payload. In order to address this inverse kinematics problem with three robots, an efficient analytic algorithm based on dialytic elimination is presented to obtain all possible solutions. If the chosen tensions are valid, several real solutions can be found. This point is demonstrated by an illustrated example.

Acknowledgements The authors gratefully acknowledge the support from NSF grants IIS-0413138, IIS-0427313 and IIP-0742304, ARO Grant W911NF-05-1-0219, ONR Grant N00014-08-1-0696, and ARL Grant W911NF-08-2-0004. The first author was supported by the PDF fellowship from the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. Henderson, J.F., Potjewyd, J. , and Ireland, B., 1999, The dynamics of an airborne towed target system with active control, *Proceedings of the Institution of Mechanical Engineers, Journal of Aerospace Engineering, Part G*, **213**, 305–319 (1999).
2. Sgarioto, D. and Trivailo, P., Cable assisted rendezvous for aircraft with surface locations. In *Proceedings of the 16th International Federation of Automatic Control World Congress*, Prague, July 4–8, pp. 1–6 (2005).
3. Williams, P., Sgarioto, D., and Trivailo, P., Optimal control of an aircraft-towed flexible cable system. In *Proceedings of the 11th Australian International Aerospace Congress*, Melbourne, Mar. 12–17, pp. 1–21 (2005).
4. Murray, R.M., Trajectory generation for a towed cable system using differential flatness. In *IFAC World Congress*, San Francisco, CA, July (1996).
5. Michael, N., Fink, J., and Kumar, V., Cooperative manipulation and transportation with aerial robots. In *Proceedings of Robotics: Science and Systems*, Seattle, Washington, USA, June 28 – July 1 (2009).
6. Fink, J., Michael, N., Kim, S., and Kumar, V., Planning and control for cooperative manipulation transportation with aerial robots. In *Proceedings of ISRR 2009 – 14th International Symposium on Robotics Research*, Lucerne, Switzerland, Aug. 31–Sept. 3 (2009).
7. Michael, N., Kim, S., Fink, J., and Kumar, V., Kinematics and statics of cooperative multi-robot aerial manipulation with cables. In *Proceedings of the ASME 2009 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Vol. DETC2009-87677, San Diego, USA, (2009).
8. Roth, B., Computations in kinematics. In: *Computational Kinematics*, J. Angeles, G. Hommel, and P. Kovacs (Eds.), Kluwer Academic Publishers, Dordrecht, pp. 3–14 (1993).

A Complete Method for Workspace Boundary Determination

O. Bohigas, L. Ros and M. Manubens

Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain; e-mail: {obohigas,llros,mmanuben}@iri.upc.edu

Abstract. This paper introduces a new method for workspace boundary determination on general lower-pair multi-body systems. The method uses a branch-and-prune technique to isolate the set of end effector singularities, and then classifies the points in such set according to whether they correspond to actual motion impediments in the workspace. The method can deal with open- or closed-chain systems, and is able to take joint limits into account. Advantages over other methods of similar applicability include its completeness and a simpler algorithmic structure. Examples are included that show its performance on benchmark problems documented in the literature.

Key words: Workspace determination, linear relaxation, closed-chain, multi-body system.

1 Introduction

A main problem of multi-body kinematics is *workspace determination*: for a multi-body system of known geometry, determine the complete set of poses (positions and orientations) that a selected body can adopt, as the system runs through all possible configurations. This special body is also known as the *end effector*, and the set of its poses as the *workspace* of the system. The issue has received substantial attention, as the availability of a good solution finds many applications to mechanism design, path planning, and task execution [1, 2].

Efficient solutions to this problem exist, but most of them are tailored to a particular robot architecture, or class of architectures. A large group of such methods adopt a constructive geometric approach to the problem. Representative of them is the work in [3], which computes the positional workspace of a spatial parallel manipulator, the work in [4], which extends the approach to deal with other physical constraints, and that in [5], which provides methods for various planar parallel manipulators. Other significant approaches include interval analysis techniques for Gough-type manipulators [6], optimization-based algorithms for fully serial/parallel robots [2], analytic methods for symmetrical spherical mechanisms [7], and topologic or algebraic-geometric studies for 3R manipulators [8, 9]. The literature on the

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_35, © Springer Science+Business Media B.V. 2010

topic is extensive, and we can only touch upon it briefly here. Elaborate surveys can be found in [1, 2].

While specific solutions are desirable, because they tend to yield faster algorithms, general solutions are required too, to analyze robots for which no specific method exists. In this paper, we argue that a solution for lower-pair mechanisms of general structure is possible, by extending a recent method for kinematic constraint solving for systems of such generality [10, 11].

Up to the authors' knowledge, only one approach of similar applicability is available in the literature, due to Haug and co-workers [12, and refs. therein]. Similarly to [12], we provide a visualization of the workspace by extracting its boundary from a set of end-effector singularities, but the strategy adopted here to formulate and compute such singularities is substantially different. In [12] the authors slice the singularity set into parallel curves, and a continuation scheme is then employed to trace all of such curves in detail. Although elegant and robust to bifurcations, such a procedure requires to be fed with at least one point for each connected component of the boundary, but no satisfactory method has been given to compute such points in general, as far as we know. In fact, the authors mention in [13] several situations that could make the technique miss some boundary segments. On the contrary, the method proposed in this paper is *complete*, in the sense that it is able to isolate all boundary segments of the workspace, including any interior barriers and voids that might be present. The method is based on formulating the equations of the singularity set in an appropriate form, and then exploiting this form to compute all singular points, using a numerical procedure based on linear relaxations [11].

2 Necessary Conditions

The allowable positions and orientations of all links in a multi-body system are usually encoded in an n_q-dimensional vector of generalized coordinates $\mathbf{q}$, subject to a system of $n_e \leq n_q$ equations of the form

$$\boldsymbol{\Phi}(\mathbf{q}) = 0, \tag{1}$$

that expresses the kinematic constraints imposed by the joints. Even if joint limits are present, these can be modelled as equality constraints, as shown in Section 5. Here $\boldsymbol{\Phi} : \mathcal{Q} \to \mathcal{E}$ is a smooth map, and $\mathcal{Q}$ and $\mathcal{E}$ are n_q- and n_e-dimensional manifolds, respectively. In the specific formulation that we will adopt, $\mathcal{Q} = \mathbb{R}^{n_q}$ and $\mathcal{E} = \mathbb{R}^{n_e}$, but they can be arbitrary manifolds in general. We focus on multi-body systems for which the solution set $\mathcal{C}$ of Eq. (1) is a smooth manifold of dimension $n_q - n_e$. This will be the case almost always, because the set of geometric parameters for which $\mathcal{C}$ fails to be a manifold has measure zero in the total space of such parameters. It is useful to consider the partition $\mathbf{q} = [\mathbf{v}^\mathsf{T}, \mathbf{w}^\mathsf{T}, \mathbf{u}^\mathsf{T}]^\mathsf{T}$, where $\mathbf{v} \in \mathcal{V}$ is a vector of n_v input variables corresponding to the actuated degrees of freedom of the multi-body system, $\mathbf{u} \in \mathcal{U}$ is a vector of n_u output variables encoding the pose of the end effector,

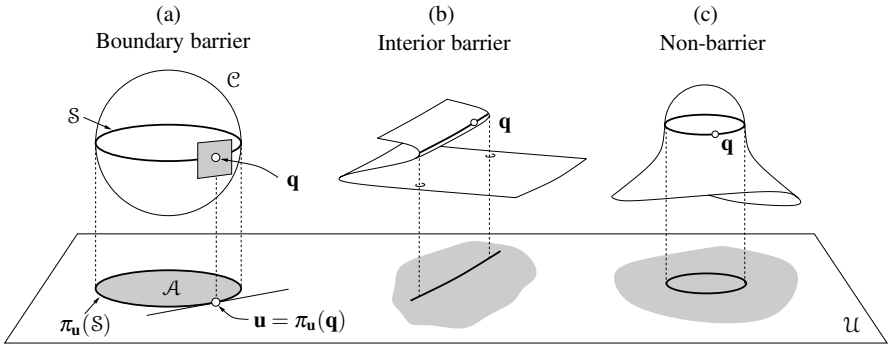


Fig. 1 (a) Sets $\mathcal{S}$ and $\pi_{\mathbf{u}}(\mathcal{S})$ when $\mathcal{C}$ is the sphere $x^2+y^2+z^2=1$ and $\pi_{\mathbf{u}}$ is the projection map $f(x,y,z)=(x,y)$. The "workspace" relative to the (x,y) variables is the projection of the sphere onto the (x,y) plane, and the boundaries of such projection correspond to points on the sphere where the tangent plane projects onto a line. (b) and (c): $\pi_{\mathbf{u}}(\mathcal{S})$ can also lie in the interior of $\mathcal{A}$.

and $\mathbf{w} \in \mathcal{W}$ is a n_w-dimensional vector encompassing the remaining intermediate variables. By defining $\mathbf{z} = [\mathbf{v}, \mathbf{w}]$, Eq. (1) can be written as $\boldsymbol{\Phi}(\mathbf{z}, \mathbf{u}) = 0$, and the workspace of the system can be defined as the set $\mathcal{A}$ of points $\mathbf{u} \in \mathcal{U}$ for which $\boldsymbol{\Phi}(\mathbf{z}, \mathbf{u}) = 0$ for some $\mathbf{z}$. In general it is easier to obtain a description of the workspace by computing its boundary, because such boundary is an object of lower dimension. A point $\mathbf{u} \in \mathcal{A}$ lies on the boundary of $\mathcal{A}$, denoted $\partial\mathcal{A}$, if every neighborhood of $\mathbf{u}$ intersects $\mathcal{A}$ and the complement of $\mathcal{A}$.

Let $\pi_{\mathbf{u}} : \mathcal{Q} \to \mathcal{U}$ denote the projection map from $\mathcal{Q}$ onto the $\mathbf{u}$ variables. That is, $\pi_{\mathbf{u}}(\mathbf{z}, \mathbf{u}) = \mathbf{u}$. Observe that $\mathcal{A}$ is exactly the image of $\mathcal{C}$ through $\pi_{\mathbf{u}}$. It is not difficult to see, moreover, that the points $\mathbf{q} \in \mathcal{C}$ that project onto some $\mathbf{u} \in \partial\mathcal{A}$ must necessarily be critical points of the projection of $\mathcal{C}$ onto $\mathcal{U}$, i.e., points $\mathbf{q} \in \mathcal{C}$ where the tangent space to $\mathcal{C}$ projects on $\mathcal{U}$ as a linear space of dimension lower than n_u. The set $\mathcal{S}$ of all critical points of the projection of $\mathcal{C}$ on $\mathcal{U}$ will be called the *singularity set* hereafter, and the notation $\pi_{\mathbf{u}}(\mathcal{S})$ will be used to refer to the projection of $\mathcal{S}$ onto $\mathcal{U}$. The situation is illustrated in Fig. 1a with an example.

Kinematically, $\mathcal{S}$ is the set of configurations in which the end effector loses instantaneous mobility [14, 15], which is the set of points $\mathbf{q} \in \mathcal{C}$ for which the matrix $d\boldsymbol{\Phi}_{\mathbf{z}} = \left[\partial \Phi_i / \partial z_j\right]$ is rank deficient. This allows a simple algebraic characterization of the points of $\mathcal{S}$. They are the points $\mathbf{q}$ that satisfy

$$\left.\begin{array}{c} \boldsymbol{\Phi}(\mathbf{z}, \mathbf{u}) = 0 \\ d\boldsymbol{\Phi}_{\mathbf{z}}{}^{\mathsf{T}} \xi = 0 \\ \xi^{\mathsf{T}} \xi = 1 \end{array}\right\}, \tag{2}$$

for some ξ, where ξ is an n_e-dimensional vector of unknowns. The first equation in (2) constrains the solutions to points on $\mathcal{C}$. The second and third equations impose the rank deficiency of $d\boldsymbol{\Phi}_{\mathbf{z}}$ (the rows of this matrix are dependent whenever they

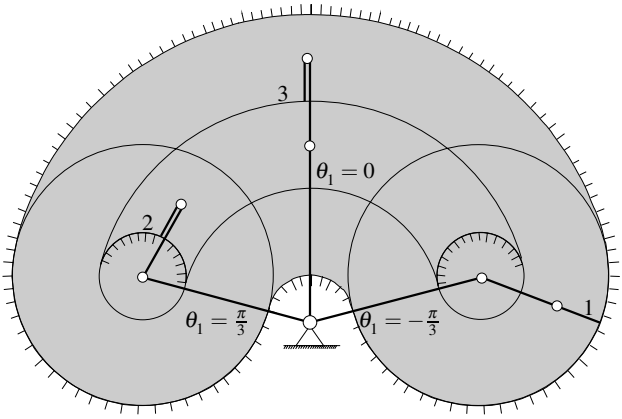


Fig. 2 Position workspace of a planar 3R manipulator, relative to the tip point of the last link, assuming that the angle θ_1 of the first revolute joint is restricted to the $[-\pi/3, \pi/3]$ range. Points corresponding to singularities are indicated in solid lines, and those relative to boundary and interior barriers are indicated with normal vectors on the forbidden side. Configurations 1, 2, and 3, are an example of a boundary barrier, an interior barrier, and a non-barrier singularity, respectively.

yield a vanishing linear combination with non-null coefficients). A preliminary idea of how the workspace boundary would look like, thus, can be gained by computing all points $\mathbf{q}$ that satisfy the previous system, and projecting them to the $\mathbf{u}$ variables, in order to obtain $\pi_{\mathbf{u}}(\mathcal{S})$.

3 Singularity Classification

Note that the criticality of $\mathbf{q}$ is a necessary but not sufficient condition for $\pi_{\mathbf{u}}(\mathbf{q})$ to lie in $\partial\mathcal{A}$, as there can be critical points projecting on the interior of $\mathcal{A}$ too. In fact, as illustrated in Fig. 1, points $\mathbf{q}$ satisfying Eqs. (2) can be classified into two broad categories. They can be *non-barrier* or *barrier* singularities, depending on whether there exists a trajectory in the neighborhood of $\mathbf{q}$ on $\mathcal{C}$, passing through $\mathbf{q}$, whose projection on $\mathcal{U}$ traverses $\pi_{\mathbf{u}}(\mathcal{S})$ or not, respectively. Points corresponding to barrier singularities, in turn, can be classified as *boundary* or *interior* singularities, according to whether they occur over $\partial\mathcal{A}$ or over the interior of $\mathcal{A}$, respectively. An example of each one of these singularity types is depicted in Fig. 2, for the particular case of a planar 3R manipulator. We next provide additional criteria to determine which of these singularity types occurs on a given $\mathbf{q}_0 \in \mathcal{S}$.

Let $\mathbf{q} = \mathbf{q}(\mathbf{v})$ be a parameterization of $\mathcal{C}$ in a neighborhood of $\mathbf{q}_0$, with $\mathbf{q}_0 = \mathbf{q}(\mathbf{v}_0)$. Let $\mathbf{n}_0$ be the normal to $\pi_{\mathbf{u}}(\mathcal{S})$ at $\mathbf{u}_0$, which can be computed as indicated in [12]. We can determine whether $\mathbf{q}_0$ corresponds to a boundary barrier by examining the sign of

$$\psi(\mathbf{v}) = {\mathbf{n}_0}^{\mathsf{T}}(\mathbf{u}(\mathbf{v}) - \mathbf{u}_0), \tag{3}$$

for all local trajectories $\mathbf{v} = \mathbf{v}(t)$ crossing $\mathbf{v}_0$ for some $t = t_0$ whose corresponding path $\mathbf{u} = \mathbf{u}(t)$ is orthogonal to $\pi_{\mathbf{u}}(\mathcal{S})$ at $\mathbf{u}_0$. This can be done by resorting to the second-order Taylor expansion of $\psi(\mathbf{v})$ around $\mathbf{v}_0$

$$\psi(\mathbf{v}) \simeq \psi(\mathbf{v_0}) + \delta\mathbf{v}^{\mathsf{T}}\psi_{\mathbf{v}}(\mathbf{v_0}) + \frac{1}{2}\delta\mathbf{v}^{\mathsf{T}}\psi_{\mathbf{vv}}(\mathbf{v_0})\delta\mathbf{v}, \tag{4}$$

where $\psi_{\mathbf{v}}$ and $\psi_{\mathbf{vv}}$ are the gradient and Hessian of $\psi(\mathbf{v})$, and $\delta\mathbf{v} = (\mathbf{v} - \mathbf{v}_0)$ is a small displacement whose corresponding $\delta u = (\mathbf{u} - \mathbf{u}_0)$ is orthogonal to $\pi_{\mathbf{u}}(\mathcal{S})$. Note here that the first term of this expansion vanishes because $\psi(\mathbf{v}_0) = \mathbf{n}_0{}^{\mathsf{T}}(\mathbf{u}_0 - \mathbf{u}_0) = 0$. Moreover, the time derivative of Eq. (3) for $\mathbf{v} = \mathbf{v}(t)$ is

$$\dot{\psi}(t) = \mathbf{n}_0{}^{\mathsf{T}}\dot{\mathbf{u}}(t), \tag{5}$$

which is the component of $\dot{\mathbf{u}}(t)$ along $\mathbf{n}_0$. As shown in [12], $\dot{\psi}(t_0)$ must vanish irrespectively of the chosen $\mathbf{v}(t)$. Thus, since for $t = t_0$ it is $\dot{\psi} = \psi_{\mathbf{v}}\dot{\mathbf{v}} = 0$ for all $\dot{\mathbf{v}}$, we conclude that $\psi_{\mathbf{v}}(\mathbf{v_0}) = 0$ too, meaning that the second term of the Taylor expansion also vanishes.

In conclusion, the sign of $\psi(\mathbf{v})$ is mostly determined by the definiteness properties of the quadratic form $\delta\mathbf{v}^{\mathsf{T}}\psi_{\mathbf{vv}}(\mathbf{v_0})\delta\mathbf{v}$. If this form is positive- or negative-definite, then all trajectories orthogonal to $\pi_{\mathbf{u}}(\mathcal{S})$ lie on one side of $\pi_{\mathbf{u}}(\mathcal{S})$ and $\mathbf{q}_0$ is a barrier singularity. If this form is indefinite, there are trajectories in $\mathcal{A}$ that cross $\pi_{\mathbf{u}}(\mathcal{S})$ and $\mathbf{q}_0$ is a non-barrier singularity. Lastly, if this form is semi-definite, we cannot deduce the singularity type unless we examine higher order terms of the Taylor expansion. The latter case may only occur on zero-measure subsets of $\mathcal{S}$, however. The definiteness test just outlined can easily be implemented by checking the eigenvalues of the matrix form of $\delta\mathbf{v}^{\mathsf{T}}\psi_{\mathbf{vv}}(\mathbf{v_0})\delta\mathbf{v}$ [12].

When $\mathbf{q}_0$ is classified as a barrier singularity, finally, it remains to determine whether $\mathbf{u}_0$ lies on $\partial\mathcal{A}$ or in the interior of $\mathcal{A}$. Note that a barrier singularity $\mathbf{q}_0$ will project in the interior of $\mathcal{A}$ if there is some $\mathbf{q} \notin \mathcal{S}$ that projects onto $\mathbf{u}_0$ in $\mathcal{U}$. This test can be implemented by checking whether equation $\Phi(\mathbf{z}, \mathbf{u}) = 0$ for $\mathbf{u}$ fixed to $\mathbf{u}_0$ has a solution $\mathbf{z}$ for which $d\Phi_{\mathbf{z}}$ is full rank.

4 Numerical Solution

We next show how Eq. (2) can be solved to determine $\mathcal{S}$, and how the classification scheme just given can be applied to selected points on $\mathcal{S}$, to obtain a detailed picture of the workspace.

As shown in [10, 11] it is always possible to write the first equation in Eq. (2) in a canonical form in which all component functions of $\Phi(\mathbf{z}, \mathbf{u})$ are quadratic polynomials. By quadratic we mean here that if q_i and q_j refer to any two of their variables, they involve monomials of linear, bilinear, or quadratic form only: q_i, $q_i q_j$, or q_i^2.

Note that if $\Phi(\mathbf{z}, \mathbf{u})$ has this special quadratic form, then all equations in System (2) will also have this form, because the entries in $d\Phi_{\mathbf{z}}$ will all be linear in their

variables. It is thus possible to use the numerical method developed in [10, 11] for systems of this kind, in order to isolate the solution set of Eqs. (2) to the desired accuracy. To make the paper self-contained, this method is reviewed briefly next.

Assuming that Eqs. (2) have the aforementioned form, the method starts by introducing the changes of variables $b_k = q_i q_j$ and $p_i = q_i^2$ for all bilinear and quadratic monomials, in order to transform the system into the expanded form

$$\left.\begin{array}{l} \mathbf{L}(\mathbf{x}) = 0 \\ \mathbf{B}(\mathbf{x}) = 0 \\ \mathbf{Q}(\mathbf{x}) = 0 \end{array}\right\}, \tag{6}$$

where $\mathbf{x}$ is a n_x-vector including the original $\mathbf{q}$ variables, and the newly-introduced p_i and b_k ones. Here, $\mathbf{L}(\mathbf{x}) = 0$ is a system of linear equations in $\mathbf{x}$, and $\mathbf{B}(\mathbf{x}) = 0$ and $\mathbf{Q}(\mathbf{x}) = 0$ are systems of bilinear and quadratic equations of the form $b_k - q_i q_j = 0$, $p_i - q_i^2 = 0$, respectively.

It is not difficult to see that, under the used formulation, each variable in $\mathbf{x}$ can only take values within a prescribed interval [11], so that from the cartesian product of all such intervals one can define an initial n_x-dimensional box $\mathcal{B}$ which bounds all solutions of Eqs. (2). The algorithm then isolates such solutions by recursively applying two operations on $\mathcal{B}$: *box shrinking* and *box splitting*.

Using box shrinking, portions of $\mathcal{B}$ containing no solution are eliminated by narrowing some of its defining intervals. This process is repeated until either (1) the box is reduced to an empty set, in which case it contains no solution, or (2) the box is "sufficiently" small, in which case it is considered a *solution* box, or (3) the box cannot be "significantly" reduced, in which case it is bisected into two sub-boxes via box splitting – which simply bisects its largest interval. To converge to all solutions, the whole process is recursively applied to the new sub-boxes, until one obtains a collection of solution boxes whose side lengths are below a given threshold σ.

The crucial operation in this scheme is box shrinking, which Porta et al. [11] implement as follows. Note first that the solutions falling in some sub-box $\mathcal{B}_c \subseteq \mathcal{B}$ must lie in the linear variety defined by $L(\mathbf{x}) = 0$. Thus, we may shrink $\mathcal{B}_c$ to the smallest possible box bounding this variety inside $\mathcal{B}_c$. The limits of the shrunk box along, say, dimension x_i can be found by solving the two linear programs

$$\begin{aligned} &\textbf{LP1: Minimize } x_i, \text{ subject to: } \mathbf{L}(\mathbf{x}) = 0, \mathbf{x} \in \mathcal{B}_c, \\ &\textbf{LP2: Maximize } x_i, \text{ subject to: } \mathbf{L}(\mathbf{x}) = 0, \mathbf{x} \in \mathcal{B}_c. \end{aligned}$$

However, note that the solutions must also lie on the parabolas $p_i = q_i^2$ of $\mathbf{Q}(\mathbf{x}) = 0$, and on the hyperbolic paraboloids $b_k = q_i q_j$ of $\mathbf{B}(\mathbf{x}) = 0$. The two facts can be taken into account by noting that the portion of the parabola $p_i = q_i^2$ lying inside $\mathcal{B}_c$ is bounded by two half planes, and that the points of $\mathcal{B}_c$ verifying $b_k = q_i q_j$ necessarily lie inside a tetrahedron defined by four points, obtained by clipping $\mathcal{B}_c$ with $b_k = q_i q_j$. Thus, the inequalities relative to such bounds can be added to **LP1** and **LP2** above, in order to take these constraints into account, which usually produces

a much larger reduction of $\mathcal{B}_c$, or even its complete elimination, if one of the linear programs is found unfeasible.

Upon termination, this algorithm will deliver a collection of n_s boxes containing all points in $\mathcal{S}$, forming a discrete envelope of this set whose accuracy can be adjusted through the σ parameter (next section illustrates such kind of output on a particular example). Finally, a point $\mathbf{q}_0 \in \mathcal{S}$ is computed for each one of the returned boxes, by solving Eq. (2) using a Newton method starting anywhere in the box, to be able to apply the classification method described in Section 3. Note that σ can always be chosen small enough so as to allow a rapid computation of $\mathbf{q}_0$.

5 A Comparative Example

The proposed technique has been implemented in C, using the libraries of the CUIK platform [11]. We next illustrate the performance of this implementation, on computing the position workspace of the mechanism in Fig. 3, i.e., the set of locations for point P, as the mechanism runs through all of its configurations. This mechanism is particularly useful to compare the results of the proposed approach with those of the continuation technique in [12], published in [16]. It was used in [2] too, as a means of verification.

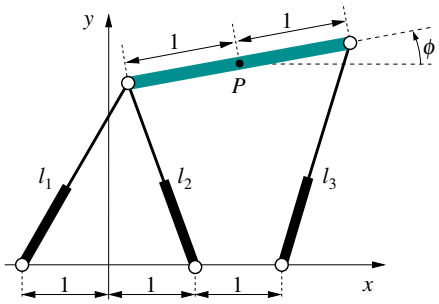


Fig. 3 A 3-RPR planar parallel mechanism.

The method in [12] starts shooting a ray through an initial point $\mathbf{u}_i \in \mathcal{A}$, corresponding to an assembled configuration of the mechanism, and tracks this ray using continuation, until a point $\mathbf{u}_b \in \partial\mathcal{A}$ is found (Fig. 4a). A second continuation process is then launched to track the connected component of $\pi_{\mathbf{u}}(\mathcal{S})$ that is reachable from $\mathbf{u}_b$ (Fig. 4b), whose points are finally classified into barrier and non-barrier singularities using the criteria of Section 3 (Fig. 4c). Note that this scheme is only able to detect the boundaries of the connected component of the workspace to which $\mathbf{u}_i$ belongs, while the algorithm we propose detects all components, as shown in Fig. 4, bottom row. To apply the presented approach, we first write Eq. (1) in the canonical form required in Section 4. For this, being $[x,y]^{\mathsf{T}}$ the coordinates of P, the slider variables l_i can be written as

$$\begin{aligned} l_1^2 &= y^2 - 2ys + s^2 + x^2 + 2x - 2xc - 2c + c^2 + 1, \\ l_2^2 &= y^2 - 2ys + s^2 + x^2 - 2x - 2xc + 2c + c^2 + 1, \\ l_3^2 &= y^2 + 2ys + s^2 + x^2 - 4x + 2xc - 4c + c^2 + 4, \end{aligned} \tag{7}$$

where c and s refer to the sine and cosine of ϕ, respectively, and thus must satisfy

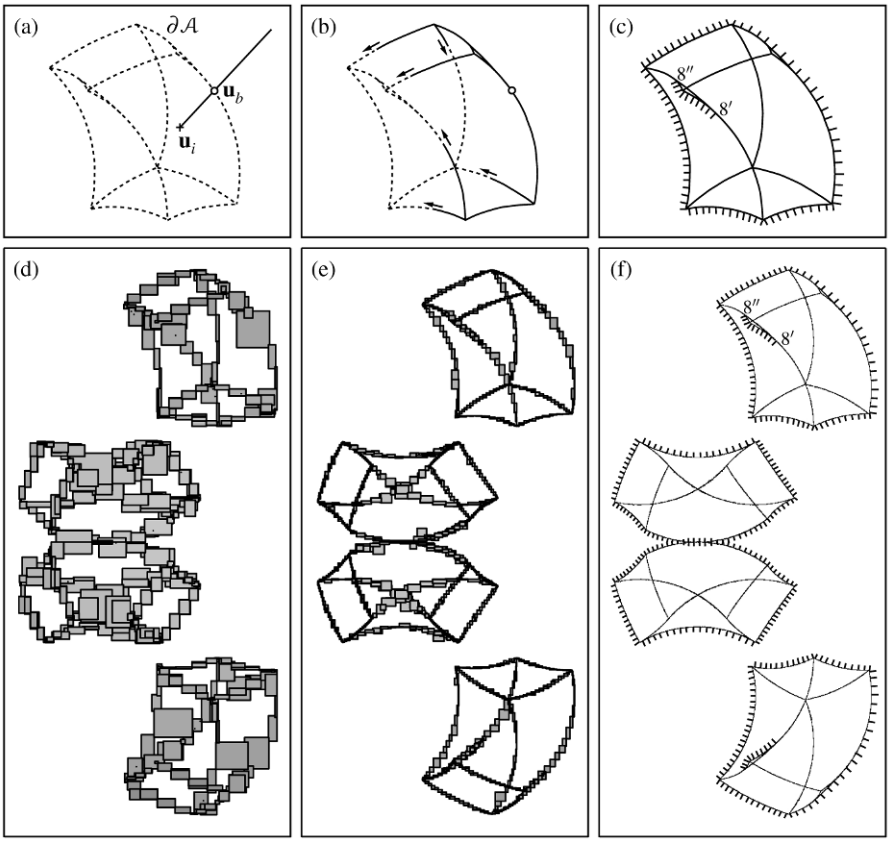


Fig. 4 Progress of the continuation algorithm in [12] (top row) compared to that of the proposed algorithm (bottom row) on computing the position workspace of the mechanism in Fig. 3. Note that the continuation algorithm is only able to isolate one connected component of the workspace, whereas the proposed one isolates them all. Figs. c and f follow the same convention as Fig. 2.

$$c^2 + s^2 = 1. \tag{8}$$

In this particular case, the slider variables l_i are only allowed to take values within prescribed ranges $[a_i, b_i]$, where $a_1 = a_2 = \sqrt{2}$, $a_3 = 1$, $b_1 = b_2 = 2$, and $b_3 = 3$. By defining $m_i = \frac{b_i + a_i}{2}$ and $h_i = \frac{b_i - a_i}{2}$, these constraints can be formulated using the slack variable technique of Optimization as

$$\begin{aligned} l_i &= m_i + h_i s_i, \\ c_i^2 + s_i^2 &= 1, \end{aligned} \tag{9}$$

for $i = 1, 2, 3$, which allows to integrate them readily into Eq. (7) as equalities. Thus, Eq. (1) is the system formed by equations (7), (8) and (9) in this case.

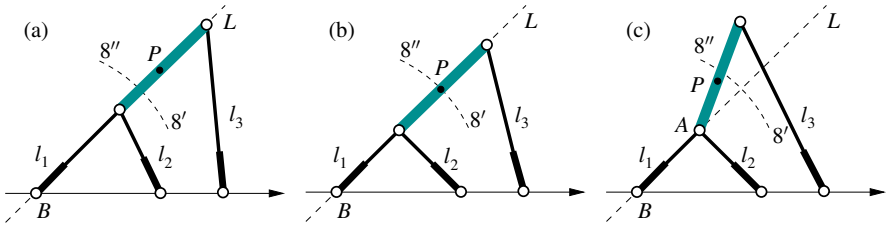


Fig. 5 A trajectory in which point P crosses the segment 8'-8".

The boundaries of the position workspace correspond to critical points of the projection of $\mathcal{C}$ onto the *xy* plane, i.e., to the solutions of Eq. (2) with $\mathbf{u} = [x, y]^{\mathsf{T}}$, and $\mathbf{z} = [c_1, c_2, c_3, c, s, s_1, s_2, s_3]^{\mathsf{T}}$. Equation (2), thus, constitutes a polynomial system of 19 equations in 20 variables in this example, with a one-dimensional solution set. The progress of the proposed algorithm on isolating this set is shown in Fig. 4, bottom row. Figures 4d and 4e show intermediate approximations of $\pi_{\mathbf{u}}(8)$ after 27 and 49 seconds, containing 1282 and 4846 boxes, respectively. Figure 4f displays the final result, which contains 152082 boxes (the boxes are too small to be appreciated). The overall computation was done using $\sigma = 0.01$, and it took 780 seconds on a parallelized version of the CUIK platform, on a grid of eight DELL Poweredge computers equipped with two Intel Quadcore Xeon E5310 processors and 4Gb of RAM each one. Figure 4f also shows the results of the classification process given in Section 3 applied to one point for each one of the returned boxes. It is worth mentioning that the segment $8'$-$8''$ was erroneously marked as an interior barrier in [16], while we detect it as a singularity of type non-barrier. This segment corresponds to point *P* tracing a circle around point *B*, when l_1 is fixed to its lowest value $\sqrt{2}$, while keeping the platform aligned with leg 1. The result in [16] must be erroneous, because *P* can really cross this segment from any of its two sides, as shown in Fig. 5. The platform can start from a position where *P* is to the right of the segment (Fig. 5a), then slide down along line *L* until it hits the segment (Fig. 5b), and, locking l_1 and l_2 to their values in this configuration, finally perform a rotation about point *A* by actuating l_3 (Fig. 5c).

6 Conclusions

This paper has introduced a new approach to compute workspace boundaries of general multi-body systems. A principal advantage of the method is its ability to converge to *all* boundary points, as discussed in the paper. Previous methods for the same purpose cannot ensure this property, since they are based on continuation, which requires the availability of one point for each connected component of the sought boundary, and no previous work on workspace analysis has shown how to compute all of such points in general, to the authors' knowledge.

The computation of an exhaustive representation of the workspace boundary becomes feasible when the workspace of the end effector is of dimension three or lower. However, for workspaces of larger dimension it turns out a difficult task, independently of the methodology used, as the curse of dimensionality must inevitably be faced. In order to visualize such workspaces, several authors introduce lower-dimensional representations of the workspace which are easier to compute and meaningful to the robot designer, like the *reachable* workspace, the *constant orientation* workspace, or the *constant position* workspace [6]. It is worth noting that all of these workspaces can be computed by the technique herein proposed, using a proper choice of the **u** variables, and fixing them to given values, if necessary.

Acknowledgements The authors thank Josep M. Porta for fruitful discussions on the topic of the paper. This work was funded by the Spanish Ministry of Science, under contract DPI2007-60858.

References

1. Merlet, J.-P., Gosselin, C.: Parallel mechanisms and robots. In: *Handbook of Robotics*. Springer-Verlag, pp. 269–285 (2008).
2. Snyman, J. A., du Plessis, L. J., Duffy, J.: An optimization approach to the determination of the boundaries of manipulator workspaces. *ASME J. of Mech. Des.* **122**(4), 447–456 (2000).
3. Gosselin, C.: Determination of the workspace of 6-DOF parallel manipulators. *ASME J. of Mech. Des.* **112**, 331–336 (1990).
4. Merlet, J. P.: Determination of the orientation workspace of parallel manipulators. *J. of Intelligent and Robotic Systems* **13**(2), 143–160 (1995).
5. Merlet, J. P., Gosselin, C. M. , Mouly, N.: Workspaces of planar parallel manipulators. *Mech. Mach. Theory* **33**(1-2), 7–20 (1998).
6. Merlet, J., et al.: Determination of 6D workspaces of Gough-type parallel manipulator and comparison between different geometries. *Int. J. of Robotics Research* **18**(9), 902–916 (1999).
7. Bonev, I. A., Gosselin, C. M.: Analytical determination of the workspace of symmetrical spherical parallel mechanisms. *IEEE Trans. on Robotics* **22**(5), 1011–1017 (2006).
8. Zein, M., Wenger, P., Chablat, D.: An exhaustive study of the workspace topologies of all 3R orthogonal manipulators with geometric simplifications. *Mech. Mach. Theory* **41**(8), 971–986 (2006).
9. Ottaviano, E., Husty, M., Ceccarelli, M.: Identification of the workspace boundary of a general 3-R manipulator. *ASME J. of Mech. Des.* **128**(1), 236–242 (2006).
10. Porta, J. M., Ros, L., Creemers, T., Thomas, F.: Box approximations of planar linkage configuration spaces. *ASME J. of Mech. Des.* **129**(4), 397–405 (2007).
11. Porta, J. M., Ros, L., Thomas, F.: A linear relaxation technique for the position analysis of multi-loop linkages. *IEEE Trans. on Robotics* **25**(2), 225–239 (2009).
12. Haug, E. J., Luh, C.-M., Adkins, F. A., Wang, J.-Y.: Numerical algorithms for mapping boundaries of manipulator workspaces. *ASME J. of Mech. Des.* **118**, 228–234 (1996).
13. Abdel-Malek, K., Adkins, F., Yeh, H. J., Haug, E.: On the determination of boundaries to manipulator workspaces. *Robotics and Comput. Integrated Manuf.* **13**(1), 63–72 (1997).
14. Zlatanov, D.: Generalized singularity analysis of mechanisms. Ph.D. dissertation, University of Toronto (1998).
15. Park, F. C., Kim, J. W.: Singularity analysis of closed kinematic chains. *ASME J. of Mech. Des.* **121**, 32–38 (1999).
16. Luh, C. M., Adkins, F. A., Haug, E. J., Qiu, C. C.: Working capability analysis of Stewart platforms. *ASME J. of Mech. Des.* **118**(2), 220–227 (1996).

PART 6

Inverse Kinematics of Humanoid-Robot Reaching through Human Visuo-Motor Learning

J. Babič[1], E. Oztop[2] and J. Lenarčič[1]

[1]*Jožef Stefan Institute, Slovenia; e-mail: {jan.babic, jadran.lenarcic}@ijs.si*
[2]*ATR Computational Neuroscience Laboratories, JST-ICORP Computational Brain Project, NICT Biological ICT Group, Japan; e-mail: erhan@atr.jp*

Abstract. To achieve human-like motion of the humanoid robots we propose an approach for robot skill-synthesis that exploits human visuo-motor learning capacity. The conceptual idea is to consider the target robot platform as a tool that can be intuitively controlled by a human. Once the robot is effortlessly controlled, the target behaviour of the robot is obtained by the human through practice. Subsequently we build the appropriate mapping between the human and the humanoid robot that is used for autonomous control of the humanoid robot. We demonstrate this idea by presenting humanoid-robot reaching obtained with the proposed approach.

Key words: Humanoid robot, reaching, visuo-motor learning, radial basis functions.

1 Introduction

If robots were able to imitate human motion demonstrated to them, acquiring complex robot motions and skills would become very straightforward. One can capture the desired motion of a human subject and map this motion to the kinematical structure of the robot. Due to the different dynamical properties of the humanoid robot and the human demonstrator, the success of this approach with regard to the stability of the humanoid robot depends on the ad-hoc mapping implemented by the researcher [1]. This approach can be considered as an open-loop control method and is illustrated in Fig. 1.

Majority of everyday human movements are not statically stable slow motions but are fast dynamically stable motions. To transfer such motions to the humanoid robot we have to consider, besides kinematics of the motion, some other important parameters that are crucial for balancing and dynamic stability of the robot. As it is usually the case, the researcher works out the details of the mapping so that the transferred action is stable on the robot. Here we propose a very different approach. Our approach uses the human demonstrator's real-time action to control the humanoid robot and to consecutively build an appropriate mapping between the human and the humanoid robot. This effectively creates a closed loop system where the human subject actively controls the humanoid robot motion in real time with the

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_36, © Springer Science+Business Media B.V. 2010

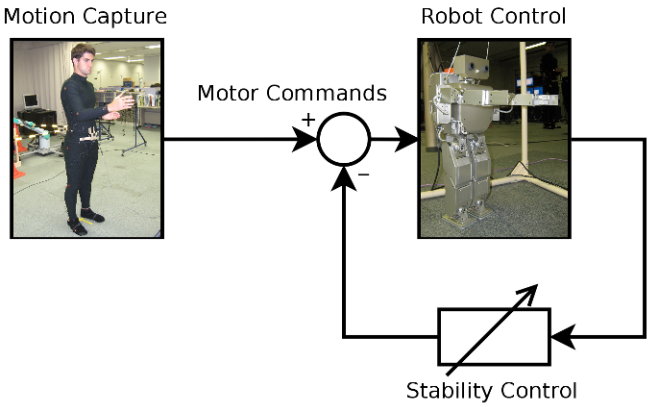


Fig. 1 Open-loop control of the humanoid robot. Motion of the human is transfered to the robot where a local controller takes care of the robot's postural stability.

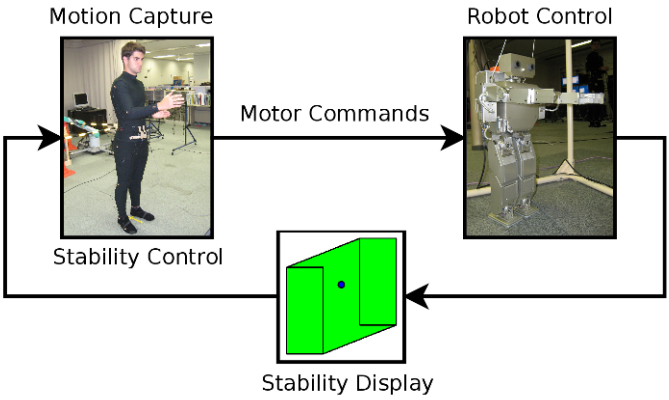


Fig. 2 Closed-loop control of the humanoid robot. Motion of the human is transfered to the robot while the robot's stability is presented to the human by a visual feedback.

requirement that the robot stays stable. This requirement can be easily satisfied by the human subject because of the human brain ability to control novel tools [2, 3]. The robot that is controlled by the demonstrator can be considered as a tool such as a car or a snowboard when one uses it for the first time. This setup requires the humanoid robot's state to be transferred to the human as the feedback information. We envision different types of visual feedback to provide the demonstrator either with the humanoid robot's view or with the view of the humanoid robot in the isometric projection. Closed-loop approach that was used in our study is illustrated in Fig. 2.

The purpose of our research is to exploit the human capability of learning to use novel tools in order to obtain a motor controller for reaching of the humanoid robot

without falling over. The construction of the motor controller has two phases. In the first phase a human demonstrator performs the reaching task on the humanoid robot via an intuitive interface. Subsequently in the second phase the obtained motions are acquired through machine learning to yield an independent motor controller.

2 Closed Loop Motion Transfer and Data Acquisition

The proposed approach can be considered as a closed loop approach where the human demonstrator is actively included in the main control loop as shown in Fig. 2. The motion of the human demonstrator was acquired by the contact-less motion capture system. By attaching an arbitrary number of optical markers to all segments of the demonstrator's body, the contact-less motion capture system together with the algorithms determines in real-time the states of the demonstrator's inner coordinate frames namely the angles in the demonstrator's joints. The role of the algorithms is to identify the kinematic model of the demonstrator's body and to calculate the joint angles of his body in accordance with the kinematical structure of the humanoid robot. The joint angles of the demonstrator were fed forward to the humanoid robot in real-time. In effect, the human acted as an adaptive component of the control system. During such control, a partial state of the robot needs to be fed back to the human subject. For statically balanced reaching skill, the feedback we used was the rendering of the position of the robot's centre of mass superimposed on the support polygon of the robot which was presented to the demonstrator by means of a graphical display. During the experiment the demonstrator did not see the humanoid robot.

The demonstrator's task was to keep the center of mass of the humanoid robot within the support polygon while performing the reaching movements with his right hand as directed by the experimenter. With a short practice session of about 15 minutes the demonstrator was able to move his body and limbs with the constraint that the robot's center of mass was within the support polygon. Hence the robot was statically stable when the demonstrator generated motions were either imitated by the robot in real-time or played back later on the robot. The robot used in the study was Fujitsu HOAP-II small humanoid robot.

Motion of the humanoid robot was constrained to the two dimensions; only the vertical axis and the axis normal to the trunk were considered. The curve in Fig. 3 shows the robot end-effector position data which was generated by the demonstrator. Together with the curve in Fig. 3 one can imagine the humanoid robot from its left side standing with the tips of the feet at the coordinates $(0, 0)$ and reaching out outwards with its right hand gliding over the curve. The long straight segment of the curve connects the beginning and the end of the reaching motion.

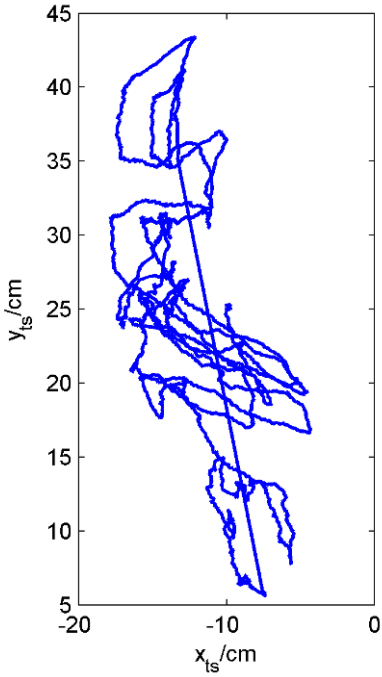


Fig. 3 The obtained trajectory of the humanoid robot's end-effector during the motion transfer.

3 Statically Stable Inverse Kinematics Determination

For each data point making up the curve in Fig. 3, the robot joint angles were recorded. Assuming m rows of the humanoid robot end-effector position $\mathbf{X}$ is formed by the data points taken from the curve in Fig. 3 and the robot joint angles $\mathbf{Q}$ is formed by the corresponding joint angles we get a non-linear relation of the form

$$\mathbf{Q} = \Gamma(\mathbf{X})\ \mathbf{W}. \tag{1}$$

By performing a non-linear data fit and solving for $\mathbf{W}$ we can afterwards make prediction with

$$\boldsymbol{q}_{pred} = \Gamma(\boldsymbol{x}_{des})\ \mathbf{W} \tag{2}$$

where $\boldsymbol{q}_{pred}$ is a vector of the predicted joint angles and $\boldsymbol{x}_{des}$ is a vector of the desired end-effector position. Using the prediction we can afterwards ask the humanoid robot to reach out for a desired position without falling over.

For non-linear data fitting the recorded positions $\mathbf{X}$ are mapped into an N dimensional space using the Gaussian basis functions given by

$$\varphi_i(\boldsymbol{x}) = e^{\frac{\boldsymbol{x}-\mu_i}{\sigma^2}} \tag{3}$$

where μ_i and σ^2 are open parameters to be determined. Each row of $\mathbf{X}$ in converted into an N dimensional vector forming a data matrix

$$\mathbf{Z} = \Gamma(\mathbf{X}) = \begin{bmatrix} \varphi_1(\boldsymbol{x}_1) & \varphi_2(\boldsymbol{x}_1) & \dots & \varphi_N(\boldsymbol{x}_1) \\ \varphi_1(\boldsymbol{x}_2) & \varphi_2(\boldsymbol{x}_2) & \dots & \varphi_N(\boldsymbol{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_1(\boldsymbol{x}_m) & \varphi_2(\boldsymbol{x}_m) & \dots & \varphi_N(\boldsymbol{x}_m) \end{bmatrix}. \tag{4}$$

Assuming we have a linear relation between the rows of $\mathbf{Z}$ and $\mathbf{Q}$, we can solve Eq. (2) for $\mathbf{W}$ in the sense of the minimum least squares by

$$\mathbf{W} = \mathbf{X}^+ \mathbf{Q} \tag{5}$$

where $\mathbf{X}^+$ represents the pseudo-inverse of $\mathbf{X}$. The residual error is given by

$$tr\left((\mathbf{XW} - \mathbf{Q})\ (\mathbf{XW} - \mathbf{Q})^T\right). \tag{6}$$

In effect, this establishes a non-linear data fit; given a desired end-effector position $\boldsymbol{x}$, the joint angles that would achieve this position are given by

$$\boldsymbol{q}_{pred} = \left(\varphi_1(\boldsymbol{x}_{des})\ \varphi_2(\boldsymbol{x}_{des})\ \dots\ \varphi_N(\boldsymbol{x}_{des})\right)\mathbf{W}. \tag{7}$$

The open parameters are N as the number of basis functions which implicitly determines $\boldsymbol{\mu}_i$ and the variance σ^2. They were determined using cross-validation. We prepared a Cartesian desired trajectory that was not a part of the recording data set and converted it into a joint trajectory with the current set values of (N, σ^2). The joint trajectory was simulated on a kinematical model of the humanoid robot producing an end-effector trajectory. The deviation of the resultant trajectory from the desired trajectory was used as a measure to choose the values of the open parameters. The values of N and σ for the experiments reported here were 20 and 60, consecutively.

The results we obtained by using Gaussian basis functions to form the non-linear mapping between $\mathbf{Q}$ and $\mathbf{X}$ are very satisfactory. Figure 4 shows the desired end-effector trajectory and the generated end-effector trajectory obtained by playing back the predicted joint angle trajectories on the humanoid robot. The light wiggly curve in Fig. 4 represents the end-effector trajectory that was generated by the human demonstrator in the first phase and subsequently used to determine the mapping $\mathbf{W}$ between the joint angles and the end-effector position.

The desired trajectory was generated by the parametric curve

$$\boldsymbol{x}(t_i) = \begin{bmatrix} cos\alpha & -sin\alpha \\ sin\alpha & cos\alpha \end{bmatrix} \begin{bmatrix} 3\,sin(t_i) - 9 \\ 15\,cos(t_i) + 26 \end{bmatrix}, \tag{8}$$

where $t_i = 0, 1/20, 1/10, \dots, 2\,\pi$ and $\alpha = 10\,\pi/180$. For the robot implementation the desired trajectory $\boldsymbol{x}(t_i)$ was mapped to $\boldsymbol{q}_{pred}(t_i)$ using Eq. (2). Afterwards a

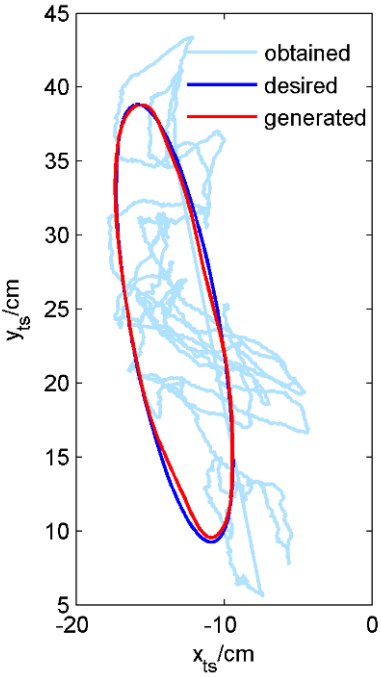


Fig. 4 The obtained end-effector trajectory generated by the demonstrator (light wiggly curve) with the desired end-effector trajectory that was used as the input for the joint angle prediction and the generated end-effector trajectory obtained by playing back the predicted joint angle trajectories on the humanoid robot.

smooth continuous $\boldsymbol{q}(t)$ function was obtained by cubic spline interpolation of the data points $(t_i, \boldsymbol{q}_{pred})$.

4 Statically Stable Motion Generation of the Humanoid Robot

The reaching skill of the humanoid robot we obtained was statically stable which means that the robot's centre of mass was inside the robot's support polygon. However, when the robot was asked to track a trajectory at speeds significantly higher than the speed of the demonstrator, the dynamics played a non-negligible effect. This can be seen from Fig. 5 where the upper plot shows the stability when the circular trajectory tracking was performed at $1/10$ Hz. When the motion was performed at twice speed, the robot became unstable as shown in the lower panel of Fig. 5. The robot could still track the desired trajectory without falling over, but just barely.

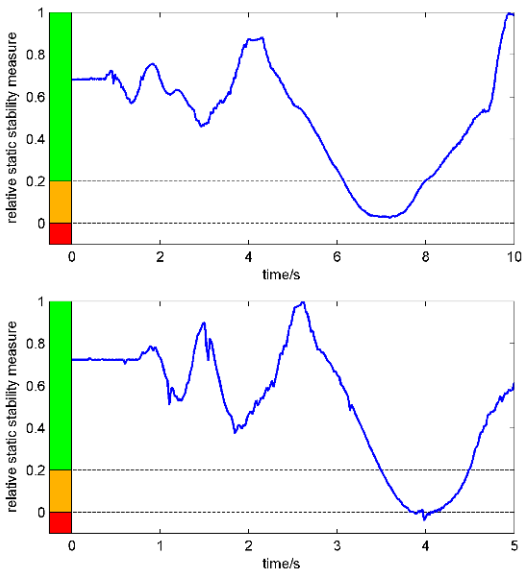


Fig. 5 The stability of the humanoid robot when the circular trajectory shown in Fig. 4 is played back at different speeds. When the robot moved fast, the dynamics effects are no longer negligible as underlined by the ditch below 0 at around 4 seconds in the lower plot.

5 Conclusions

A goal of imitation of motion from demonstration is to remove the burden of robot programming from the experts by letting non-experts to teach robots. The most basic method to transfer a certain motion from a demonstrator to a robot would be to directly copy the motor commands of the demonstrator to the robot [4]. This approach proved to be very efficient for certain open-loop tasks. However, this simple approach is generally not possible to implement. Either the motor commands may not be available to the robot or the differences between the demonstrator's body and the robot are so big that a direct transfer of motor commands is not possible. One way of solving this problem is to modify the motor commands produced by the demonstrator with a sort of a local controller or by on-line learning [5]. The situation in our approach is different because the correct motor commands for the robot are produced by the human demonstrator. For this convenience, the price one has to pay is the necessity of training to control the robot to achieve the desired action. Basically, instead of expert robot programming our method relies on human visuo-motor learning ability to produce the appropriate motor commands on the robot, which can be played back later or used to obtain controllers through machine learning methods as in our case of reaching.

The main result of our study is the establishment of the methods to synthesize the robot motion using human visuo-motor learning. To demonstrate the effectiveness of

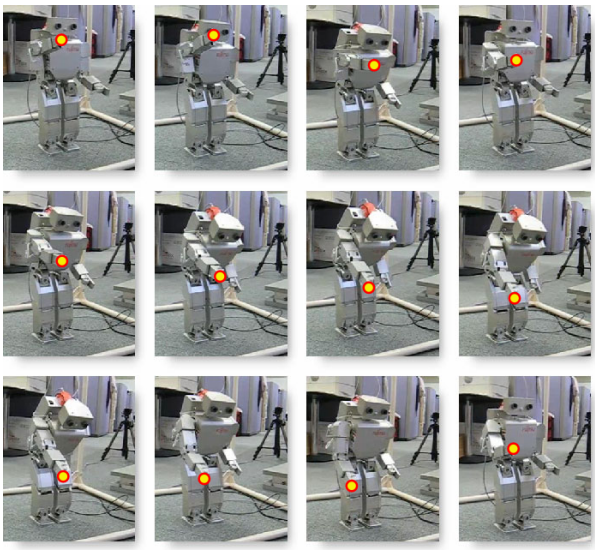


Fig. 6 Video frames representing the statically stable reaching motion of the humanoid robot obtained with the proposed approach.

the proposed approach, statically stable reaching was implemented on a humanoid robot using the introduced paradigm. A sequence of video frames representing the statically stable autonomous trajectory tracking obtained with our method is shown in Fig. 6.

Acknowledgements The research work reported here was made possible by Japanese Society for Promotion of Science and Slovenian Ministry of Higher Education, Science and Technology.

References

1. Schaal, S.: Is imitation learning the route to humanoid robots? *Trends Cogn Sci.* **3**, 233–242 (1999).
2. Oztop, E., Lin, L.-H., Kawato, M., Cheng, G.: Dexterous skills transfer by extending human body schema to a robotic hand. In *IEEE-RAS International Conference on Humanoid Robots*, Genova, Italy (2006).
3. Goldenberg, G., Hagmann, S.: Tool use and mechanical problem solving in apraxia. *Neuropsychology* **36**, 581–589 (1998).
4. Atkeson, C.G., Hale, J.G., Pollick, F.E., Riley, M., Kotosaka, S., Schaal, S., Shibata, T., Tevatia, G., Ude, A., Vijayakumar, S., Kawato, M.: Using humanoid robots to study human behavior. *IEEE Intelligent Systems* **15**, 45–56 (2000).
5. Gams, A., Ijspert, A.J., Schaal, S., Lenarčič, J.: On-line learning and modulation of periodic movements with nonlinear dynamical systems. *Autonomous Robots* **27**, 3–23 (2009).

Automated Fitting of an Elastokinematic Surrogate Mechanism for Forearm Motion from MRI Measurements

J. Xu[1], P. Kasten[2], A. Weinberg[3] and A. Kecskeméthy[1]

[1]*Chair for Mechanics and Robotics, University of Duisburg-Essen, Germany;*
e-mail: {jing.xu, andres.kecskemethy}@uni-due.de
[2]*Department of Orthopaedic Surgery, University of Dresden, Germany;*
e-mail: Philip.Kasten@uniklinikum-dresden.de
[3]*Department for Pediatric Surgery, University of Graz, Austria;*
e-mail: annelie.weinberg@t-online.de

Abstract. Forearm rotation (pro-supination) involves a non-trivial combination of rotation and translation of two bones, namely, radius and ulna, relatively to each other. Early works regarded this relative motion as a rotation about a fixed (skew) axis. However, this assumption turns out not to be exact. This paper regards a spatial-loop surrogate mechanism involving two degrees of freedom with an elastic coupling for better forearm motion prediction. The model parameters are not measured directly from the anatomical components, but are fitted by reducing the errors between predicted and measured values in an optimization loop. For non-invasive measurement of bone position, magnetic resonance imaging (MRI) imaging is employed. We present a method to self-calibrate the arm position in the MRI scanning tube and fitting the model parameters from a few, coarse MRI scans. Results show a good concordance between measurement and simulation.

Key words: Forearm kinematics, surrogate mechanism, MRI motion fitting.

1 Introduction

Performance of forearm pro- and supination motion (i.e., forearm rotation from palm down to palm up and vice versa), is one of the crucial functional motions of the upper extremity, which is used in everyday tasks such as turning a door handle or fixing a screw with a screw-driver. This motion is of particular interest in practice due to an increasing number of surgery treatments of the forearm. According to statistical data, for children, 20% of all recorded fractures occur in the forearm, making this the most frequent form of extremity fractures [1]. Therefore, there is a need for a clear description of forearm motion, especially under variable physiologic loading conditions and at different rotational positions of the forearm.

Forearm motion research began in the early 20th century. Fick in 1904 presented the first kinematic model describing forearm motion, which stated that the forearm rotated around the ulna around a constant axis of rotation [2] (see Fig. 1a). However, such a simplified model renders an unrealistic strong tilting in the maximal pronation. Based on MRI findings, more comprehensive kinematic models were developed that include the motion of the ulna. Kapandji and Nakamura et al. stated

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_37, © Springer Science+Business Media B.V. 2010

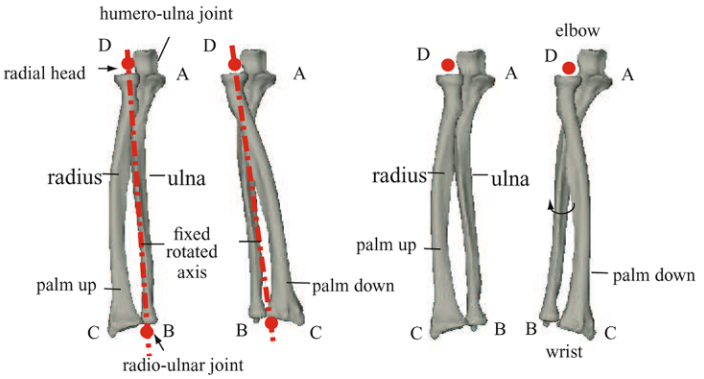
that rotational motion of the forearm does not occur around a constant axis, but that a complex motion comprising rotation and translation of radius relative to ulna takes place [3–5]. Robbin showed in a study that the axis of rotation is a variable screw axis [6]. This means that the ulna has an evasive motion of forearm rotation. Kasten et al. have proved that in a cadaveric study a lateral swaying rotation of ulna with respect to humerus takes place [7]. They introduced a surrogate mechanism for the pro- and supination, where the joints were taken into account by a simplified mechanical analogy [8]. Using this model, it was possible to predict rotational impairments based on angular deformities.

Kecskeméthy and Weinberg later extended this basic model by introducing a two-degree-of-freedom surrogate mechanism including virtual springs to incorporate elastic components, for instance ligaments [9]. This model features an elastokinematic coupling between axial displacement and lateral swaying of the humero-ulna (elbow) articulation and behaves, due to the strong elastic coupling, with respect to external loads as a mechanism with virtually a single degree of freedom. Recent publications have verified the lateral swaying motion predicted by this model [10], and are using it for external surrogate mechanisms design [11].

This paper is concerned with the problem of determining automatically the model parameters of the surrogate mechanism as well as the relative location of the bone geometry with respect to the mechanism links by use of MRI measurements at a few configurations of the forearm. In Section 2, the details of the proposed surrogate mechanism are summarized for better reference. Section 3 describes the proposed fitting procedure based on MRI slices at different rotational configurations. Finally, Section 4 presents some results of the method for a sequence of measurements made with and without external load. The results show a good agreement between measurement and simulation, suggesting that the model is suitable for forearm rotation prediction.

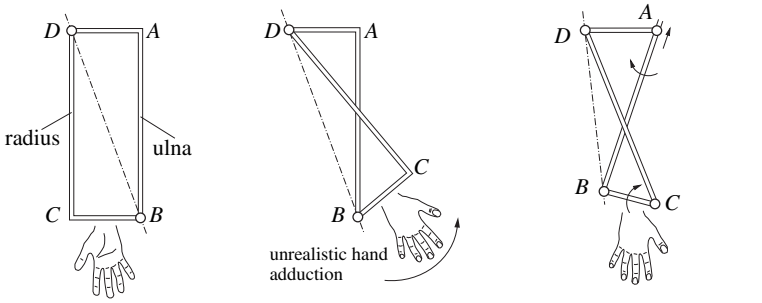
2 Forearm Elastokinematic Surrogate Mechanism

The forearm bony structure resembles a parallel mechanism containing two bones, the radius and ulna, which are coupled at their two ends by the elbow (proximal side) and the wrist (distal side). The rotation of the hand about the longitudinal axis brings the hand from supination (palm up) to pronation (palm down) and vice versa. Ulna and radius are coupled at the proximal end to the humerus (upper arm) at the elbow via the radial head (a spherical pit rotation on a spherical condyle of the humerus) and the humero-ulnar connection resembling a rotational axis. At the distal end, ulna and radius are connected by a radio-ulnar joint containing a sliding and rotating relative motion. All bones are tightly fixed together at the joints by ligaments, and ulna and radius are additionally bonded together under stress by the membrana interossea. Figure 1 shows the basic difference between early and recent forearm rotation models. In the early models, it was assumed that ulna remains fixed with respect to the elbow and that radius rotates about the axis interconnecting the



(a) Fixed-axis model (b) 2-DOF model

Fig. 1 Simple and extended model of forearm pose at supination and pronation (right hand).



(a) supination (b) tilting of hand in pronation (c) compensation of tilting

Fig. 2 Deficiencies of the fixed-axis model.

radial head and the radio-ulnar joint [2]. However, newer findings show that the ulna performs a lateral swaying rotation with respect to the elbow (lateral motion of point B in Fig. 1b), which means that the humero-ulnar (elbow) joint is not a perfect revolute joint but allows some tilting and dislocation.

The main effects are shown in Fig. 2. In the fixed-axis model, the hand would be tilted unrealistically to the inside in the pronation position. In order to compensate this tilting, a small penetration and a small lateral swaying rotation of ulna at the elbow joint is required (Fig. 2c). This requires that an additional revolute joint and an additional prismatic joint are placed between elbow and ulna at point A, and an additional revolute joint is placed between radius and wrist at point C. In such a way, a parallel mechanism arises which has two degrees of freedom.

Figure 3 shows a corresponding surrogate mechanism that allows one to model the effects described above [9]. The mechanism is assumed to operate at a fixed flexural position of the elbow, thus the proper elbow joint is not shown. The surrogate mechanism consists of a closed kinematic chain with four joints and two degrees

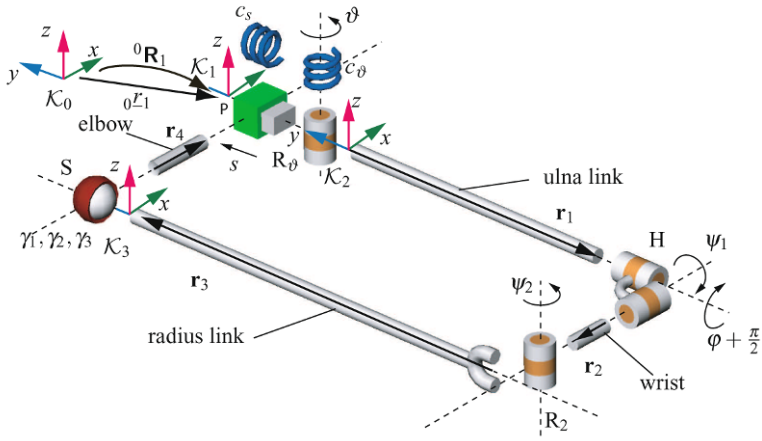


Fig. 3 Two degrees of freedom surrogate mechanism for pro- and supination.

of freedom. The prismatic joint P and the revolute R on the proximal end of ulna, represent a small dislocation s and and a small lateral swaying ϑ of the ulna with respect to the humerus. The Hooke joint H represents the distal radio-ulnar joint, consisting of the actual pro-supination angle φ and the torsional angle ψ_1. The joint R_2 describes the aperture ψ_2 between ulna and radius, and the spherical joint S represents the radial head at the proximal end of radius. The vectors $\underline{r}_1$, $\underline{r}_2$, $\underline{r}_3$, $\underline{r}_4$ represent the corresponding link lengths. At the humero-ulnar joints, two virtual springs with corresponding spring stiffnesses c_s, c_ϑ, respectively, are placed which summarize all elastic effects at these joints, including ligaments, capsules, etc.

As global home system, the inertial frame $\mathcal{K}_0$ is introduced. In this system, MRI data is collected. For the forearm description, we introduce an elbow frame $\mathcal{K}_1$ which may move with respect to the inertial frame from measurement to measurement with corresponding radius vector ${}_0\underline{r}_1$ and relative rotation matrix ${}^0\mathbf{R}_1$. The frames $\mathcal{K}_2$ and $\mathcal{K}_3$ are bone-fixed and are placed at the origin of the corresponding links. However, bone geometry and bone axes (not shown in Fig. 3) will be offset with respect to these links (see Section 3).

The kinematics of the skeleton model can be described by the loop closure condition:

$$\begin{aligned}&\mathtt{Rot}\,[x,\psi_1]\circ\mathtt{Rot}\,[z,\psi_2]\circ\mathtt{Trans}\,[r_3]\circ\mathtt{Rot}\,[y,\gamma_1]\circ\mathtt{Rot}\,[z,\gamma_2]\circ\mathtt{Rot}\,[x,\gamma_3]\\&=\mathtt{Trans}\,[-r_2]\circ\mathtt{Rot}\,[y,-(\varphi+\pi/2)]\circ\mathtt{Trans}\,[-r_1]\circ\mathtt{Rot}\,[z,-\vartheta]\\&\quad\circ\mathtt{Trans}\,[y,-s]\circ\mathtt{Trans}\,[-r_4]\,.\end{aligned}\tag{1}$$

Assuming all vectors to be coplanar at the supination position, the translation vectors of the model can be arranged as follows:

$$\underline{r}_1 = \begin{bmatrix} 0 \\ -r_1 \\ 0 \end{bmatrix}; \quad \underline{r}_2 = \begin{bmatrix} -r_2 \\ 0 \\ 0 \end{bmatrix}; \quad \underline{r}_3 = \begin{bmatrix} 0 \\ r_3 \\ 0 \end{bmatrix}; \quad \underline{r}_4 = \begin{bmatrix} r_4 \\ 0 \\ 0 \end{bmatrix}. \tag{2}$$

Due to the particular geometry of the loop, the variables can be determined in closed-form as functions of the two degrees of freedom s and φ as follows [9]:

$$\vartheta(s,\varphi) = \vartheta^* \pm \cos^{-1}(C/\overline{C}), \text{ where } \begin{cases} A = 2\,(r_2\, r_4 \sin\varphi - r_1\, s) \\ B = 2\,(r_2\, s \sin\varphi - r_1\, r_4) \\ C = r_1^2 + r_2^2 + r_4^2 + s^2 - r_3^2 \\ \overline{C} = \sqrt{A^2 + B^2}, \ \tan\vartheta^* = \frac{B}{A} \end{cases} \tag{3}$$

$$\sin\psi_2 = -\frac{r_2 + \sin\varphi\,(r_4\cos\vartheta + s\sin\vartheta)}{r_3}\,, \tag{4}$$

$$\tan\psi_1 = -\frac{\cos\varphi\,(r_4\cos\vartheta + s\sin\vartheta)}{r_1 + r_4\sin\vartheta - s\cos\vartheta}\,. \tag{5}$$

Static equilibrium involves two conditions (corresponding to the two degrees of freedom), for example

$$c_\varphi\,(\varphi - \varphi^*) = J_\varphi\, c_\vartheta\, \vartheta\,, \tag{6}$$

$$c_s\, s = J_s\, c_\vartheta\, \vartheta\,, \tag{7}$$

where c_ϑ and c_s are the virtual stiffnesses of the two input joints, and J_s and J_φ denote the Jacobian coefficients arising from the kinematical transmission

$$\dot{\vartheta} = [J_s\,,\, J_\varphi]\,[\dot{s}\,,\, \dot{\varphi}]^{\mathrm{T}}\,. \tag{8}$$

Using Eq. (6), one can "operate" the loop by setting an arbitrary pro-supination stiffness coefficient c_φ and "sliding" the spring offset φ^* to user-prescribed values. The loop will then move to its new equilibrium position and for the user it seems that she is operating a single input.

3 Model Parameter Fitting from MRI Measurements

The geometric and elastic parameters of the surrogate mechanism can not be determined directly from the image data, but need to be fitted so that the computed positions coincide with the measurements. The measurements consist in scanning bone cross sections of the forearm at different pro-supination angles $\varphi_1, \ldots, \varphi_N$. For each pro-supination angle, a reduced set of MRI slices $S_1, \ldots, S_M$ is scanned in order to insure a small measurement time. In our case, we chose $M = 5$ so that scanning time for one pro-supination angle is approximately 10 sec. This ensures that all slices for one position correspond roughly to the same forearm position.

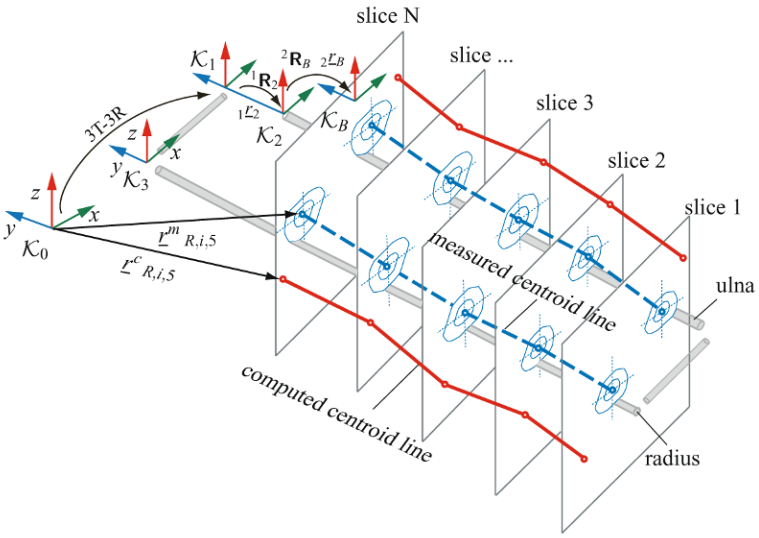


Fig. 4 Computed and measured bone marrow centroid line.

For model fitting, we regard as characteristic curves associated to the bones the centroids of the bone marrow cross sections. Let $\underline{r}^m_{i,j}$ denote the radius vector of the measured centroid at the j-th slice of the i-th measurement (corresponding to a new pro-supination angle), and let $\underline{r}^c_{i,j}$ denote the corresponding computed radius vector to the same target point. Both vectors are assumed to be decomposed in $\mathcal{K}_0$. An additional index U or R shall be included for denoting the corresponding slices for the ulna and radius bones, respectively. The minimization target function is

$$f = \sum_{i=1}^{N} \sum_{j=1}^{M} \left\{ || \underline{r}^m_{U,i,j} - \underline{r}^c_{U,i,j} ||^2 + || \underline{r}^m_{R,i,j} - \underline{r}^c_{R,i,j} ||^2 \right\}. \tag{9}$$

In the minimization, four sets of design variables arise: (1) the mechanism proper parameters $r_1, r_2, r_3, r_4, c_s, c_\vartheta$; (2) the rigid-body motion of the elbow, for each measurement, represented by three translations x_0^j, y_0^j, z_0^j and three rotations $\chi_1^j, \chi_2^j, \chi_3^j$; (3) the relative location of bone geometry frame $\mathcal{K}_B$ with respect to the link reference frame, involving three constant translations $\kappa_1, \kappa_2, \kappa_3$ and three constant rotations $\varepsilon_1, \varepsilon_2, \varepsilon_3$ for each bone; and (4) the pro-supination angle offset $(\varphi^*)^j$, which rotates the mechanism to the optimal configuration. All design variables must be determined simultaneously, as they are highly coupled, totalling $18 + 7N$ design variables for N measurements.

In order to allow for the forearm to change position from measurement to measurement, a self-calibrating procedure for determining the material points of the forearm bones at a set of scan slices is proposed. Prior to the motion scan, a fine MRI scan with numerous slices is performed as reference scan. Such a reference scan is

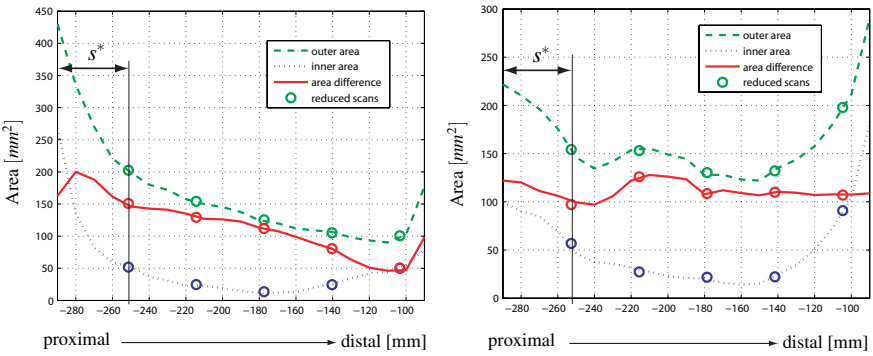


Fig. 5 Cross-sectional area for ulna and radius (supination angle $\varphi = 90°$).

needed anyway for smooth bone geometry rendering in subsequent applications of the simulation package in clinical practice and thus does not represent an additional effort. From the reference scan, one can obtain, using automatic segmenting, a quite smooth distribution of the bone marrow centroid lines and the cross sectional areas along the MRI axial direction. An example of corresponding measured profiles for cross sectional areas is shown in Fig. 5. As the change of cross sections is rather smooth and significant, it is easy to fit in longitudinal direction a reduced set of scans into this curve, yielding the axial offset s^* to the first slice of the reduced set of scans with respect to the bone origin. This defines the material point on which each slice of the coarse scan cuts the bone (shown in Fig. 5 as circles), yielding quite an accurate matching of cross sectional areas (Fig. 5). At this point, the tilting of the forearm with respect to the MRI axial direction was not regarded. However, this will be further analyzed in future papers.

Once the material points are identified, the bone-fixed coordinates of the bone marrow centers can be determined by linear interpolation of the reference bone marrow centroid line of the reference scan, yielding corresponding bone-fixed vectors ${}_B^{B}\underline{\xi}_{i,j}$. The corresponding computed bone marrow centers then become

$$\underline{r}^{C}_{i,j} = \underline{r}_{2/3} + {}^{0}\mathbf{R}_{2/3}\left({}^{2/3}_{2/3}\underline{r}_B + {}^{2/3}\mathbf{R}_B\, {}^{B}_{B}\underline{\xi}_{i,j}\right), \tag{10}$$

where the indexes 2 and 3 correspond to ulna and radius, respectively, and the index B stands for the corresponding bone reference frame (ulna or radius). The notation ${}^{i}_{i}\underline{b}_B$ denotes a vector from the origin of frame $\mathcal{K}_i$ to the origin of frame $\mathcal{K}_B$ in coordinates of frame $\mathcal{K}_i$, and ${}^{i}\mathbf{R}_j$ denotes the orthogonal rotation matrix transforming coordinates with respect to frame $\mathcal{K}_j$ to coordinates with respect to $\mathcal{K}_i$. Here, ${}^{2/3}_{2/3}\underline{r}_B$ and ${}^{2/3}\mathbf{R}_B$ are the (unknown) design parameters describing the constant pose of the bone geometry with respect to the corresponding mechanism link.

Table 1 Numerical data and stiffness coefficients.

	Radius (mm)	Ulna (mm)	Elbow (mm)	Wrist (mm)	c_s N/mm	c_ϑ Nmm	ratio c_s/c_ϑ	$\vartheta_{\max}$ (°)	$s_{\max}$ (mm)
without torque	252.7	252.6	23.3	23.2	0.0763	12.97	169.98	7.98	0.595
pron. torque	252.5	252.4	23.4	23.2	0.0716	12.25	171.09	7.91	0.594
supin. torque	252.6	252.5	23.1	23.4	0.0727	12.41	170.71	7.88	0.601

4 Simulation Results

The described procedure was applied to a trial of three runs of five pro-supination angles each with the same test person. The reference scan was performed with 42 slices at 10 mm distance and the motion scans were performed with 5 slices at 37 mm distance. The hand was rotated from vertical palm ($\varphi = 0°$) in two steps of 45° to supination, i.e., palm up ($\varphi = -90°$), and in two steps to pronation, i.e., palm down ($\varphi = 90°$). The angles were prescribed by a simple plastic device featuring a rotatable handle on a box placed at the end of the MRI tube (not shown here for lack of space). The device allowed us also to produce, via a Bungee cord, a resisting torque while operating the handle. The three runs corresponded to (1) no resistive torque, (2) torque of 4.2 Nm against pronation, and (3) torque of 4.2 Nm against supination.

Optimization was performed with the built-in Matlab function `fmincon`. The typical computational times were 30 min for one optimization run. The initial values were chosen close to the anatomical lengths. The corresponding results are shown in Table 1. It is seen that the mechanism parameters (including the stiffness of the virtual springs) are quite stable and do not depend on the external torque (which is quite high). Thus, deformations at the humero-ulnar joint (dislocation and swaying angle) are basically decoupled from the external loads. Moreover, the resulting translational dislocation is very small, so it does not impair the functionality of the humero-ulnar joint, while allowing the important sway motion of the ulna to take place by approximately 7° at full pronation.

Figure 6 shows the resulting plots for the swaying angle ϑ, the humero-ulnar gap s, the aperture angle ψ_1 and the radio-ulnar torsion angle ψ_2 as a function of the pronation angle φ, respectively. Each figure actually contains *three* plots, one each for the three load cases. However, as seen in the figure, one can not recognize any difference between them. Thus, the wrist load has no influence on the forearm kinematics, and the virtual springs at the two degrees of freedom virtually operate like a kinematical coupling between the two degrees of freedom of the surrogate mechanism.

Table 2 shows the resulting errors between computed and measured bone marrow centers. One can see that the maximal errors are achieved at maximum pronation (palm down) and are below 2 mm. Moreover, the maximally achieved supination and pronation angles agree well with the clinical study performed by Weinberg et al. [12] (supination: $84.4 \pm 5.49°$, pronation $66.4 \pm 7.6°$). Also, the amplitude of

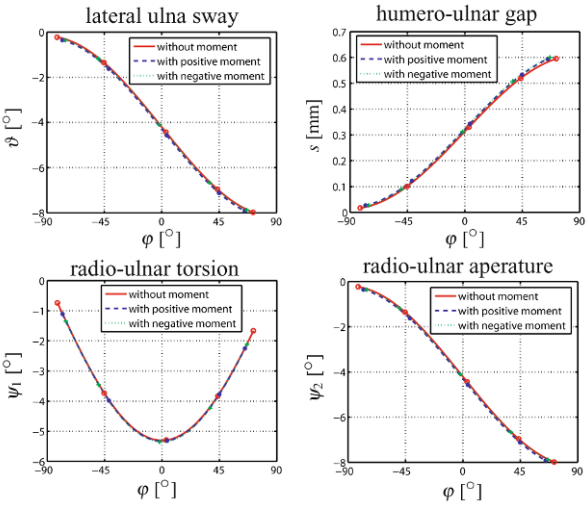


Fig. 6 Histories of elbow variables with different load status.

Table 2 Matching errors of bone marrow centroids (elastokinematic model).

Pro-supination Angle φ (identified)	Error variance of computed marrow centroid from measured centroid on MRI (measurement without outer torque)									
	cross-section 1		cross-section 2		cross-section 3		cross-section 4		cross-section 5	
	radius	ulna	radius	ulna	radius	ulna	radius	ulna	radius	ulna
-82.01	0.74	0.67	0.93	0.14	0.52	0.16	0.48	0.25	0.83	1.19
-45.26	1.17	0.47	0.84	0.23	0.16	0.44	0.32	0.13	0.69	1.01
-3.03	0.81	0.89	0.22	0.77	0.10	0.89	0.52	0.94	1.19	1.11
43.72	1.22	0.57	0.61	0.92	0.32	1.02	0.43	0.30	1.09	0.39
71.67	1.74	1.19	1.58	0.31	1.43	0.95	1.19	1.47	1.14	1.63

the lateral sway angle ϑ matches the value of the clinical study (min: 6°, max 8°). Therefore, the results seem to indicate that the fitting procedure presented in this paper is acceptable for clinical purposes.

5 Conclusions

By the presented surrogate mechanism, an elastokinematic device for simulation of the pro-supination rotation of the forearm has been developed. By attaching two virtual springs to the degrees of freedom, a mechanism with virtually one degree of freedom is produced whose kinematical behaviour is basically independent of the wrist moments. The identification of the model parameters as well as the relative

location of bone geometry with respect to the mechanism links is achieved by a data fitting procedure taking characteristic points of MRI slices and matching them with simulated ones. The model is self calibrating in the sense that it can recognize small axial arm motions within the MRI tube by the identifying the axial location of measured cross sectional area within a smooth reference distribution. The results render good agreement between the observed motion and the simulated ones, indicating that the method could be used in clinical practice. Future work shall include a better self calibration procedure for motions within the MRI tube, a tool for surgery planning for broken and malaligned bones, as well as a study with several test persons to validate the method.

References

1. E. Jonasch and E. Bertel. *Verletzungen bei Kindern bis zum 14. Lebensjahr*. Hefte zur Unfallheilkunde. Springer, Berlin/Heidelberg/New York (1981).
2. R. Fick. *Handbuch der Anatomie und der Mechanik unter Berucksichtigung der bewegenden Muskeln*. Fischer Verlag, Jena (1904).
3. I.A. Kapandji. *Funktionelle Anatomie der Gelenke*, Volume 1. Enke, Stuttgart (1999).
4. T. Nakamura, Y. Yabe, and Y. Horiuchi. A biomechanical analysis of pronation-supination of the forearm using magnetic resonance imaging: Changes of the interosseous membrane of the forearm during pronation-supination. *Nippon-Seikeigeka-Gakkai-Zasshi*, **68**, 14–25 (1994).
5. T.T. Nakamura, Y. Yabe, Y. Horiuchi, and N. Yamazaki. In vivo motion analysis of forearm rotation utilizing magnetic resonance imaging. *Clin. Biomech.*, **14**, 315–320 (1999).
6. M.L. Robbin, K.N. An, R.L. Linscheid, and E.L. Ritman. Anatomic and kinematic analysis of the human forearm using high-speed computed tomography. *Med. Biol. Eng. Comput.*, **24**(2), 164–168 (1986).
7. P. Kasten, M. Krefft, J. Hesselbach, and A.-M. Weinberg. Computer simulation of forearm rotation in angular deformities: a new therapeutic approach. *Injury, Int. J. Care Injured*, **33**, 807–813 (2002).
8. P. Kasten, M. Krefft, J. Hesselbach, and A.-M. Weinberg. Kinematics of the ulna during pronation and supination in a cadaver study: implications for elbow arthroplasty. *Clin. Biomech.*, **19**, 31–35 (2004).
9. A. Kecskeméthy and A. Weinberg. An improved elasto-kinematic model of the human forearm for biofidelic medical diagnosis. *Multibody System Dynamics*, **14**, 1–21 (2005).
10. K. Nojiri, N. Matsunaga, and S. Kawaji. Modeling of pro-supination for forearm skeleton based on MRI. In: *Proceedings of the 17th Word Conference, the International Federation of Automatic Control*, Seoul (Korea), July 6–11, pp. 14767–14772 (2008).
11. S. Miyaguchi, K. Nojiri, N. Matsunaga, and S. Kawaji. Impedance control of pro-supination based on the skeleton model of upper limbs. In: *Proceedings International Conference on Control, Automation and Systems*, October 17–20, pp. 968–973 (2007).
12. A.-M. Weinberg, I.T. Pietsch, M. Krefft, H.C. Pape, M. van Griensven, M.B. Helm, H. Reilmann, and H. Tscherne. Die Pro- und Supination des Unterarms. *Unfallchirurg*, **104**, 404–409 (2001).

Self-Motions of 6–3 Stewart–Gough Type Parallel Manipulators

Adolf Karger

Faculty of Mathematics and Physics, Charles University, Praha, Czech Republic; e-mail: adolf.karger@mff.cuni.cz

Abstract. In the paper we show that every 6–3 Stewart–Gough (SG) platform has self-motions. This yields an example of self-motions for a class of SG platforms which are not architecturally singular. We present three types of self-motions, general self-motion if the platform is in some sense congruent to the base, a butterfly type self motion and a spherical four-bar mechanism self-motion. For the last two cases there is no restriction for the geometry of the platform. We believe that this yields all possible self motions of the 6–3 platform, but we are not able to give the proof, corresponding equations are too complex.

Key words: Parallel manipulators, self-motions, Borel–Bricard motions.

1 Introduction

Self-motions of parallel manipulators of Stewart–Gough type (SG platforms) deserve detailed study, as they can complicate the design and performance of such manipulators. If a manipulator is at a self-motion position, it will immediatelly colapse, an unwanted phenomena. To describe all geometries of SG platforms which could have self-motions at some position is a very complicated problem, which at the moment is very far from its solution. On the other side, SG platforms, which have a self-motion at each possible position, are classified. They are known as architecturaly singular SG platforms. It is not difficult to avoid architecturally singular SG platforms, as easy criteria exist for architectural singularity.

To describe SG platforms which have self-motion only at some positions is a problem which is much more difficult than the description of architecturally singular SG platforms. Only very few results in this respect are known. One is given in [1, 2], where the resulting class is a generalization of the original platform, proposed by Stewart and Gough, see [3, 4]. There exist also some negative results. For manipulators with affinely equivalent platform and base we have no self-motions unless they are architecturally singular (with the exception of translatory motions), see [5, 6]. This could suggest that non-trivial self-motions for non-architecturally singular manipulators almost do not exist. It seems that this is not the case, the

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_38, © Springer Science+Business Media B.V. 2010

problem is that it is very difficult to find them. In this paper we discuss 6–3 manipulators, for which the problem of self-motions can be studied almost to the end. Obtained self-motions are known as spatial motions, we give explicit fomulas for them as self-motions of parallel manipulators. We also show that non-trivial 6–3 manipulators cannot be architecturally singular.

The geometry of 6–3 platforms is very special in comparison to the general case, which makes all the computations relatively simple. Unfortunately, even in that case we cannot prove that we have found all self-motions of this type of manipulators because corresponding equations are very large.

Self-motions of a parallel manipulators are characterized as motions of the manipulator with constant length of its legs. This means that they belong to the category of paradoxical motions, as in this case the manipulator should be a structure. They are also connected to singular positions, they can be also characterized as motions consisting from singular positions satisfying additional condition of fixed leg length.

2 Equations of Self-Motions

We shall study self-motions of 6–3 Stewart–Gough platforms with non-degenerated platform and non-degenerated planar base. This means that we have

$$m_1 = m_2, \quad m_3 = m_4, \quad m_5 = m_6, \tag{1}$$

where m_i and M_i, $i = 1,\dots,6$ are points of the platform and base, respectively.

In that case points m_1, m_3, m_6 move on circles. We shall suppose that centers of corresponding spheres lie on intersecting lines and therefore we may also suppose that

$$M_2 = M_3, \quad M_4 = M_5, \quad M_6 = M_1. \tag{2}$$

We choose coordinate systems in platform and base in such a way, that

$$m_1 = [a_1, b_1], \quad m_3 = [0,0], \quad m_5 = [a_5, 0],$$

$$M_1 = [0,0], \quad M_2 = [A_2, 0], \quad M_4 = [A_4, B_4], \tag{3}$$

the third coordinte is supposed to be equal to zero. We suppose that triangles $\triangle m_1 m_3 m_5$ and $\triangle M_1 M_2 M_4$ are non-degenerated, which means that $a_5 b_1 A_2 B_4 \neq 0$. We shall use the Study representation of the displacement group in the denotation of [7–9]. Details can be found also in [10, 11]. A self-motion of the SG platform appears if the distance of corresponding points of the platform and base remains fixed and the platform still can move with respect to the base.

As points of the base are fixed and points of the platform move on spherical curves, we obtain at the same time a Borel–Bricard motion, see [12, 13]. During a self-motion the manipulator must be at a singular position at each instant, which means that lines $m_i M_i$, $i = 1,\dots,6$ belong to a linear complex and the Jacobian must be singular, see [9, 11, 14, 15].

Let m be a point in the space of the platform and M be in the space of the base. Computations are based on the fact that during a self-motion the distance of points m_i and M_i, $i = 1,\dots,6$ remains fixed. Condition for all motions of the moving space which preserve the distance of the point m to M is given by equation $h = 0$ presented below. This equation was derived by Husty in 1996 in [7] and all computational details can be found also in [11].

$$\begin{aligned} h = {} & R\kappa + 4(y_0^2+y_1^2+y_2^2+y_3^2) + 2(x_3^2-x_0^2)(Aa+Bb) + 2(x_2^2-x_1^2)(Aa-Bb) \\ & + 4[x_0x_3(Ab-Ba) - x_1x_2(Ab+Ba) + (x_0y_1-y_0x_1)(A-a) \\ & + (x_0y_2-y_0x_2)(B-b) - (x_1y_3-y_1x_3)(B+b) + (x_2y_3-y_2x_3)(A+a)] = 0, \end{aligned} \tag{4}$$

where $R = A^2+B^2+a^2+b^2-d(mM)^2$, and $d(mM)$ is the distance of points M and m, $\kappa = x_0^2+x_1^2+x_2^2+x_3^2$. Parameters $x_0,\dots,x_3,y_0,\dots,y_3$ are Study parameters of the motion, restricted to conditions $\kappa = 1$, $U_7 = x_0y_0+x_1y_1+x_2y_2+x_3y_3 = 0$.

Here we start our computations. Interested reader can use a computer session [16] to facilitate computations. We substitute m_i and M_i, $i = 1,\dots,6$ into (4) and obtain six equations

$$h_1 = 0,\dots,h_6 = 0, \tag{5}$$

we have denoted $S_i = R_i - R_1$, $U_i = h_i - h_1, i = 2,\dots,6$. We solve equations $U_2 = 0, U_3 = 0, U_6 = 0, U_7 = 0$ for y_0,y_1,y_2,y_3 and substitute the result into remaining ones. Three equations $U_5 = 0, U_6 = 0, h_1 = 0$ remain. Here we have to suppose that $x_0x_2+x_1x_3 \neq 0$, as this expression appears in the denominator of $y_i, i = 0,1,2,3$.

3 The Butterfly Motion

At first we shall have a look at the special case $x_0x_2+x_1x_3 = 0$. In that case we can write $x_3 = -kx_0, x_2 = kx_1$, where k is an unknown function. We would like to show that k is a constant. We solve equations $U_2 = 0, U_4 = 0, U_5 = 0, U_7 = 0$. The denominator is equal to kx_1 and we have to recognize several cases.

(a) Let $kx_1 \neq 0$. We express U_3 and U_6 and compute the resultant with respect to x_0. We obtain an equation of degree four in k and if k is not constant, this equation must be identically satisfied. Computation shows that this is not possible. For this reason k must have a constant nonzero value. $U_3 - U_6 = 0$ is an equation of the form $f_1x_0^2 + f_2x_1^2 = 0$, where $f_1 - f_2 = 8kB_4a_5$. This shows that we have no solution in this case as both f_1 and f_2 cannot be equal to zero.

(b) Let $kx_1 = 0$.

(b1) Let $k = 0$. We solve $U_2 = 0, U_3 = 0, U_4 = 0, U_7 = 0$ for y_0,y_1,y_2,y_3. The determinant is in that case not equal to zero. Now U_5, U_6, h_1 must be identically equal to zero. We obtain the following solution (here $s = S_2 - 2A_2a_1$).

$$R_1 = -s(s+4A_2a_1)/(4A_2^2), \quad S_3 = s(A_2+a_1)/(A+2), \quad S_4 = s(A_4+a_1)/(A_2),$$

$$S_5 = 2A_4a_5 + s(A_4 - a_5 + a_1), \quad S_6 = s(a_1 - a_5)/(A_2), \quad t_1 = s/(2A_2), \quad t_2 = t_3 = 0, \tag{6}$$

where t_1, t_2, t_3 represent the translation vector of the motion. The solution is a "butterfly" like motion of the manipulator. In that case points M_1, M_2, m_3, m_5 lie in one line, the point m_1 rotates around that line. Let us notice that this motion exists for all 6–3 manipulators, we have no restriction for the geometry of the platform.

(b2) Let $x_1 = 0$. Then also $x_2 = 0$ and we obtain the same solution as above.

Remark. Let us notice that the problem has many symetries, one is due to the fact that the normal vector of the platform plane has two orientations, other ones come from renumbering of the points. We do not distinguish corresponding solutions, as they are geometrically the same.

4 The General Case

Let us continue with the general case again. Let us remark at first that in the case of planar platform and base there is a linear combination of equations $h_i = 0$ which is quadratic in $x_j, j = 0,1,2,3$ and does not contain $y_j, j = 0,1,2,3$. It can be shown that the manipulator is architecturally singular iff this quadratic equation is identically equal to zero, see for instance [17].

We see that from $U = U_4 - U_5$ we can factorize out κ and we are left with three equations,

$$U = 0, \quad U_5 = 0, \quad h_1 = 0, \tag{7}$$

with $U[32], U_5[164], h_1[3282]$, where the number in square brackets means the number of terms (divided by plus or minus sign) in the expression. $U = 0$ is the quadratic equation mentioned above.

Equations (7) represent three algebraic surfaces of degrees 2, 4, and 8 in the projective space determined by homogeneous coordinates x_0, x_1, x_2, x_3. Self-motion appears if these three surfaces have a curve in common. This is expressed by the fact that common curves of any two surfaces in (7) have a common component. It is easy to see that resulting equations are so complicated that they cannot give any general result. Therefore we shall try to find some simpler possibilities which can happen. The simplest case would be if U would be identically equal to zero, which follows the architectural singularity. Let us study this possibility. U is of the form

$$U = r_0(x_0^2 + x_1^2) + r_2(x_2^2 + x_3^2) + r_4x_0x_3 + r_5x_1x_2, \tag{8}$$

where $r_4 - r_5 = b_1A_2$.

This follows that a manipulator of this type cannot be architecturally singular, because $b_1A_2 \neq 0$ and therefore r_4 and r_5 cannot be equal to zero at the same time.

Another possibility we shall study is the case where U factorizes into two linear factors. We can suppose that the equation of one of them has the form $x_0 = g_1x_1 + g_2x_2 + g_3x_3$, which means that the coefficient at x_0 is not equal to zero, as the other pssibility does not bring anything new. Substitution into $U = 0$ yields six equations

$$r_0 g_3^2 + r_4 g_3 + r_3 = 0, \quad r_0(g_1^2+1) = 0, \quad r_0 g_2^2 + r_2 = 0,$$

$$g_1(2r_0 g_3 + r_4) = 0, \quad g_2(2r_0 g_3 + r_4) = 0, \quad 2r_0 g_1 g_2 + r_5 = 0, \tag{9}$$

for three unknowns g_1, g_2, g_3. The only solution is $r_0 = r_2 = r_5 = 0$ and we obtain $U = x_0 x_3 = 0$. We can suppose $x_0 = 0$, as the other possibility yields the same solution. Substitution yields (see the demostration at Fig. 1)

$$S_6 = S_5 + S_3 - S_4 - S_2, \quad A_4 = a_1 A_2/a_5, \quad B_4 = b_1 A_2/a_5, \quad r_4 = -8A_2 b_1 \neq 0, \quad r_3 = 0.$$

This result can be expressed in a geometric way as follows:

The quadratic equation $U = 0$ *factorizes into two linear ones iff triangle* $\triangle M_1 M_2 M_4$ *is similar to the triangle* $\triangle m_3 m_5 m_1$ (in the given order of points).

We continue with the factorization. U_5 is an expression of the form

$$U_5 = j_3 x_1^3 + j_2 x_1^2 x_2 + j_1 x_1 x_3^2 + j_0 x_2 (x_2^2 + x_3^2) + j_4 x_1 x_2^2. \tag{10}$$

We have to discuss conditions under which curves given by $U_5 = 0$ and $h_1 = 0$ have a common factor.

(A) The general possibility is that U_5 is a qubic curve and it appears as a factor in $h_1 = 0$.

(A1) This is always the case if U_5 disappears identically. The solution is

$$A_2 = a_5, \quad B_4 = B_1, \quad A_4 = a_1, \quad R_6 = R_3, \quad R_5 = R_2, \quad R_4 = R_1.$$

This means that triangles $\triangle m_3 m_5 m_1$ and $\triangle M_1 M_2 M_4$ are congruent (in given order) and this is the sufficient and necessary condition for the existence of self-motion of this kind.

Remark. This does not mean that the upper platform is congruent to the lower platform, because for that we need the same corresponding distances, for instance the distance of $m_1 m_2$ is zero, but the distance of $M_1 M_2$ is equal to a_1, which is nonzero. The case of congruent platform and base yields no self-motions, see [5].

The motion curve in the space of Euler parameters is a planar curve of degree six, its equation has 247 terms and therefore it is not presented.

5 The Four-Bar Self-Motion

(A2) The other possibility is that U_5 is a nonzero factor in h_1. As U_5 is a cubic curve with zero coefficient at x_3^3, the highest coefficient at x_3 in h_1, which is of degree eight, must be also zero. This yields $S_5 = S_2 a_5/A_2 + S_4 - 2a_1(A_2 + a_5)$ and substitution into U_5 shows that U_5 factorizes, and so a factor of U_5 must be contained in h_1.

(B) Let us study the possibility that U_5 factorizes into a linear and quadratic factors. This means that it must contain a line. Let us suppose that it is given by equation $x_3 = Ax_1 + Bx_2$. Substitution into (10) yields four equations

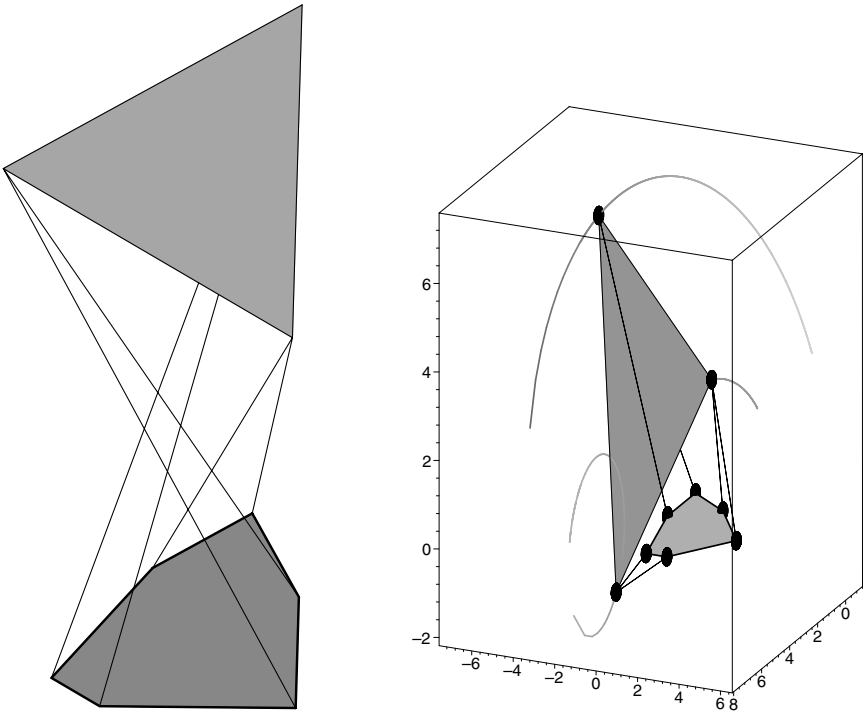


Fig. 1 6–3 manipulator with self-motion (left) and parts of three circular trajectories (right).

$$j_3 + j_1 A^2 = 0, \quad j_2 + j_1 AB + j_0 A^2 = 0, \quad j_1 B^2 + 2 j_0 AB + j_4 = 0, \quad j_0 (B^2 + 1) = 0.$$

From the last equation we have $j_0 = 0$, and U_5 factorizes into

$$U_5 = x_1 (j_3 x_1^2 + j_2 x_1 x_2 + j_1 x_3^2 + j_4 x_2^2). \tag{11}$$

We express S_5 from $j_0 = 0$,

$$S_5 = S_2 a_5 / A_2 + S_4 + 2 a_1 (a_5 - A_2) \tag{12}$$

and substitute the result into $h_1 = 0$. We have to express condition under which the quadratic factor of U_5 is a factor in h_1, the linear one was already considered. We notice that x_3 appears only in even powers in U_5 and h_1, we express x_3^2 from $U_5 = 0$ and substitute the result into h_1. The coefficient at x_3^2 in U_5 must be nonzero, as otherwise $U_5 \equiv 0$ and we obtain a solution already known. After substitution of x_3 into h_1 we obtain

$$h_1 = (a_1 x_1 + b_1 x_2)^2 ((a_5 - a_1) x_1 - b_1 x_2)^2 (k_2 x_2^2 + k_1 x_1 x_2 + k_0 x_1^2),$$

where h_1 must be identically equal to zero. We shall consider only the last factor, as the other ones yield "butterfly" motions. We can suppose $(A_2 - a_5)(a_5 - 2a_1) \neq 0$, as the opposite does not bring anything new. As the coefficient at R_1 is not zero, we express R_1 from $k_1 = 0$ and substitute the result into k_0, k_2. We obtain two equations, $k_2 = 0, k_0 = 0$. Each of them splits into two factors, let us write $k_0 = k_{01}k_{02}, k_2 = k_{21}k_{22}$, where $k_{21} = (S_2 a_5 - 2A_2^2 a_1)$.

We have to consider separate cases again.

(i) Let $k_{21} = 0$. Both factors in k_0 yield

$$S_2 = 2A_2^2 a_1/a_5, \quad S_4 = 2(A_2^2 - a_5^2)(a_1^2 + b_1^2)/a_5^2, \quad R_1 = -S_4/2. \tag{13}$$

(iia) $k_{22} = 0, k_{01} = 0$ yields again (13).

(iib) Let $k_{02} = 0, k_{22} = 0$. Equation $k_{02} = 0$ is linear in S_4. If the coefficient at S_4 is equal to zero, we obtain (13). Therefore we can suppose that this coefficient is not zero. We express S_4 and substitute the result into $k_{22} = 0$.

We obtain a quadratic equation for S_3, we solve it and substitute both S_4 and S_3 into already computed x_3^2. We obtain

$$x_3^2 = -(f^2 s^2 + a_1^2 t^2 \pm 2a_1 fst)[x_1 b_1 (f - a_1) - x_2(fa_1 + b_1^2 + st)]^2/(b_1^4 (f^2 - a_1^2)^2), \tag{14}$$

where for simplicity we write

$$a_5 - a_1 = f, \quad \sqrt{(a_1^2 + b_1^2)} = s, \quad \sqrt{(b_1^2 + f^2)} = t.$$

As $(f^2 s^2 + a_1^2 t^2)^2 - 4a_1^2 f^2 s^2 t^2 = b_1^4 (f^2 - a_1^2)^2 > 0$, we see that in this case there is no real solution of the problem.

This shows that we are left with (13) as the only solution. Let us concentrate at the geometrical meaning of (13). Substitution shows that the point m_1 during the whole motion coincides with the point M_4. Points m_3 and m_5 run on circles with centers in points M_1 and M_2. Let us recall that $\triangle M_1 M_2 M_4$ is similar to the $\triangle m_3 m_5 m_1$. We see that the manipulator has self-motion which is a spherical four-bar mechanism. Formulas connected with this motion are rather complicated and to simplify them we shall choose a more suitable system of coordinates. At the same time we shall have a look if there exist more general manipulators which can perform this four-bar self-motion.

More generally let us look for self-motions for which the point m_3 is during the whole self-motion coincident with the point M_1. As both points are origins of the corresponding coordinate systems, we are looking for self-motion of the manipulator, for which the translational part is identically equal to zero. This means that we have to solve equations (5) with the suplementary condition $y_0 = y_1 = y_2 = y_3 = 0$. Substitution yields $R_1 = S_3 = S_4 = S_6 = 0$. Three equations remain, $U_2 = 0, U_5 = 0, \kappa = 1$.

This is a solution, the motion is represented as the intersection curve of two quadrics in three-dimensional projective space, projected at the unit sphere. We obtain a spherical four-bar motion.

6 Conclusions

To describe all self-motions of manipulators of Stewart-Gough type seem to be a very complicated problem. Even in the case of 6–3 manipulators we obtain complicated formulas and we are not able to prove that we have described all possible self-motions. The curve describing the self-motion is in general not parametrizable and therefore it is not possible to visualize trajectories of such a motion in general. Approximate parametrization could be the corresponding tool to do it.

Acknowledgements The research reported here was supported by Grant MSM 0021620839 of the Ministry of Education and Sports of the Czech Republic.

References

1. Karger A., Husty M.: Classification of all self-motions of the original Stewart–Gough platform, *Computer-Aided Design* **30**, 205–215 (1998).
2. Karger A: New self-motions of parallel manipulators. In: J. Lenarčič and P. Wenger (Eds.), *Advances in Robot Kinematics*, Springer, Dordrecht, pp. 275–282 (2008).
3. Gough V.E.: Contribution to discussion to papers on research in automobile stability and control in tyre performance. By Cornell Staff, *Proc. AutoDiv. Inst. Mech. Eng.*, 392–395 (1956).
4. Stewart D.: A platform with six degrees of freedom. *Proc. of the Inst. of Mech. Eng.* **180**(1), 371–386 (1965).
5. Karger A.: Singularities and self-motions of equiform platforms. *Mech. Mach. Theory* **36**, 801–815 (2001).
6. Karger A.: Singularities and self-motions of a special type of platforms. In: J. Lernarčič and F. Thomas (Eds.), *Advances in Robot Kinematics*, Kluwer Academic Publishers, Dordrecht, pp. 355–365 (2002).
7. Husty M.L.: An algorithm for solving the direct kinematics of general Stewart–Gough platforms. *Mech. Mach. Theory* **31**(4), 365–380 (1996).
8. Husty, M.L., Karger A.: Self-motions of Griffis–Duffy type parallel manipulators. In *Proceedings of the 2000 IEEE international Conference on Robotics & Automation*, San Francisco, CA, pp. 7–12 (2000).
9. Karger A.: Architecture singular planar parallel manipulators. *Mech. Mach. Theory 38*, 1149–1164 (2003).
10. Bottema O., Roth B.: *Theoretical Kinematics*, Dover Publishing (1992).
11. Husty M., Karger A., Sachs H., Steinhilper W.: *Kinematik und Robotik*. Springer (1997).
12. Borel E.: Mémoire sur les déplacements à trajectoires sphérigues. *Mém. présentés par divers savants, Paris (2)*, **33**, 1–128 (1908).
13. Bricard R.: Mémoire sur les déplacements à trajectoires sphérigues. *J. École Polyt. (2)*, **11**, 1–96 (1906).
14. Merlet J.P: Singular configurations of parallel manipulators and Grassmann geometry. *Int. J. Robotics Research*, *8*, 45–56 (1992).
15. Röschel O., Mick S.: Characterisation of architecturally shaky platforms. In: J. Lenarčič and M.L. Husty (Eds.), *Advances in Robot Kinematics*, Kluwer Academic Publishers, Dordrecht, pp. 465–474 (1998).
16. www.ag.jku.at/karger/selfmotions.txt
17. Karger A.: Parallel manipulators with simple geometrical structure. In: M. Ceccarelli (Ed.), *Proceedings of EUCOMES 08*, Springer, Dordrecht, pp. 463–470 (2009).

Constraint Compliant Control for a Redundant Manipulator in a Cluttered Environment

S. Rubrecht[1], V. Padois[1], P. Bidaud[1] and M. de Broissia[2]

[1]*Institut des Systèmes Intelligents et de Robotique, Université Pierre et Marie Curie, Paris, France; e-mail: {rubrecht, padois, bidaud}@isir.fr*
[2]*Bouygues Travaux Publics, 78280 Guyancourt, France; e-mail: m.debroissia@bouygues-construction.com*

Abstract. In this paper, we present an iterative kinematics control law for redundant manipulators overcoming some usual problems associated to cluttered environments (constraints violations, oscillations, environment dilatation). The Constraints Compliant Control approach relies on a passive avoidance scheme (no motion generation for constraints avoidance) on a limited number of constraints selected from a vicinity analysis. A solution scaling based on the feasible motions with respect to the constraints enables to reach the frontiers of the workspace. Two missions described as sequences of key frames are simulated to compare the performances of the Constraint Compliant Control with state of the art control laws. The obtained computation times remain acceptable to consider a use in real time.

Key words: Redundancy resolution, inverse velocity kinematics, control.

1 Introduction

Industrial robotics manipulator missions have evolved from well defined tasks in structured environments to missions where the tasks are more complex (involving multiple objectives among which various constraints) and where the environments are not always known in advance and can be harsh and cluttered. As an example, the motivations of the presented work consider the use of a teleoperated manipulator aimed at working in a tunnel boring machine cutter head (see Fig. 1 and [1]).

In such environments, the use of redundant manipulators enables to specify simultaneously various goals on different parts of the manipulator, explicitly or automatically. It gives the possibility to fulfill different tasks while strictly satisfying a certain amount of constraints (collision avoidance, joint position and velocity limits). However, to our knowledge, there is no multi-objective method that guarantees a safe behaviour whithout substantially reducing the motion capabilities, especially when the manipulator is expected to get close to the constraints surfaces (which may be operational surfaces). This is the problem tackled in this paper.

Consider a manipulator with n DOFs and a set of objectives (not necessarily imposed to the same part of the manipulator) involving m operational displacements

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_39, © Springer Science+Business Media B.V. 2010

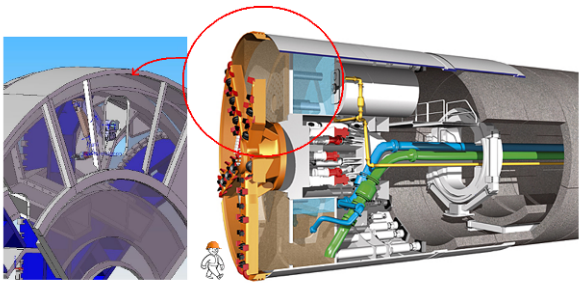


Fig. 1 Tunnel boring machine: manipulator in the excavation room.

$\boldsymbol{\delta x}$. The relation between the operational and the joints displacements (considered here as the actuators input signal, extension to torque controlled actuators is trivial)

$$\boldsymbol{\delta x} = J\boldsymbol{\delta q} \tag{1}$$

where J is the Jacobian associated to the objectives. In order to satisfy the constraints, usual avoidance strategies rely on an avoidance term $\boldsymbol{\delta x}_C$ derived from a potential based on the inverse of the distance to the constraint [2]. This strategy is called *active*, as a motion is generated to avoid the constraint. Thus, the expected behaviour will result from the combination of the operational tasks ($(J_T,\boldsymbol{\delta x}_T)$, concatenation of the tasks) and the constraints ($(J_C,\ \boldsymbol{\delta x}_C)$, concatenation of the constraints). In the particular cases of joints related constraints (joint position and velocity limits), the constraint jacobian matrix is $[0,\ldots,0,\pm1,0,\ldots,0]$, the ±1 rank being at the joint number in the manipulator chain, the sign depending on the limit type (maximum or minimum).

These motions can be strictly prioritized in a multiobjective control law (see [4])

$$\boldsymbol{\delta q} = J_1^{\#}\boldsymbol{\delta x}_1 + (J_2 P_{J_1})^{\#}(\boldsymbol{\delta x}_2 - J_2 J_1^{\#}\boldsymbol{\delta x}_1) \tag{2}$$

where indexes 1 and 2 can be replaced by C and T and conversely. The # is a pseudoinversion operator (see [3]) and P_{J_1} is a projector on the kernel of J_1 (details about inversions and projectors are exposed in Section 3.1).

Maciejewski [4] and Chaumette [5] proposed to put the constraints avoidance at the bottom of the task hierarchy ($1 \leftarrow T$, $2 \leftarrow C$; referenced as control law A). It enables to stricly fulfill the tasks, but in case of conflict between the tasks and the constraints, the manipulator may violate the constraints. To avoid this, Sentis [6] puts the constraints at the top of the hierarchy ($1 \leftarrow C$, $2 \leftarrow T$; referenced as control law B). However, this choice has a bad impact on the reachability of the manipulator as the avoidance motions need activation thresholds, which results in oscillations on the functional parts of the manipulator. Moreover, as the constraints are not homogeneous (obstacle avoidance involves body lengths that depends on the obstacle position w.r.t. the robot, while joint limits do not), it is not possible to use a bounded potential. It implies a consequent size of the threshold, as the potential

should grow from zero to infinity along it, and also the possible presence of infinite terms in the control law.

In order to avoid oscillations and infinite terms, Baerlocher uses in [7] an iterative approach based on an activation matrix that pre-multiplies the Jacobian of the tasks. The algorithm deactivates the joints which violate their boundaries, so they are not included in the model inversion. However, this approach does not include other types of constraints and it often resorts to iterations in cases for which usual approaches find directly an *admissible* (*i.e.* constraints compliant) solution.

This paper presents a framework that extends the approach of Baerlocher to the collisions avoidance and satisfies the joint velocity limits; in particular, when the environment is not overconstrained (number of DOF sufficient to track the tasks while complying to the constraints), the control law is equivalent to the classical control law A. A particular attention has been paid to keep the computation times in the range of the one obtained with control law A and B.

In Section 2, the Constraint Compliant Control (CCC), relying on the principle of passive avoidance, is introduced. Then, Section 3 presents the simulations of two missions and the comparative results of the control laws presented previously (A, B and CCC). Finally, Section 4 gathers the conclusions and the work perspectives.

2 Constraint Compliant Control

This section exposes our contribution. First, the passive avoidance principle is developped, according to which the robot should not move to avoid static constraints. Then, the control law expression is exposed; finally, the whole algorithm is presented.

2.1 Passive Avoidance

The approach of Baerlocher in [7] is equivalent to the insertion of a superior hierarchical level in which operational displacement would be null ($\boldsymbol{\delta x}_C = 0$). It has the advantage to satisfy the joint boundaries in all cases. The extension to every static constraints (static obstacles, joint position and velocity limits) is done using the following control law

$$\boldsymbol{\delta q} = J_C^{\#}\mathbf{0} + (J_T P_{J_C})^{\#}(\boldsymbol{\delta x}_T - J_T J_C^{\#}\mathbf{0}) = (J_T P_{J_C})^{\#}\boldsymbol{\delta x}_T \tag{3}$$

which ensures a strict compliance with each constraint in J_C. This avoidance method is *passive* as no motion is generated by the proximity to a constraint. As what is described by the constraints expression is actually static in the physical world, no arbitrary motion is needed to satisfy them. Anyway, it is not often mandatory to forbid motions in all the constraints directions; it is not even desired, as it prevents from getting closer to the constraints, but also from moving away from them. So,

it is relevant to iterate on the constraints combinations to find the set of constraints that must be passively avoided to obtain an *appropriate* solution $\boldsymbol{\delta q}$, i.e. that minimizes the operational error while being admissible. Let J_{CC} be a matrix containing a combination of lines of J_C. The control law then becomes

$$\boldsymbol{\delta q} = (J_T P_{J_{CC}})^{\#} \boldsymbol{\delta x}_T \tag{4}$$

Iterations are carried out to find the lines combination J_{CC} which yields an appropriate $\boldsymbol{\delta q}$. In particular, the admissibility test is performed using Eq. (7).

2.2 Active Avoidance in Additional Objective

The solutions admissibility can be increased by adding an extra term of active avoidance on the constraints that are not avoided passively:

$$\boldsymbol{\delta q} = \underbrace{(J_T P_{J_{CC}})^{\#} \boldsymbol{\delta x}_T}_{\text{Passive avoidance}} + \underbrace{(J_{\overline{CC}} P_{\left[\begin{smallmatrix} J_{CC} \\ J \end{smallmatrix}\right]})^{\#} (\boldsymbol{\delta x}_{\overline{CC}} - J_{\overline{CC}} (J_T P_{J_{CC}})^{\#} \boldsymbol{\delta x}_T)}_{\text{Active avoidance}} \tag{5}$$

where $J_{\overline{CC}}$ is the complement of the lines of J_{CC} in J_C, $\boldsymbol{\delta x}_{\overline{CC}}$ is the desired operational avoidance displacement associated to $J_{\overline{CC}}$[1] and $P_{\left[\begin{smallmatrix} J_{CC} \\ J \end{smallmatrix}\right]}$ is the projector on the kernel of the concatenation of J_{CC} and J. The extra term tends to move the manipulator away from the constraints as long as the tasks are not impacted. In particular, when J_{CC} is empty, the control law behaves like control law A and a single iteration is needed when the situation is not overconstrained. The avoidance coefficients norm $|\delta x_{\overline{CC},i}|$ can be limited to avoid values tending to infinity. Finally, there are no oscillations on the functionnal part as the active avoidance term is under the task related term in the hierarchy.

2.3 Particular Case of the Joint Velocity Limit – Scaling

The joint velocity limits should not be addressed with passive avoidance, as it would stop a joint to prevent it from going too fast. If a joint displacement is too high, the solution is scaled to reach the maximal admissible displacement. To keep the motion coherency, the scaling is done in a way that preserves the operational direction

$$\boldsymbol{\delta q} \leftarrow \boldsymbol{\delta q} \min_{0 \le i \le n} \frac{\delta q_{i,max}}{||\delta q_i||} \tag{6}$$

[1] In a practical aspect, $J_{\overline{CC}}$ (resp. $\boldsymbol{\delta x}_{\overline{CC}}$) can be replaced by J_C (resp. $\boldsymbol{\delta x}_C$, desired operational avoidance displacement associated to J_C) in Eq. (5) without any consequence on the result.

where $\delta q_{i,max}$ is the maximum displacement of joint i on one iteration. This step is carried out only if a joint displacement δq_i is higher than $\delta q_{i,max}$.

The test of admissibility with respect to the constraints is performed through

$$J_C \boldsymbol{\delta q} \leq \boldsymbol{l} \tag{7}$$

where $\boldsymbol{l}$ is the concatenation of the maximum instantaneous displacements. In order to increase the validity of the solutions, the scaling is extended to all the solutions

$$\boldsymbol{\delta q} \leftarrow \boldsymbol{\delta q} \min_{\substack{0 \leq i \leq m \\ (J_C \delta q)_i > 0}} \left(\frac{\boldsymbol{l}_i}{(J_C \boldsymbol{\delta q})_i} \right) \tag{8}$$

The scaling is done only if Eq. (7) is not verified. This step enables to obtain an admissible solution for each constraints combination in J_{CC}. Actually, as all the constraints are in the form of Eq. (7), the solutions space is convex and contains the null solution (no motion). So, in every direction of the joint displacement space, there exists an admissible solution, which norm is null in the worst case. This method enables to get as close as possible to the constraints if there is a persistent demand in that way. The general algorithm of a control iteration is given by Algorithm 1.

Algorithm 1 : CCC

$J_C \leftarrow$ constraints under the active avoidance threshold
$\boldsymbol{\delta q}_{comb} \leftarrow \mathbf{0}$ rad; $\boldsymbol{\delta q} \leftarrow \mathbf{0}$ rad
$\text{Err}_{comb} \leftarrow 0$ m; $\text{Err} \leftarrow 1$ m; $\epsilon \leftarrow 10^{-2}$ m;
for all Lines Combinations J_{CC} in J_C **do**
  $\boldsymbol{\delta q}_{comb}$ Computation - Eq. (5)
  Admissibility Test - Eq. (7)
  Scaling - Eq. (8)
  $\text{Err}_{comb} \leftarrow ||J \boldsymbol{\delta q} - \boldsymbol{\delta x}_T||$
  if $\text{Err}_{comb} \leq \text{Err}$ **then**
    $\text{Err} \leftarrow \text{Err}_{comb}$; $\boldsymbol{\delta q} \leftarrow \boldsymbol{\delta q}_{\text{comb}}$
    if $\text{Err} \leq \epsilon$ **then**
      break
    end if
  end if
end for
send $\boldsymbol{\delta q}$

3 Implementation and Comparative Results

This section presents the results of control laws A, B and CCC on two missions in the same environment. The proposed environment is composed of a column and a wall; the manipulator has 7 DOFs, all the links being 1 DOF rotational joints. The environment and the manipulator are represented in Fig. 2. The presented experi-

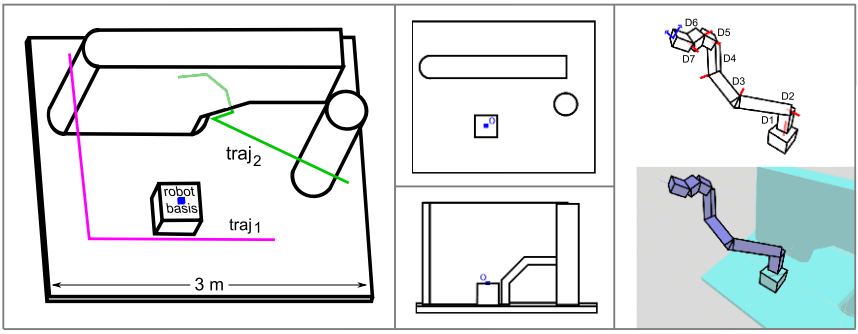


Fig. 2 Views of the environment and the two trajectories to track; manipulator schemes.

ments simulate two inspection missions involving trajectories close to the obstacles. For each mission, the manipulator must track a 3 coordinates trajectory: it is assumed that the effector (camera) has the orientation DOFs needed to observe the points to be inspected. For the sake of simplicity, the results presented here do not integrate joint boundaries avoidance even though the proposed framework can deal with this type of constraint without any specific difficulty.

- *Mission 1: Go around the wall by the left side.* The environment is barely constrained in that area, the manipulator tracks a trajectory ($traj_1$, in Fig. 2) of 330 points on a single way of 3.50 m, so a displacement of 1.1 cm is expected for each iteration. The mission is *feasible*, i.e. the number of DOFs of the manipulator enables to fulfill the mission while avoiding the constraints.
- *Mission 2: Reach a point behind the wall.* The trajectory ($traj_2$, in Fig. 2) has 560 points, for a go and come back trajectory (getting out of a very constrained configuration can be a problem). The total distance is 5.20 m long, so the expected displacement is 0.9 cm at each iteration. The mission is not *feasible* as the manipulator is not long enough to reach the furthest point.

3.1 Implementation

The implementation is done in C++, and uses KDL (see [8]) and Boost::ublas[2] libraries. We detail the following implementation elements:

- *Desired displacement* $\boldsymbol{\delta x}_T$: difference between the current position and the current trajectory point.
- *Pseudoinversion*: in the control law Eq. (5), in order to avoid inversion problems in the neighborhood of singularities, the pseudoinversions with exponent # are done using the Damped Least Square (DLS) method (see [9]), where the damping factor λ is chosen as 0.50. A common way of computing P_J is given by

[2] http://www.boost.org/doc/libs/1_41_0/libs/numeric/ublas/doc/index.htm

$P_J = (I - J^{\#}J)$; in practice, the DLS method induces an error that distorts the projection, making the influence of lower priority objectives on upper priority objectives possible, which is not acceptable in our case. A safe way to compute P_J can be obtained directly from the SVD of J which provides an access to the projector on the kernel of J without requiring its inversion (as mentionned in [10]). For the same reasons, a product $(J_a P_{J_b})^{\#}$ is always pre-multiplicated by P_{J_b}.

- *Active avoidance*: for control laws A and B, the active avoidance threshold is fixed to 15 cm, the gains are proportional (factor 2.5 10^{-3}) to the inverse of the distance to the constraint. For the CCC, the active avoidance is fixed to 4 cm, the gains are the same than for control laws A and B but the maximum value of the avoidance magnitude δx_C is fixed to 0.25 (distance of 1 cm between the manipulator closest point and the obstacle).
- *Distance computation*: the distance computation is provided by the collision detection package SWIFT++.[3] The information given by SWIFT++ is, for each segment, the point of the segment that is the closest to the obstacle. The approach that consists in constraining only the closest point to the obstacle (based on [4] for the whole manipulator and on [11] for convex segments) is valid in continuous time. In discrete time, progressive displacements can violate constraints because of segment rotations. To our knowledge, no work has been carried out on that topic; a contribution of Kanehiro et al. in [12] deals with not strictly convex segments, in which the problems are similar to those encountered when getting from continuous to discrete time. To avoid the effects of this phenomenon, an envelope of 2 cm has been added around the environment.
- *Joint instantaneous displacement limit*: $\boldsymbol{\delta q_{max}}$ has been fixed to 0.02 rad.

3.2 Results and Analysis

Figures 3 and 4 present the results obtained on the 2 missions with control laws A, B and CCC. The computation times are given comparatively as they depend on the implementation and computing power.

- *Mission 1: Go around the wall by the left side.*
  - *Control law A*. As the behavior is identical to the CCC on mission 1 (see 2.2), it is not represented in Fig. 3.
  - *Control law B*. The operational position mean error is 3.5 cm, the tracking is not optimal especially at the end where the effector gets close to the wall: oscillations are generated due to the thresholds (observable on both graphs) and the operational error grows up to 13.1 cm.
  - *Control law CCC*. The behavior is identical to control law A: as there is no collisions when tracking the path with active avoidance at a lower level, the

[3] http://gamma.cs.unc.edu/SWIFT++

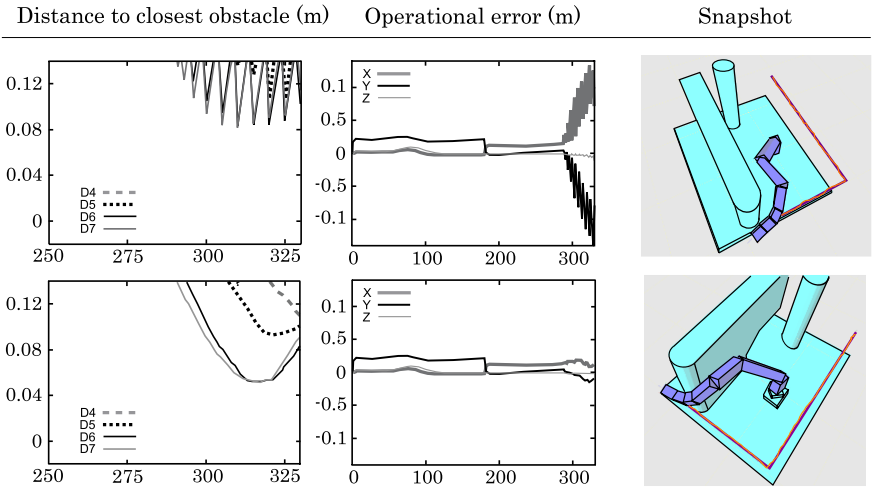


Fig. 3 Mission 1 results. Graphs of line 1 are obtained with control B, line 2 with control CCC.

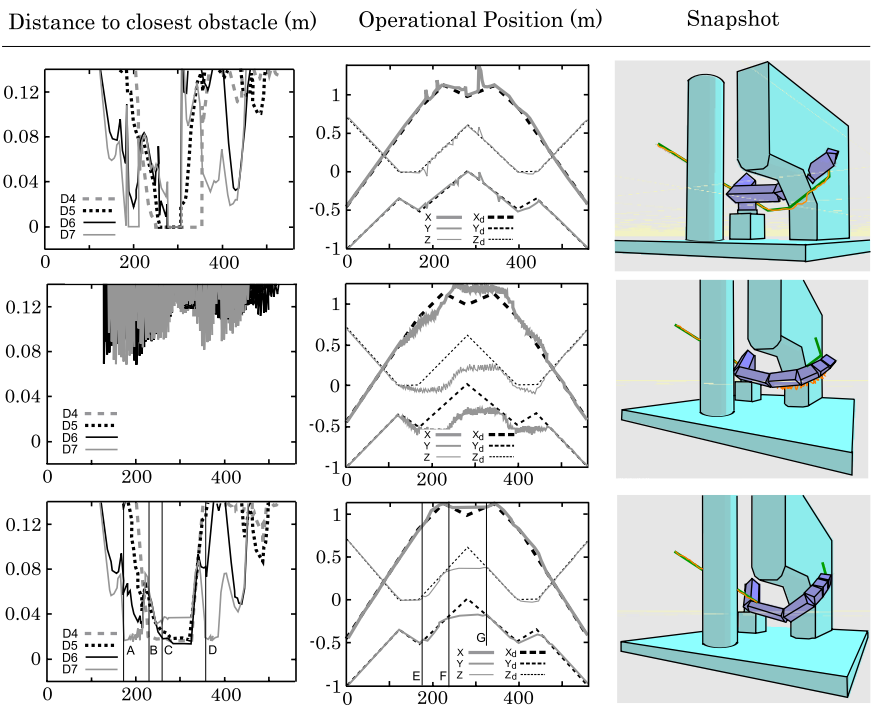


Fig. 4 Mission 2 results. Graphs of line 1 are obtained with control A, line 2 with control B, line 3 with control CCC.

passive avoidance is not used and the manipulator has the same tracking error along the trajectory (operational position error inferior to 2.6 cm).

- *Mission 2: Join a point behind the wall.*
  - *Control law A*. The operational position mean error is 3.8 cm and reaches 35 cm on peaks. When the situation gets overconstrained, collisions occur (the distances to the obstacle reach 0 for DOFS 4, 5 and 7) as it is not possible to satisfy tasks and constraints; the peaks on the operational positions comes from avoidance terms tending to infinity.
  - *Control law B*. The oscillation phenomenon is higher than for mission 1 and the manipulator remains far from the path (operational position mean error 22.8 cm, max error 59.4 cm)
  - *Control law CCC*. When the manipulator comes close to the environment (enveloppe distance: 2 cm), the passive avoidance clamps the directions to the obstacles (point A, B, C and D on the first column graph) and the concerned segments moves along the orthogonal directions. When the manipulator seems completely stuck (point E on the second column graph), the scaling step enables little displacements to track as much as possible the desired displacements. Finally, when the direction is inversed (point F on the 3rd column graph), the manipulator is able to get away from the constraints directly. The computation time grows up 6.7 times w.r.t. the ones obtained with control law *A* on mission 1.

4 Conclusion and Perspectives

The *CCC* is an iterative control method that solves the hierarchical multi-objective control problem while satisfying any number of fixed constraints: obstacles, joint boundaries, joints velocity limits. The passive avoidance principle and the solutions scaling enables to overcome the drawbacks of active avoidance at the top (optimality loss, oscillations) or at the bottom (constraints violation, infinite terms) of the hierarchy, while ensuring a computation time low enough to consider its use in real time on classical manipulators.

The *CCC* performances can be compared to the one obtained with convex optimization algorithms (even if the latter cannot ensure strict priorities between the hierarchy levels). As an example, the algorithm QuadProg++[4] satisfies the constraints with a maximum computation time of 3 times the one of control law *A*. Nevertheless, the insufficiency of the avoidance method (see Section 3.1) added to the algorithm trend to run along the constraints make the manipulator not able to get away of the most constrained configuration: it does not track the second part of the trajectory.

The work perspective includes:

[4] http://sourceforge.net/projects/quadprog/

- The velocity continuity can be handled by including the limits of joint acceleration constraints but they introduce incompatibilities with obstacles and joint position limits; the use of virtual constraints (dampers) is under evaluation.
- Locally, the constraints combination choice can be guided by the constraints criticity or by favouring combinations retained for previous iterations, etc.
- In a more global scope, this choice can be adapted to the missions and it can depend on the parts of the robot: active avoidance for better motion capabilities, passive avoidance to work close to the constraints.

Acknowledgements This work is a part of the TELEMACH project. It is supported by the French National Research Agency (ANR), Interactive Systems and Robotics Program 2007 (PSIROB07).

References

1. Rubrecht, S., Padois, V. and Bidaud, P.: Evolutionary design of a robotic manipulator for a highly constrained environment. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, Workshop on Exploring Nnew Horizons in Evolutionary Design of Robots*, pp. 59–64 (2009).
2. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research* **5**(1), 90–98 (1986).
3. Ben Israel, A. and Greville, T.N.E.: *Generalized Inverses: Theory and Applications*, 2nd edition. Springer (2003).
4. Maciejewski, A. and Klein, C.: Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. *The International Journal of Robotics Research* **4**(3), 109–117 (1985).
5. Chaumette, F. and Marchand, E.: A new redundancy-based iterative scheme for avoiding joint limits application to visual servoing. In: *IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 1720–1725 (2000).
6. Sentis, L. and Khatib, O.: Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. *The International Journal of Humanoid Robotics* **2**(4), 505–518 (2005).
7. Baerlocher, P. and Boulic, R.: An inverse kinematic architecture enforcing an arbitrary number of strict priority levels. *The Visual Computer: International Journal of Computer Graphics* **20**(6), 402–417 (2004).
8. Bruyninckx, H.: Open Robot Control Software: The OROCOS project. In: *IEEE International Conference on Robotics and Automation*, Vol. 3, pp. 2523–2528 (2001).
9. Nakamura, Y. and Hanafusa, H.: Inverse kinematics solutions with singularity robustness for robot manipulator control. *Journal of Dynamic Systems, Measurement and Control* **108**, 163–171 (1986).
10. Salaun, C., Padois, V. and Sigaud, O.: Control of redundant robots using learned models: An operational space control approach. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 878–885 (2009).
11. Faverjon, B. and Tournassoud, P.: A local based approach for path planning of manipulators with a high number of degrees of freedom. In: *IEEE International Conference on Robotics and Automation*, Vol. 4, pp. 1152-1159 (1987).
12. Kanehiro, F., Lamiraux, F., Kanoun, O., Yoshida, E. and Laumond, J-P.: A local collision avoidance method for non-strictly convex polyhedra. In: *Proceedings of Robotics: Science and Systems IV* (2008).

Geometric Interpolation by Quartic Rational Spline Motions

B. Jüttler[1], M. Krajnc[2] and E. Žagar[2]

[1]*Institute of Applied Geometry, Johannes Kepler University, Linz, Austria;*
e-mail: bert.juettler@jku.at
[2]*IMFM and FMF, University of Ljubljana, Slovenia;*
e-mail: {marjeta.krajnc, emil.zagar}@fmf.uni-lj.si

Abstract. We discuss piecewise rational motions with first order geometric continuity. In addition we describe an interpolation scheme generating rational spline motions of degree four matching given positions which are partially complemented by associated tangent information. As the main advantage of using geometric interpolation, it makes it possible to deal successfully with the unequal distribution of degrees of freedom between the trajectory of the origin and the rotation part of the motion.

Key words: Motion design, geometric interpolation, rational spline motion, geometric continuity.

1 Introduction

Geometric interpolation techniques [1] generate spline curves from given geometric data, such as points, tangents, and perhaps even curvature information. As an advantage, they generally require lower polynomial degrees than standard methods. For instance, a polynomial cubic in the plane can match two points with associated tangents and curvatures [2], while the interpolation of two points with associated first and second derivatives needs curves of degree five.

Geometric interpolation techniques are closely related to the concept of geometric continuity, which generalizes the notion of derivative continuity by eliminating the influence of the parameterization [3]. The present paper introduces the new notion of geometrically continuous rational spline motions and describes a first geometric interpolation method for motion design using piecewise quartic motions.

Bennett biarcs, which form another subset of the class of quartic rational spline motions, have recently been used for geometrically interpolating two positions with associated tangent information [4]. It was observed that the collisions between fixed planes and moving points (and similar for fixed points and moving planes) can be detected simply by solving quartic polynomials. However, as a disadvantage, the class of motions described by Bennett biarcs cannot handle motions with changing chirality (i.e., orientation of the instantaneous screw motions).

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_40, © Springer Science+Business Media B.V. 2010

The method described in this paper uses two segments of rational quartic motions for interpolating three positions, two of which are complemented by associated tangent information. Compared to Bennett biarcs, they can handle one additional position, and no problems with the chirality of the motion are present. Still, the collision between moving points and fixed planes leads simply to quartic polynomials, while this is no longer true for collisions between moving planes and fixed points.

The remainder of this paper is organized as follows. In Section 2 geometrically continuous rational spline motions are presented. The next section introduces the interpolation problem for quartic rational spline motion, and Section 4 provides its explicit construction together with some numerical examples. The paper is concluded by some ideas for the future work.

2 Geometrically Continuous Rational Spline Motions

A rigid body motion is described by the trajectory $\mathbf{v}(t) = (v_1(t), v_2(t), v_3(t))^\top$ of the origin of the moving system and by the 3×3 rotation matrix $R(t)$. Using quaternions $\mathbf{q} = (q_0, q_1, q_2, q_3)^\top$, the rotation matrix R can be represented by

$$R = \frac{1}{q_0^2+q_1^2+q_2^2+q_3^2} \begin{bmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2(q_1q_2-q_0q_3) & 2(q_1q_3+q_0q_2) \\ 2(q_1q_2+q_0q_3) & q_0^2-q_1^2+q_2^2-q_3^2 & 2(q_2q_3-q_0q_1) \\ 2(q_1q_3-q_0q_2) & 2(q_2q_3+q_0q_1) & q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix},$$

see [5]. The trajectory of an arbitrary point $\widehat{\mathbf{p}}$ of the moving system is

$$\mathbf{p}(t) = \mathbf{v}(t) + R(t)\,\widehat{\mathbf{p}}. \tag{1}$$

In particular we are interested in *rational spline motions* which are obtained by choosing rational spline (i.e., piecewise rational) functions $q_i(t)$ and $v_i(t)$ representing the coordinates of the quaternion and of the trajectory.

Rational motions can be classified by the degree of their trajectories, which is called the *degree* of the motion, see [6, 7]. In particular, by considering quadratic polynomials $q_i(t)$ one obtains rational motions of degree four or higher. In order to obtain rational motions of degree four, the three functions v_i should be chosen as

$$v_i = \frac{w_i}{q_0^2+q_1^2+q_2^2+q_3^2}, \quad i = 1,2,3, \tag{2}$$

where w_i are quartic polynomials.

Note that there is a remarkable discrepancy in the number of degrees of freedom (i.e., free coefficients) which can be used for specifying the rotation matrix and the trajectory of the origin. The rotation matrix of a quartic motion of this type is controlled by 11 free parameters (4 (number of q_i) $\times 3$ (number of coefficients per quadratic polynomial q_i) -1 (normalization)). On the other hand, the trajectory of

the origin possesses $3 \times 5 = 15$ degrees of freedom. Consequently, the trajectory of the origin is far more flexible than the rotation part.

This discrepancy becomes even larger for motions of higher degree. In order to obtain a motion of degree $2d$, the quaternion can be chosen as a polynomial of degree d with $4(d+1)$ coefficients (where one of them can be eliminated by a normalization). The trajectory of the origin, constructed according to (2), then provides $3(2d+1) = 6d+3$ degrees of freedom.

Consider a rational spline curve $\mathbf{p}(t)$ with domain $I = [0, M]$, where M is the number of segments of the curve, and knots (i.e., segment end points) at the integers. This curve is said to be geometrically smooth of first order (also called G^1 or tangent continuous) if it is continuous (C^0) and satisfies

$$\forall s \in I^\circ \quad \exists \lambda(s) > 0 : \quad \lim_{t \downarrow s} \mathbf{p}'(t) = \lambda(s) \lim_{t \uparrow s} \mathbf{p}'(t),$$

where I° denotes the interior of the parameter interval I. In other words, the curve has a well-defined unit tangent vector everywhere. The proportionality factor λ is equal to 1 everywhere, except for the knots, where it is assumed to be positive.

We say that a rational spline motion is G^1 smooth if all its trajectories (1) are G^1 continuous, where additionally the proportionality factors λ are independent of $\hat{\mathbf{p}}$. This is the case if and only if the curve in 12-dimensional space which is defined by the components of $\mathbf{v}(t)$ and $R(t)$ is G^1 continuous. This curve can be considered as the image of the motion under the kinematic mapping studied in [8]. All point trajectories are obtained as images of this curve under affine mappings.

3 Interpolation Problem

In the remainder of this paper we consider the following geometric interpolation problem in motion design. For $2m+1$ given positions, each of which is represented by the coordinates $\mathbf{C}_j$ of the origin of the coordinate system along with a normalized quaternion $\mathbf{Q}_j$ (considered as fourdimensional vector), $j = 0, \ldots 2m$, find a G^1 smooth rational spline motion which interpolates these positions. Moreover, we assume that every second position (with an even index j) is complemented by associated derivative information $\mathbf{t}_j$ for the motion of the origin and by $\mathbf{u}_j$ for the quaternions. This additional derivative information is either specified by the user or it can be estimated from the data, e.g., using the methods described in [9].

Note that each rotation can be represented by two normalized quaternions, which differ by the sign of their components. We choose the signs so that the standard inner product $\mathbf{Q}_j^\top \mathbf{Q}_{j+1}$ in $\mathbb{R}^4$ is positive.

More precisely, we are looking for seven continuous rational spline functions $\mathbf{v} = (v_1, v_2, v_3)^\top$ and $\mathbf{q} = (q_0, q_1, q_2, q_3)^\top$ with domain $I = [0, 2m]$ which satisfy the position interpolation conditions

$$\mathbf{v}(j) = \mathbf{C}_j, \quad \mathbf{q}(j) = \rho_j \mathbf{Q}_j, \quad j = 0, \ldots, 2m, \tag{3}$$

the tangent interpolation conditions

$$\lim_{t\downarrow 2k}\mathbf{v}'(t)=\sigma_k\,\mathbf{t}_{2k},\quad \lim_{t\downarrow 2k}\mathbf{q}'(t)=\sigma_k\,\rho_{2k}\,\mathbf{u}_{2k},\quad k=0,\ldots,m, \tag{4}$$

where for $t=2m$ the limits from the left must be used, and the G^1 conditions

$$\lim_{t\downarrow j}\mathbf{v}'(t)=\lambda_j\lim_{t\uparrow j}\mathbf{v}'(t),\quad \lim_{t\downarrow j}\mathbf{q}'(t)=\lambda_j\lim_{t\uparrow j}\mathbf{q}'(t),\quad j=1,\ldots,2m-1. \tag{5}$$

The scaling factors ρ_j, σ_k and λ_j are unknown and should all be positive.

It should be noted that the G^1 conditions are sufficient, but not necessary for the motion to be smooth. Indeed, they guarantee that the quaternion curve $\mathbf{q}(t)$ is G^1, but also certain non-G^1-smooth quaternion curves may give tangent continuous rational motions. Currently we are not aware of any sensible use of these additional degrees of freedom, and we therefore decided not to use them.

4 Construction of Quartic Rational Spline Motions

In this section, the technique of parabolic geometric interpolation in $\mathbb{R}^3$ will be used to construct quartic rational spline motions, see [10, 11] for the related discussion in the space of curves. Since geometric interpolation schemes depend on the dimension of the space, the quaternion data will be transformed to a particular three-dimensional subspace, without changing the geometric meaning.

Consider first the case with only two segments, i.e., $m=1$. We start with the construction of the spherical part of the motion. Recall that proportional quaternions represent the same rotation. We choose $\rho_0=\rho_2=1$. By the QR decomposition

$$\left[\mathbf{u}_0,\,\mathbf{Q}_2-\mathbf{Q}_0,\,\mathbf{u}_2\right]=F\,U,\quad F:=\left[\mathbf{f}_1,\,\mathbf{f}_2,\,\mathbf{f}_3\right]\in\mathbb{R}^{4\times 3},\quad U\in\mathbb{R}^{3\times 3},$$

where $F^\top F=I$ and U is an upper triangular matrix, we obtain an orthonormal basis $\{\mathbf{f}_i\}_{i=1}^3$ of the threedimensional hyperplane $\mathscr{S}$ spanned by $\mathbf{u}_0$, $\mathbf{Q}_2-\mathbf{Q}_0$ and $\mathbf{u}_2$.

In order to work in this threedimensional hyperplane, we must choose ρ_1 so that $\rho_1\,\mathbf{Q}_1-\mathbf{Q}_0\in\mathscr{S}$ too. If $\mathbf{Q}_1-\mathbf{Q}_0$ is already in $\mathscr{S}$, then $\rho_1=1$, otherwise

$$\rho_1=\frac{\det(\mathbf{f}_1,\mathbf{f}_2,\mathbf{f}_3,\mathbf{Q}_0)}{\det(\mathbf{f}_1,\mathbf{f}_2,\mathbf{f}_3,\mathbf{Q}_1)}. \tag{6}$$

Obviously ρ_1 will be positive if and only if both determinants in (6) have the same sign.

In the next step, we construct a parabolic G^1 polynomial spline $\mathbf{q}$ with knots $0,1,2$ satisfying

$$\mathbf{q}(2j)=\mathbf{Q}_{2j},\quad \mathbf{q}'(2j)=\sigma_j\,\mathbf{u}_{2j},\quad j=0,1, \tag{7}$$

and

$$\mathbf{q}(1) = \rho_1 \mathbf{Q}_1, \quad \lim_{t\downarrow 1} \mathbf{q}'(t) = \lambda_1 \lim_{t\uparrow 1} \mathbf{q}'(t). \tag{8}$$

Clearly, the standard parabolic interpolation scheme in $\mathbb{R}^3$ does not provide enough freedom to solve this problem, thus the geometric interpolation technique must be applied (for general geometric methods see e.g. [1]). Let $\mathbf{q}_j := \mathbf{q}|_{[j,j+1]}$, $j = 0, 1$, be the quadratic polynomial pieces of $\mathbf{q}$ (segments of parabolas). They can be written in Bernstein–Bézier form as

$$\mathbf{q}_0(t) := \mathbf{Q}_0 B_{2,0}(t) + \mathbf{Q}_{01} B_{2,1}(t) + \rho_1 \mathbf{Q}_1 B_{2,2}(t), \tag{9}$$

$$\mathbf{q}_1(t) := \rho_1 \mathbf{Q}_1 B_{2,0}(t-1) + \mathbf{Q}_{12} B_{2,1}(t-1) + \mathbf{Q}_2 B_{2,2}(t-1), \tag{10}$$

where $B_{n,i}(t) = \binom{n}{i} t^i (1-t)^{n-i}$ are the Bernstein basis polynomials of degree n. Basic properties of Bézier curves (see e.g. [3]) imply

$$\mathbf{q}_0'(0) = 2(\mathbf{Q}_{01} - \mathbf{Q}_0), \quad \mathbf{q}_1'(2) = 2(\mathbf{Q}_2 - \mathbf{Q}_{12}).$$

Now (9) and (10), along with the interpolation conditions (7), lead to

$$\mathbf{Q}_{01} = \mathbf{Q}_0 + \frac{1}{2}\sigma_0 \mathbf{u}_0, \quad \mathbf{Q}_{12} = \mathbf{Q}_2 - \frac{1}{2}\sigma_2 \mathbf{u}_2. \tag{11}$$

The G^1 condition (8) can be rewritten as

$$\mathbf{Q}_{12} - \rho_1 \mathbf{Q}_1 = \lambda_1(\rho_1 \mathbf{Q}_1 - \mathbf{Q}_{01})$$

and (11) finally gives the nonlinear system

$$-\frac{1}{2}\lambda_1 \sigma_0 \mathbf{u}_0 + \lambda_1 \Delta\mathbf{Q}_0 + \frac{1}{2}\sigma_2 \mathbf{u}_2 = \Delta\mathbf{Q}_1$$

for $\sigma_0, \sigma_1, \lambda_1$, where $\Delta\mathbf{Q}_0 := \rho_1 \mathbf{Q}_1 - \mathbf{Q}_0$ and $\Delta\mathbf{Q}_1 := \mathbf{Q}_2 - \rho_1 \mathbf{Q}_1$. Fortunately, this system can be transformed into to the linear form

$$F^\top \left[-\tfrac{1}{2}\mathbf{u}_0,\ \Delta\mathbf{Q}_0,\ \tfrac{1}{2}\mathbf{u}_2\right] \begin{bmatrix} \lambda_1 \sigma_0 \\ \lambda_1 \\ \sigma_2 \end{bmatrix} = F^\top \Delta\mathbf{Q}_1,$$

and Cramer's rule yields

$$\sigma_0 = 2\frac{D_1}{D_2}, \quad \sigma_2 = 2\frac{D_4}{D_3}, \quad \lambda_1 = \frac{D_2}{D_3}, \tag{12}$$

where $D := F^\top \left[\mathbf{u}_0, \Delta\mathbf{Q}_0, \Delta\mathbf{Q}_1, \mathbf{u}_2\right]$, and D_i is the determinant of the submatrix of D with i-th column omitted.

The sufficient and necessary conditions for σ_0, σ_2 and λ_1 to be positive is that all determinants D_i, $i = 1, 2, 3, 4$, have the same sign.

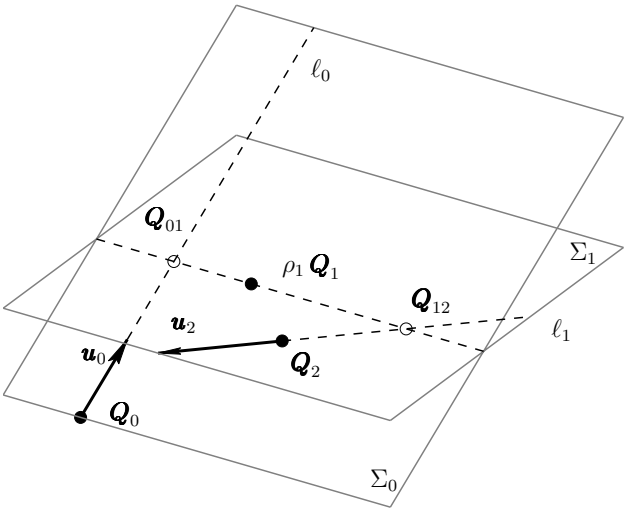


Fig. 1 Geometric construction of control points $\mathbf{Q}_{01}$ and $\mathbf{Q}_{12}$ given by (11).

There is also a simple geometric interpretation of this solution (see Figure 1). Let Σ_0 be the plane spanned by $\rho_1 \mathbf{Q}_1$ and the line ℓ_0 through $\mathbf{Q}_0$ in the direction of $\mathbf{u}_0$. Further, let Σ_1 be another plane defined by $\rho_1 \mathbf{Q}_1$ and the line ℓ_1 through $\mathbf{Q}_2$ in the direction of $\mathbf{u}_2$. This two planes determine the line which intersects ℓ_0 and ℓ_1 in $\mathbf{Q}_{01}$ and $\mathbf{Q}_{12}$, respectively. The solution is admissible if this two intersections lie on the proper sides of the points $\mathbf{Q}_0$ and $\mathbf{Q}_2$.

We are now left to construct the trajectory $\mathbf{v}$ of the origin of the moving coordinate system. Let $\mathbf{v}$ be composed of two quartic rational spline curves $\mathbf{v}_j$,

$$\mathbf{v}_j := \mathbf{v}|_{[j,j+1]}, \quad \mathbf{v}_j = \frac{\mathbf{w}_j}{r_j}, \quad r_j := \|\mathbf{q}_j\|^2, \quad j = 0,1,$$

where $\mathbf{w}_0$, $\mathbf{w}_1$ are the unknown polynomial curves of degree ≤ 4, and $\mathbf{q}_0, \mathbf{q}_1$ are obtained from the interpolation of the rotational part. Note that the parameters $\sigma_0, \sigma_1, \lambda_1$ are already fixed by (12). The interpolation conditions (3), (4) and (5) clearly do not provide sufficiently many equations to uniquely determine $\mathbf{w}_0, \mathbf{w}_1$. One way to use the additional freedom is to to restrict $\mathbf{w}_j$ to a cubic polynomial and prescribe also the tangent direction $\mathbf{t}_1$ at $\mathbf{C}_1$. Then a standard cubic interpolation scheme can be applied to determine $\mathbf{w}_j$, $j = 0,1$, that satisfy

$$\mathbf{w}_j(k) = r_j(k)\,\mathbf{C}_k, \quad k = j, j+1, \quad j = 0,1,$$

and

$$\begin{aligned} &\mathbf{w}_0'(0) = r_0(0)\,\sigma_0\,\mathbf{t}_0 + r_0'(0)\,\mathbf{C}_0, \quad \mathbf{w}_0'(1) = r_0(1)\,\mathbf{t}_1 + r_0'(1)\,\mathbf{C}_1, \\ &\mathbf{w}_1'(1) = \lambda_1\, r_1(1)\,\mathbf{t}_1 + r_1'(1)\,\mathbf{C}_1, \quad \mathbf{w}_1'(2) = \sigma_2\, r_1(2)\,\mathbf{t}_2 + r_1'(2)\,\mathbf{C}_2. \end{aligned}$$

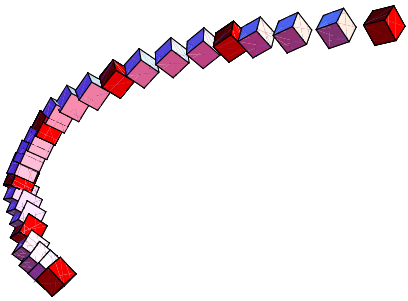


Fig. 2 Motion of a cube with seven interpolated positions.

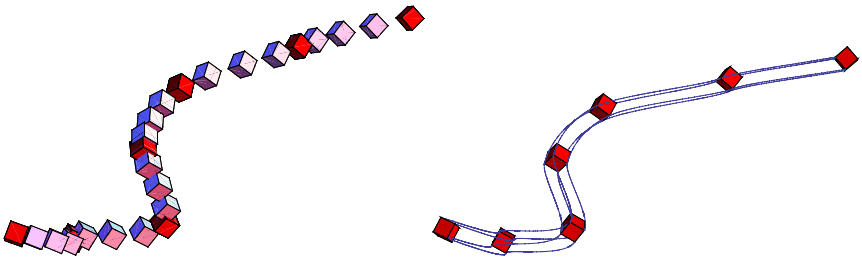


Fig. 3 Motion of a cube with seven interpolated positions (left) and the trajectories of corners of one side of the cube (right).

The presented scheme for $m = 1$ is clearly entirely local. Consequently, for $m > 1$, it can be applied on consecutive pairs of segments. This gives a G^1 rational quartic spline motion.

Let us demonstrate the performance of the presented interpolation scheme by some numerical examples.

Figure 2 shows the motion of a cube with seven interpolated positions. The input data were the positions of the center and the unit quaternions which corresponds to the rotations. The directions $\mathbf{t}_j$ were estimated using local parabolas (Bessel scheme), and the quaternion directions $\mathbf{u}_j$ were obtained as proposed in [9]. Figure 3 shows another motion of a cube together with the trajectory of corners of one side of the cube.

5 Conclusion

This paper was devoted to piecewise rational motions with first order geometric continuity. As an application of these motions we described an interpolation scheme

which generates rational spline motions of degree four from a sequence of given positions, some of which are complemented by associated tangent information. As demonstrated by the results, the use of geometric interpolation makes it possible to deal successfully with the uneven distribution of the degrees of freedom between the trajectory of the origin and the rotation part of the rational motions.

Future work will focus on the asymptotic behaviour of the interpolation scheme, where we plan to analyze the relation between the sampling density (i.e., the distance between the positions which are to be interpolated) and the behaviour and existence of the interpolating motion, and on the extension of these concepts to motions with higher order of smoothness.

References

1. Mørken, K., Scherer, K.: A general framework for high-accuracy parametric interpolation. *Math. Comp.*, **66**(217), 237–260 (1997).
2. de Boor, C., Höllig, K., Sabin, M.: High accuracy geometric Hermite interpolation. *Comput. Aided Geom. Design*, **4**(4), 269–278 (1987).
3. Hoschek, J., Lasser D.: *Fundamentals of Computer Aided Geometric Design*. AK Peters, Wellesley, MA (1993).
4. Schröcker, H.-P., Jüttler, B.: Motion interpolation with Bennett biarcs. In Kecskemethy, A., Müller, A. (Eds.), *Proc. Computational Kinematics A*. Springer, pp. 101–108 (2009).
5. Bottema, O., Roth, B.: *Theoretical Kinematics*. Dover (1990).
6. Jüttler, B., Wagner, M.G.: Kinematics and animation. In: G. Farin, J. Hoschek, and M.-S. Kim (Eds.), *Handbook of Computer Aided Geometric Design*, pp. 723–748. Elsevier, Amsterdam (2002).
7. Röschel, O.: Rational motion design – A survey. *Comput. Aided Design* **3**0(3), 169–178 (1998).
8. Rath, W.: A kinematic mapping for projective and affine motions and some applications. In: Dillen, F. et al. (Eds.), *Geometry and topology of submanifolds, VIII*. World Scientific, pp. 292–301 (1996).
9. Horsch, T., Jüttler, B.: Cartesian spline interpolation for industrial robots. *Comput. Aided Design*, **30**, 217–224 (1998).
10. Mørken, K.: Parametric interpolation by quadratic polynomials in the plane. In: *Mathematical Methods for Curves and Surfaces*. Vanderbilt Univ. Press, Nashville, pp. 385-402 (1995).
11. Kozak, J., Žagar, E.: On geometric interpolation by polynomial curves. *SIAM J. Numer. Anal.*, **42**(3), 953–967 (2004).

Position Level Kinematics of the Atlas Motion Platform

R. Beranek and M.J.D. Hayes

Department of Mechanical and Aerospace Engineering, Carleton University, Canada; e-mail: jhayes@mae.carleton.ca; rberanek@connect.carleton.ca

Abstract. Atlas is a novel six degree of freedom vehicle simulator motion platform where orienting is decoupled from positioning, and unbounded rotation is possible about any axis. Angular displacements are achieved by manipulating the spherical exterior of the cockpit with three omniwheel actuators. A significant challenge to practical implementation of the design is dynamic slip at each omniwheel-sphere interface. The dynamic slip renders the velocity level constraints nonholonomic, in turn meaning that the position level kinematics are undefined. This paper proposes a numerical integration algorithm to provide an estimate of the platform orientation. The algorithm is based on solving the associated quaternionic differential equation given constant omniwheel angular rates. For sufficiently small time intervals of changing omniwheel rates, the algorithm can be applied recursively to estimate the sphere position level kinematics given omniwheel angular velocity as input. Experimental results suggest that dynamic slip may be identified and compensated.

Key words: Unbounded angular displacement, position and velocity level kinematics, quaternionic differential equation. nonholonomic constraints.

1 Introduction

The Atlas motion platform [1] was introduced as a practical alternative to the Stewart–Gough hexapod [2, 3], used principally for motion simulator platforms. A table top technology proof-of-concept demonstrator is illustrated in Fig. 1. The Atlas concept consists of a cockpit encased in a sphere which rests on three omnidirectional wheels. The three omnidirectional wheels are arranged on the edges of an equilateral triangle giving an angular separation of 120° in the XY-plane, see Fig. 2(b). The elevation angle of each omnidirectional wheel relative to the XY-plane is 40°. The reason for the equilateral configuration is to achieve even force and torque distribution on the omnidirectional wheels, however the elevation angle of 40° was selected for ease of manufacturing and assembly. The sphere/omniwheel assembly is connected to three independent linear motion stages. The omniwheels control the orientation of the sphere, while the linear stages provide for the translation of the platform along all three linear axes. This allows for a full six degree of freedom (DOF) motion with unbounded rotation about any axis.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_41, © Springer Science+Business Media B.V. 2010

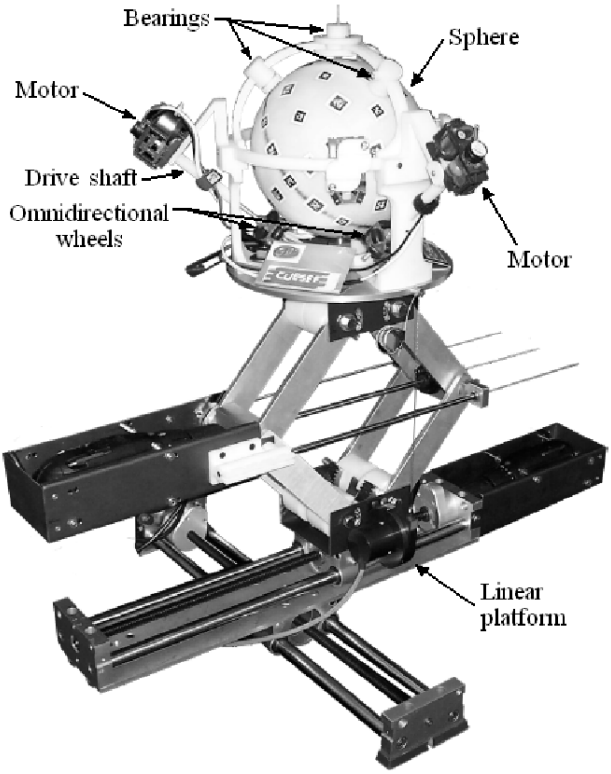


Fig. 1 The Atlas table-top 6-DOF demonstrator highlighting the omnidirectional wheel actuation concept.

The Atlas motion platform is not unique in its ability to provide unbounded angular displacement. For example, the Eclipse II architecture [4] possesses this ability. However, uniqueness of the Atlas platform lies in its kinematic model, which leads to remarkably simple, closed form velocity closure equations. Moreover, its orientation workspace is not constrained by structural interferences, or rotation limits of the spherical joints, compared to the Eclipse II.

One of the principal challenges for the Atlas concept is that the kinematics have so far only been defined at the velocity level [5]. The position level kinematics are undefined because the velocity constraints are nonholonomic due to the presence of dynamic slip at the omniwheel-sphere interface [6]. This paper presents an approach to estimate the Atlas platform orientation, starting from the velocity level kinematics derived in [5]. The approach consists of integrating the quaternionic differential equation [7,8] assuming constant omniwheel angular velocity inputs. Measurements obtained from experiments yield information on how well the the quaternion solution estimates the orientation of the sphere for constant omniwheel speeds. A simple

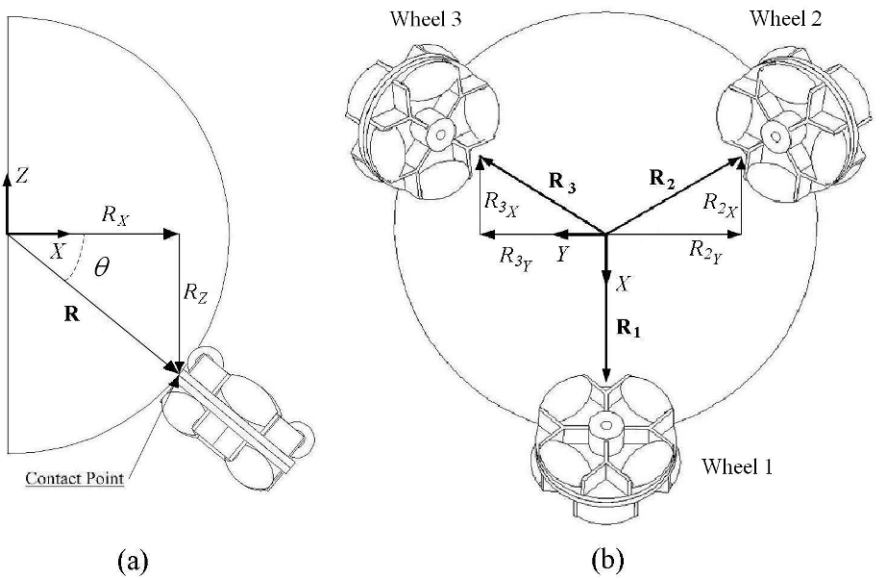


Fig. 2 Configuration of the original Atlas spherical platform: (a) front view; (b) bottom view.

recursive algorithm for estimating orientation given nonconstant velocity inputs is also put forward.

2 Atlas Velocity Level Kinematic Model

The velocity level kinematic model of the Atlas platform is derived in detail in [5], and is briefly summarized in what follows. The translational displacements generated with the XYZ linear platform are completely decoupled from the rotational displacements of the sphere. Modelling the linear velocity of the geometric centre of the sphere is straightforward and typically represented as a simple linear term which must be added to the more demanding spherical kinematic model. Therefore, without loss in generality, only the spherical kinematics need be considered here.

It is convenient to perform velocity analysis of a manipulator with its Jacobian. It is, by definition, a mapping between time rates of change. By convention, in robotics it is the mapping between the time rates of change of the joint variables to the time rates of change of the position and orientation of the end effector [9].

Changes in orientation of the Atlas motion platform are achieved with three active omnidirectional wheels arranged on the edges of an equilateral triangle giving an angular separation of 120° in the XY-plane, see Fig. 2(b). The elevation angle of each omnidirectional wheel relative to the XY-plane is $\theta = 40°$. The radius of the

table top demonstrator sphere is $R = 10.16$ cm, while the radius of each omniwheel is $r = 2.00$ cm.

For this configuration of omnidirectional wheels the resulting mapping between the velocities in *joint space* and those in *Cartesian space* is:

$$\mathbf{\Omega} = \mathbf{J}\boldsymbol{\omega} = \frac{r}{3R}\begin{bmatrix} -2\csc\theta & \csc\theta & \csc\theta \\ 0 & \sqrt{3}\csc\theta & -\sqrt{3}\csc\theta \\ -\sec\theta & -\sec\theta & -\sec\theta \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}, \tag{1}$$

where $\mathbf{\Omega}$ is the angular velocity vector of the sphere expressed in the inertial sphere coordinate system illustrated in Fig. 2, $\mathbf{J}$ is the Jacobian of the manipulator, and $\boldsymbol{\omega}$ is the array of angular rates of the three actuating omniwheels, r represents the radius of the omniwheel (assuming all three to be equal), R is the external radius of the sphere, and θ is the elevation angle of each omniwheel. Substituting the numerical values for r, R, and θ leads to

$$\mathbf{J} = \begin{bmatrix} -0.2042 & 0.1021 & 0.1021 \\ 0 & 0.1768 & -0.1768 \\ -0.0856 & -0.0856 & -0.0856 \end{bmatrix}. \tag{2}$$

Inspection of the system Jacobian expressed by either Eq. (1) or Eq. (2) reveals that, unlike typical manipulator Jacobians, $\mathbf{J}$ is time invariant and depends only on design constants. These constants can be chosen such that the Jacobian will possess full rank and that the orienting workspace of the sphere is configurationally singularity free. Moreover, because the sphere can have any orientation about any point within reach of the sphere centre, the reachable workspace is fully dexterous.

Because the Jacobian of the system is time invariant and constant, once the configuration has been determined, acceleration-level kinematics can be obtained by simple differentiation of the expression, yielding:

$$\dot{\mathbf{\Omega}} = \mathbf{J}\dot{\boldsymbol{\omega}}. \tag{3}$$

3 Atlas Position Level Kinematic Model

Obtaining the expression for the orientation of the platform, however, is not as simple. In this work quaternions are employed because the unbounded and singularity-free nature of the design calls for a singularity-free representation. Integration of the quaternionic differential equation is required [8]:

$$\dot{q} = \frac{1}{2}\Omega \circ q, \tag{4}$$

where q is the unit quaternion describing the orientation of the system, and $\Omega \circ q$ is a quaternion product.

The quaternion product can be expressed as a matrix product [7]:

$$\frac{d\mathbf{q}}{dt} = \mathbf{F}_q(\boldsymbol{\Omega})\mathbf{q}, \tag{5}$$

where $\mathbf{F}_q(\boldsymbol{\Omega})$ is a skew symmetric matrix of the sphere angular velocities expressed in the sphere inertial coordinate system, and is defined to be:

$$\mathbf{F}_q(\boldsymbol{\Omega}) = \frac{1}{2}\begin{bmatrix} 0 & -\Omega_x & -\Omega_y & -\Omega_z \\ \Omega_x & 0 & \Omega_z & -\Omega_y \\ \Omega_y & -\Omega_z & 0 & \Omega_x \\ -\Omega_z & -\Omega_y & -\Omega_x & 0 \end{bmatrix}. \tag{6}$$

Equation (5) yields a set of four simple ordinary differential equations. If the angular velocity $\boldsymbol{\Omega}$ is constant and the initial conditions $\mathbf{q}(t_0)$ are known, then the solution to Eq. (6) can be written as [10]

$$\mathbf{q}(t) = \boldsymbol{\phi}_q(t_0, t, \boldsymbol{\Omega})\mathbf{q}(t_0), \tag{7}$$

such that the transition matrix is

$$\boldsymbol{\phi}_q(t_0, t, \boldsymbol{\Omega}) = e^{\mathbf{F}_q(\boldsymbol{\Omega})\Delta t} = \cos(\|\boldsymbol{\Omega}\|\Delta t/2)\mathbf{I} + 2\frac{\sin(\|\boldsymbol{\Omega}\|\Delta t/2)}{\|\boldsymbol{\Omega}\|}\mathbf{F}_q(\boldsymbol{\Omega}), \tag{8}$$

with $\Delta t = t - t_0$, and $\|\boldsymbol{\Omega}\| = \sqrt{\Omega_x^2 + \Omega_y^2 + \Omega_z^2}$.

The solution represented by Eq. (7) can be expressed as the quaternion product

$$\mathbf{q}(t) = \mathbf{q}(t_0) \circ \boldsymbol{\phi}_q(t_0, t, \boldsymbol{\Omega}), \tag{9}$$

or

$$\mathbf{q}(t) = \begin{bmatrix} q(t_0)_1 & -q(t_0)_2 & -q(t_0)_3 & -q(t_0)_4 \\ q(t_0)_2 & q(t_0)_1 & -q(t_0)_4 & q(t_0)_3 \\ q(t_0)_3 & q(t_0)_4 & q(t_0)_1 & -q(t_0)_2 \\ q(t_0)_4 & -q(t_0)_3 & q(t_0)_2 & q(t_0)_1 \end{bmatrix} \begin{bmatrix} \cos\frac{\|\boldsymbol{\Omega}\|t}{2} \\ \frac{\Omega_x}{\|\boldsymbol{\Omega}\|}\sin\frac{\|\boldsymbol{\Omega}\|t}{2} \\ \frac{\Omega_y}{\|\boldsymbol{\Omega}\|}\sin\frac{\|\boldsymbol{\Omega}\|t}{2} \\ \frac{\Omega_z}{\|\boldsymbol{\Omega}\|}\sin\frac{\|\boldsymbol{\Omega}\|t}{2} \end{bmatrix}. \tag{10}$$

Given the magnitudes of the three omniwheel angular rates, the corresponding angular velocity of the sphere is determined using Eq. (1). The subsequent orientation of the sphere at any time t after an initial time t_0 is then estimated using Eq. (10), ignoring the effects of dynamic slip.

4 Experimental Validation

To validate the solution provided by integrating the quaternionic differential equation, several simple cases were examined where omniwheel constant angular rates were specified and the corresponding sphere angular velocity was measured. For the

first three, each angular velocity was selected such that it caused the sphere to spin about one of its inertial axes according to Eq. (1). The resulting motion of the table top demonstrator (Fig. 1) observed appeared to be consistent with the model.

To validate Eq. (10) an arbitrary axis for the rotation of the sphere was selected using arbitrary constant angular velocities for each omniwheel. The sphere motion predicted by Eq. (10) was compared to motion data recorded. The measurement system consists of a three axis gyroscope (MicroStrain 3DM-GX1) mounted inside the sphere and a camera-based external motion system that tracks the relative displacement of markers on the surface of the sphere, see Fig. 3. Ideally, the nondeterministic gyroscope drift is zeroed with data from the external camera-based system and the internal sensor data are fused using unscented Kalman filter techniques [11]. However, the vision system is not functioning reliably yet and only gyroscope data is currently available.

Several experiments were run where omniwheel angular rates were specified and the resulting motion of the sphere was tracked and recorded as output from the three axis gyroscope. The time history of the angular displacements about the sphere inertial axes that was predicted is in the same ball park as those measured for all runs. Results, comparing predicted and measured orientation about the inertial coordinate system, from a run where the three omniwheel angular velocities were prescribed as $[\omega_1, \omega_2, \omega_3] = [5, -5, 5]$ rads/s are illustrated in Fig. 4. The relatively large tracking drift in yaw, pitch, and roll is likely due to the nondeterministic drift of the gyroscopes. However, it may also be partly due to random errors associated with dynamic slip and variation in omniwheel angular velocities. Nevertheless, the results suggests that position level kinematics modelled by Eq. (10) are representative of the physical system if the dynamic slip is identified with the measurement system and compensated.

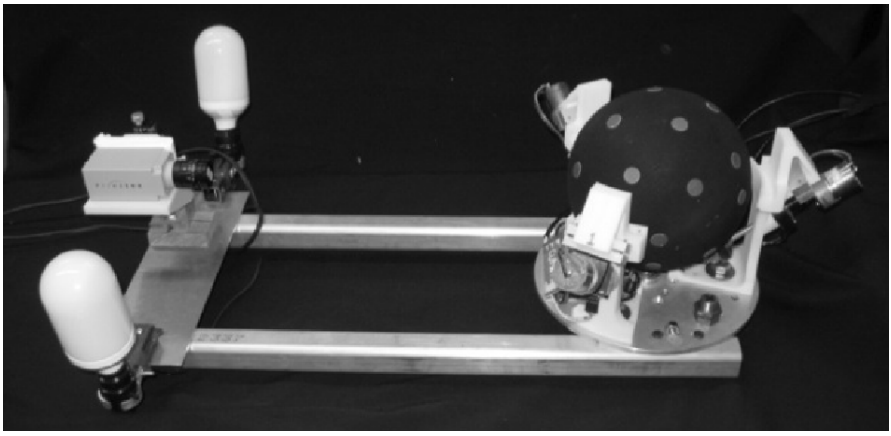


Fig. 3 Sphere orientation measurement system.

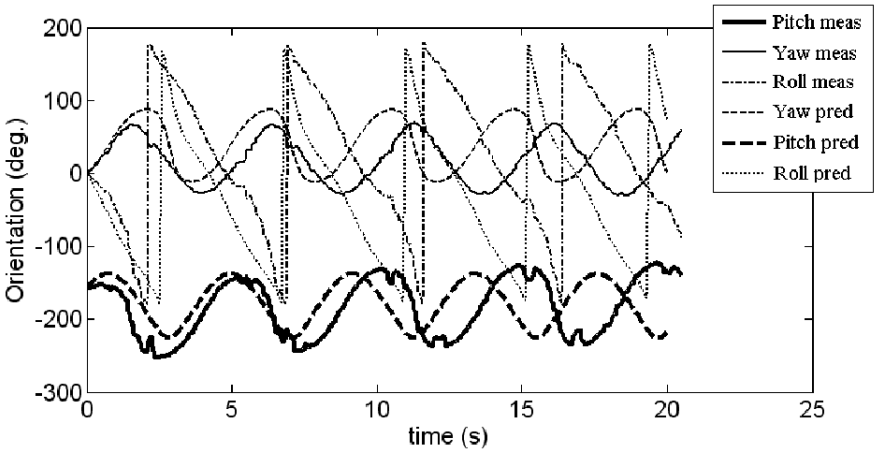


Fig. 4 Predicted and measured sphere angular velocities.

5 Orientation Estimate for Nonconstant Velocity Inputs

The solution determined with the integration of the quaternionic differential equation is only valid for constant omniwheel angular velocities. Regardless, this solution can be used in a numerical integration algorithm to provide an estimate of the sphere orientation for a general trajectory. The relation expressed by Eq. (9) can be used to construct a recursive estimate. Consider

$$\mathbf{q}(k) = \mathbf{q}(k-1) \circ \boldsymbol{\phi}_q(\Delta t, \boldsymbol{\omega}(k-1)), \tag{11}$$

where k is a time index and Δt is the time interval between omniwheel velocity measurements.

The solution is therefore a recursive estimate where the orientation is an extrapolation given the previous angular velocities of the omniwheels and previous orientation at time $k-1$. For sufficiently small Δt, the assumption that the motion between time steps is linear may be used and a reasonable estimate of the sphere orientation can be obtained for any general set of omniwheel angular velocities over an arbitrary trajectory.

6 Conclusions

Because of the presence of dynamic slip at each sphere-omniwheel interface, the velocity level kinematic constraints of the Atlas sphere are nonholonomic. The orientation of the sphere at any time t given constant omniwheel angular velocity inputs can be predicted using the solution to a quaternion based differential equation. The

predicted time history of the change in orientation of the sphere possesses the same tendencies as the measured change in orientation, however measured values tend to drift, and there appears to be some bias about some inertial coordinate system axes. The drift may give an indication of error imposed by the dynamic slip, and could possibly be used for error correction, however the gyroscope drift must first be zeroed.

This solution can also be adapted to a numerical integration algorithm which can effectively be used as a general solution to the platform kinematics, providing an estimate of the platform orientation for any set of varying omniwheel input. This may allow for a good estimate of the orientation that could, in combination with the measurement system, be used in the control algorithm for the platform. Future work will focus on experimental validation of the constant angular velocity solution as well as validation of the proposed numerical solution.

References

1. M.J.D. Hayes, R.G. Langlois: Atlas: A novel kinematic architecture for six DOF motion platforms. *Transactions of the Canadian Society for Mechanical Engineering*, **29**(4), 701–709 (2005).
2. V.E. Gough: Contribution to discussion of papers on research in automobile stability, control and tyre performance. In: *Proc of Auto Div. Inst. Mech. Eng.*, pp. 392–394 (1956).
3. D. Stewart: A platform with 6 degrees of freedom. *Proc. of the Institution of Mechanical Engineers, Part 1*, **180**(15), 371–378 (1965).
4. J. Kim, J.C. Hwang, J.S. Jim, C.C. Iurascu, F.C. Park, Y.M. Cho: Eclipse II: A new parallel mechanism enabling continuous 360-degree spinning plus three-axis translational motions. *IEEE Transactions on Robotics and Automation*, **18**(3), 367–373 (2002).
5. M.J.D. Hayes, R.G. Langlois, A. Weiss: Atlas motion platform generalized kinematic model. In: *Proceedings of the Second International Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators*, Montpellier, France, pp. 227–234 (2008).
6. J.H. Ginsberg: *Advanced Engineering Dynamics*, 2nd edition. Cambridge University Press (1995).
7. S. Särkkä: Notes on quaternions. Internal Technical Document, Helsinki University of Technology (2007). http://www.lce.hut.fi/∼ssarkka/. Cited 15 January 2010.
8. A.L. Schwab, J.P. Meijaard: How to draw Euler angles and utilize Euler parameters. In: *Proceedings of ASME IDETC/CIE* (2006).
9. J.C. Craig: *Introduction to Robotics*, 2nd edition. Addison Wesley (1989).
10. D. Zwillinger: *Handbook of Differential Equations*, 3rd edition. Academic Press (1998).
11. J. Linseman: A UKF-based orientation estimator for the Atlas platform. M.A.Sc. Thesis, Dept. of Mech. & Aero. Eng., Carleton University, Ottawa, ON, Canada (2010).

An Optimum Path Planning for LARM Clutched Arm

Hao Gu and Marco Ceccarelli

LARM: Laboratory of Robotics and Mechatronics, DiMSAT, University of Cassino, Cassino (Fr), Italy; e-mail: {hao.gu, ceccarelli}@unicas.it

Abstract. A path planning for the LARM clutched arm is presented. As a low-cost easy-operation solution for humanoid robots, the LARM clutched arm is a novel robotic arm with only one actuator and a 3D workspace. An algorithm is proposed for path planning of the arm motion with multi-objective criteria that takes into account the clutch and motor controlled motion capability. Numerical examples are reported to show the feasibility of the proposed path planning and the efficiency of the LARM clutched arm.

Key words: Humanoid robots, robotic arms, clutch systems, path planning.

1 Introduction

Nowadays, most of the robotic arms are designed as multi-actuator systems to mimic the DOFs of human arms. They act quite flexibly and human likely, but they are very expensive and complicated systems for no expert users. It is possible to design robotic arms with less DOFs in order to reduce the cost and complexity of robotic arms. Reduced number of actuators can be achieved with specific alternative design, as the nonholonomic manipulators [1], or by using clutches, as attempted in [2].

Recently, a new robotic arm that has been named as LARM clutched arm, has been proposed in [3, 4]. Karbasi's design in [2] is a modular drive system as a trasmission solution rather than a robot system as it is the case for LARM arm, in which all the clutches and actuator have been integrated in a small sized shoulder mechanism for humanoid robots. The key functionality of the clutches in the mechanical design of LARM design is aimed to transmit a controlled rotation from the actuator, while Karbasi's design is aimed to regulate the output velocity with a constant input rotation of the actuator by PWM controllers.

In this paper, the problem of path planning is formulated as based on a proper operation for the actuator and clutches to obtain an optimum arm movement. The proposed path planning algorithm takes into account the discrete motion features of LARM clutched arm, which is different from the path planning of traditional robot arms.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_42,

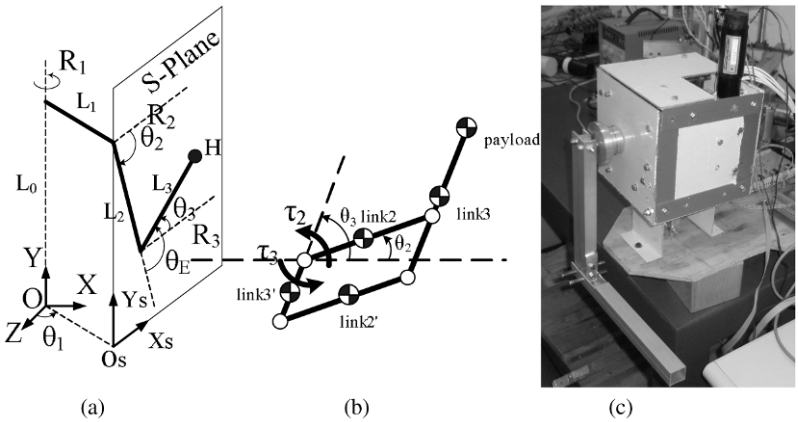


Fig. 1 The proposed LARM clutched arm: (a) scheme; (b) dynamic mode; (c) prototype.

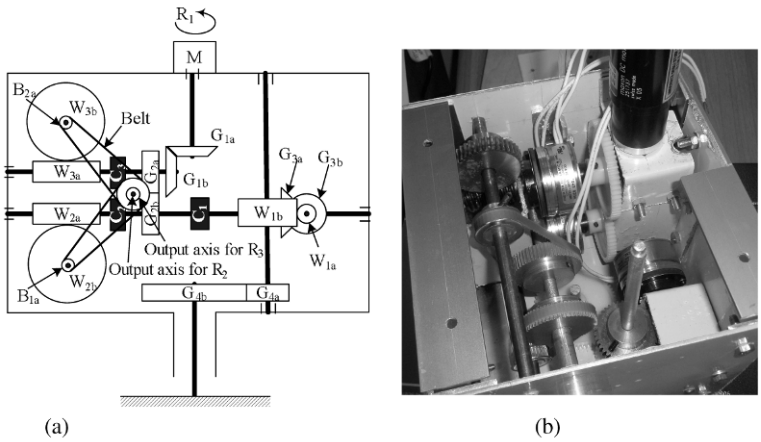


Fig. 2 The clutch system in the arm shoulder: (a) scheme; (b) prototype.

2 The LARM Clutched Arm

As shown in Fig. 1, the proposed new robotic arm has only one motor, but it can move in a 3D space. The three rotations of the arm motion have been defined as R_1 to R_3 respectively. θ_1 to θ_3 are the corresponding joint angles. L_i ($i = 0$–4) are the link lengths of the arm structure. It is a gearing system consisting of spur gears, bevel gears, belt wheels, worms and worm wheels. For details, see [1]. Electromagnetic clutches C_i ($i = 1$–3), activate the gear train for R_1, R_2 and R_3 rotations, respectively. The self-locking feature of worm couplings is used for locking the rotations when clutches are released.

Table 1 The operation modes for LARM clutched arm.

Operation Mode	Active Rotation	C_1	C_2	C_3
OP0	none	0	0	0
OP1	R_1	1	0	0
OP2	R_2	0	1	0
OP3	R_3	0	0	1
OP4	$R_1 \& R_2$	1	1	0
OP5	$R_1 \& R_3$	1	0	1
OP6	$R_2 \& R_3$	0	1	1
OP7	$R_1 \& R_2 \& R_3$	1	1	1

Table 1 lists the possible operation modes for the arm, where "0" means that a clutch is disconnected and "1" a clutch is connected. OP1 to OP7 can give positive and negative motions as depending on the rotating direction of the motor. Consequently, in the path planning, operation modes can be indicated as OP $= +i$ ($i = 0$ to 7), where the sign refers to the direction of the corresponding joint angles.

Thus, LARM clutched arm has the following novel characteristics:

- light-weight limb design, with low cost, low energy consumption, and easy-operation;
- 1-DOF clutched design with seven operation modes in a 3D space;
- avoiding overheat of motor, since the motor works only for actuating the arm but not for maintaining its posture once the arm stops.

3 Formulation for Path Planning

As a 1-DOF solution, the LARM clutched arm moves discretely. The seven operation modes $OP1$ to $OP7$ in Table 1 can be expressed as simple vectors to plan motions in Joint-Space.

The kinematic model of LARM clutched arm can be considered as a two-link planar manipulator rotating around the Y axis with the radial distance of L_1, Fig. 1(a). The rotating plane for upper arm and forearm is defined as S-plane with coordinate frame X_sY_s as shown in Fig. 1(a). The coordinates of the manipulator endpoint H can be formulated by using expressions as function of the kinematic variables that require the FKP (Forward Kinematic Problem) and the IKP (Inverse Kinematic Problem) for LARM clutched arm to be solved. Referring to the motion range of the prototype, the LARM clutched arm has the following constraints as

$$\theta_1 \in [-50\,\text{deg}, 130\,\text{deg}], \theta_2 \in [-150\,\text{deg}, 90\,\text{deg}], \theta_E \in [5\,\text{deg}, 150\,\text{deg}], \quad (1)$$

in which $\theta_E = \theta_3 - \theta_2$ is the angle for elbow as shown in Fig. 1(a).

Figure 3 illustrates general characteristics for the movement of the LARM clutched arm. The prescribed task is given by the start and end points of the tra-

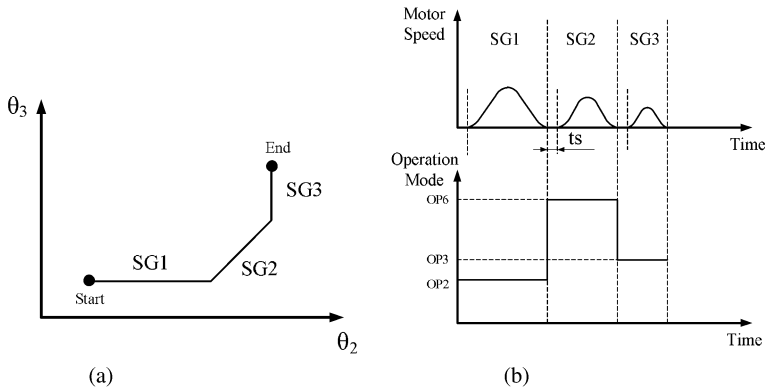


Fig. 3 Characteristics of the movement of LARM clutched arm.

jectory. Traditionally, the path planning task for a manipulator with nDOFs can be described using n knots in the trajectory of each kth joint of the manipulator [5]. However, for the LARM clutched arm, the traditional way is not suitable, since it has only one actuator.

In Fig. 3, the arm moves from start point to end point in three segments (SG1-SG3). Used operation modes for this path are OP2, OP6 and OP3 sequentially as shown in Fig. 3(b). When a certain operation mode has been set, the single motor works to output a desired angular displacement for one segment. ts in Fig. 3(b) is the so-called mode setting time for each segment, which is necessary to make sure clutches have been set into the desired states before the motor runs. In order to have a smooth motion for each segment, the motor is controlled with a cubic function for angular speed.

Therefore, the path planning task for the LARM clutched arm is to seek a proper sequence of segments with certain operation modes and angular displacements. Consequently, variables of the path planning for LARM clutched arm are: OP, operation mode; D, angular displacement; n, number of segment.

In order to reduce time and energy consumption in a path planning for the LARM clutched arm minimum travelling time and minimum mechanical energy of actuator can be considered together in a multi-objective function as

$$\begin{aligned} &\min \quad f = w_1 T_{tra} + w_2 W_{act} \\ &\text{subject to constraints in Eq. (1),} \end{aligned} \tag{2}$$

where T_{tra} is the travelling time and W_{act} is the mechanical energy of actuator, w_1, w_2 are weighting factors with the condition $0 \leq w_1 \leq 1$ and $w_1 + w_2 = 1$. The travelling time T_{tra} can be formulated as

$$T_{tra} = \sum_{k=1}^{n} t_k \tag{3}$$

with $t_k = ts + 2\sqrt{D_k/a_{AV}}$, in which, t_k is the travelling time for kth segment, D_k is the displacement of motor for kth segment, a_{AV} is the average acceleration for motor speeding up.

Figure 1(b) illustrates a dynamic model for the LARM clutched arm, which consists of link2, link2′, link3, link3′ and a payload. Mass centres of links can be identified as shown in Fig. 1(b). τ_2, τ_3 are actuating torques for rotations about θ_2, θ_3 respectively, which can be computed from

$$\begin{bmatrix} \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \tag{4}$$

with

$$\begin{gathered} M_{11} = (m_2/2 + m_3 + m_p)L_2^2, \quad M_{12} = J\cos(\theta_3 - \theta_2)/4, \\ M_{21} = M_{12}, \quad M_{22} = m_2L_4^2 + m_3(L_3 + L_4)^2 + m_4L_4^2/4 + m_pL_3^2, \\ C_1 = -J\dot{\theta}_3^2 \sin(\theta_3 - \theta_2)/4 + (m_2 + m_3 + m_p)gL_2\cos\theta_2, \\ C_2 = J\dot{\theta}_2^2 \sin(\theta_3 - \theta_2)/4 - [m_2L_4 - m_3(L_3 - L_4) + m_4L_4 - 2m_pL_3]g\cos\theta_3/2, \\ J = -2m_2L_2L_4 + m_3(L_3 - L_4)L_2 + 4m_pL_2L_3, \end{gathered} \tag{5}$$

where m_2 is the mass for link2 and link2′; m_3, m_4 and m_p are masses for link3, link3′ and payload, respectively. L_4 is the link length of link3′.

If energy losses in the joints and gear trains are assumed to be negligible, the mechanical energy of actuator can be computed as the output mechanical energy in the form

$$W_{act} = \sum_{k=1}^{n} \left[\int_0^{t_k} (|\tau_2\dot{\theta}_2| + |\tau_3\dot{\theta}_3|)dt \right] \tag{6}$$

The LARM clutched arm does not move smoothly like those robotic arms with multi-actuators, since it has limited operation modes. However, an optimum path planning can be achieved by using limited options in a search reduced space. A* search algorithm can be suitable to solve the path planning problem for LARM clutched arm, since it solves the optimization path by using a heuristic function in order to search among the path possibilities as few as possible [6].

A flowchart of the proposed path planning with A* algorithm is shown in Fig. 4, where a node is a defined data structure with attributes of the coordinate in Joint Space, the current operation mode, the current travelling time and the current energy consumption. A father node is the selected node which can generate son nodes with the proposed operation modes in Table 1.

4 Numerical Examples

Numerical simulations of the proposed path planning have been computed with different value of weight factors, as listed in Table 2. For these examples, the start point

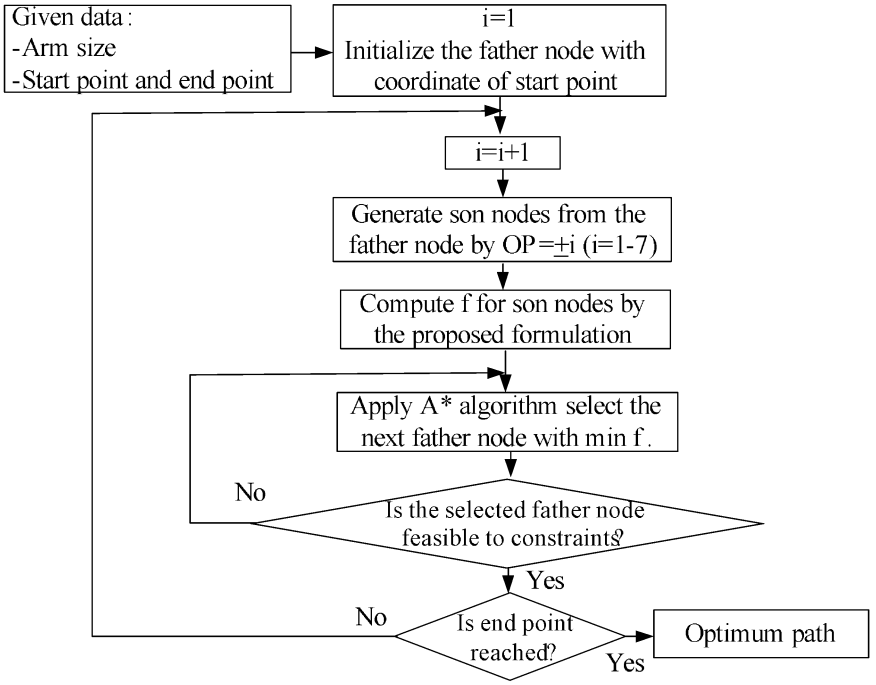


Fig. 4 A flowchart for the proposed path planning.

Table 2 Results of numerical simulations of the path planning in Fig. 4.

Objective Function	Travelling Time (s)	Work by Actuator (J)
$f = T_{tra}$	1.8130	27.4893
$f = W_{act}$	19.6518	1.6807
$f = 0.5T_{tra} + 0.5W_{act}$	13.1050	3.0094
$f = 0.9T_{tra} + 0.1W_{act}$	5.8556	3.0094

is ($\theta_2 = -90$ deg, $\theta_3 = 0$ deg) while the end point is ($\theta_2 = -50$ deg, $\theta_3 = 30$ deg). Link lengths for L_0 to L_4, are 1.0, 0.15, 0.3, 0.25 and 0.05 m, respectively. Masses for m_2 to m_4 and m_p are 0.3, 0.3, 0.05 and 1.0 kg. a_{AV} is 8,000 deg/s^2.

The path with minimum travelling time and maximum work by actuator is obtained when the objective function only considers travelling time. On the contrary, the path with minimum work by actuator and minimum travelling time is obtained when the objective function only takes into account the energy. It means that the criteria of T_{tra} and W_{act} are competitive to each other. Thus, with the objective function considering simultaneously both the criteria, the path can be computed as a compromising result. Moreover, the solution for the multi-objective function can converge to a result with a minimum for W_{act} only. Therefore, in order to have a good performance both in travelling time and energy consumption, the weight-

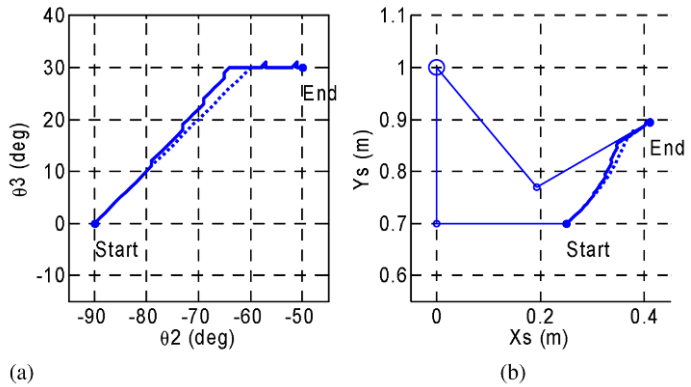


Fig. 5 Planned path with $f = T_{tra}$ (dotted line) and $f = 0.9T_{tra} + 0.1W_{act}$ (gross line): (a) in Joint-Space; (b) in Cartesian Space.

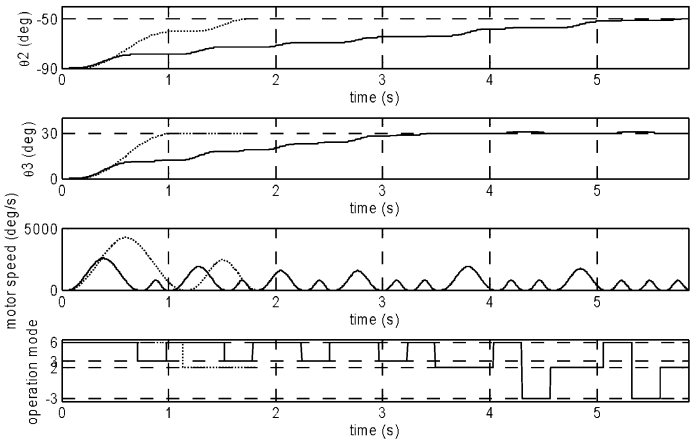


Fig. 6 Time sequence in the path planning with $f = T$ (dotted line) and $f = 0.9T_{tra} + 0.1W_{act}$ (gross line).

ing factor for T_{tra} should be bigger than that for W_{act}. As shown in Table 2, the travelling time is optimized properly with $f = 0.9T_{tra} + 0.1W_{act}$ more than with $f = 0.5T_{tra} + 0.5W_{act}$. Nevertheless, the two cases show the same work for the actuator.

Figure 5 illustrates the computed optimum path both in Joint Space and in Cartesian Space. The dotted line indicates the path with $f = T_{tra}$, while the gross line indicates the path with $f = 0.9T_{tra} + 0.1W_{act}$. For a short travelling time, the LARM clutched arm moves along the shortest path with large segments by using few operation modes. Nevertheless, when the energy criterium is considered, the LARM clutched arm moves with short segments by changing operation modes frequently, as shown in the path with $f = 0.9T_{tra} + 0.1W_{act}$.

Figure 6 illustrates the solved path planning, in which the time evolution of 2, 3, motor speed and operation modes have been reported. Similarly, the dotted lines are for the procedure with $f = T_{tra}$, while the gross lines are for the procedure with $f = 0.9T_{tra} + 0.1W_{act}$. The discrete motion feature of the LARM clutched arm can be recognised from the time evolution of operation modes, where the integer number i indicates the implemented operation mode OP. Because of the smooth motor speed with the cubic function, the evolution of 2 and 3 are still smooth enough for anthropomorphic applications.

5 Conclusions

A formulation of point-to-point optimum path planning for LARM clutched arm has been proposed as a multi-objective optimization problem with considerations of both mechanical work of actuator and travelling time, by taking into account the peculiar design and operation of the clutched system. Numerical results have shown the engineering feasibility of the proposed formulation for practical applications.

References

1. Nakamura, Y., Chung, W., Sordalen, O.J.: Design and control of the nonholonomic manipulator, *IEEE Transactions on Robotics and Automation*, **17**(1), 48–59 (2001).
2. Karbasi, H., Khajepour, A., Huissoon, J.P.: Unidrive modular robot: Dynamics, control, and experiments. *Journal of Dynamic Systems, Measurement, and Control*, **128**, 969–975 (2006).
3. Gu, H., Ceccarelli, M., Carbone, G.: Design and simulation of a 1-DOF anthropomorphic clutched arm for humanoid robots. *International Journal of Humanoid Robots*, **7**(1), 1–26 (2009).
4. Gu, H., Ceccarelli, M.: Simulation of combined motions for a 1-DOF clutched robotic arm. In: *Proceeding of ICMA 2009, IEEE International Conference of Mechatronics and Automation*, Paper 928381, Changchun (2009).
5. Carbone, G., Ceccarelli, M., Oliveira, P.J., Saramago, S.F.P., Carvalho, J.C.M.: An optimum path planning for Cassino parallel manipulator by using inverse dynamics. *Robotica*, **26**, 229–239 (2008).
6. Russell, S.J., Norvig, P.: *Artificial Intelligence: A Modern Approach*. Prentice Hall, Upper Saddle River (2002).

A Simple Kinematic Model of a Human Body for Virtual Environments

T. Koritnik, T. Bajd and M. Munih

Faculty of Electrical Engineering, University of Ljubljana, Slovenia;
e-mail: {tomazk, bajd, marko}@robo.fe.uni-lj.si

Abstract. We developed a simple kinematic model of a human body for real-time visualization applications in graphical virtual environments. For practical reasons a reduced number of active markers in optical measurements were employed to assess the values of joint variables, which caused computational issues in the configurations of the model that involved kinematic singularities. A method of handling the singularities by using simple algorithms is presented, enabling smooth and natural-appearing movements of the virtual figure without significantly affecting the natural ranges of human-like motion. The applicability of the model is demonstrated by a virtual mirror – a virtual reality application for real-time visualization of body movements enabling a visual feedback – which is useful in medical and performance studies.

Key words: Human modelling, virtual environment, kinematic singularity, motion visualization.

1 Introduction

Assessment, parameterization, and visualization of the human body movements are of utmost importance in modern analysis and diagnosis of various structural and functional pathologies, in rehabilitation, as well as in human movement science of sporting activities, ergonomics, bionics, and humanoid robotics [1]. Computer-aided modelling and simulation provide possibilities of novel approaches in the field of human motion analysis and synthesis. Accurate computer models and simulations enable the studies of motion patterns and parameters analytically as well as numerically, that were not feasible with classic techniques (photometry, accelerometry). Results are made readily accessible in cases where in vivo measurements and investigations are not possible.

Kinematic model can be used for visualization of the human body in the third-person virtual environments where a virtual figure representing the whole body or parts of it is observed on the computer screen from an external point of view. The figure can be placed into a three-dimensional virtual environment which is a model of the real physical environment with various objects of varying complexity. The concept of model visualization in virtual reality has recently entered the field of

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_43, © Springer Science+Business Media B.V. 2010

rehabilitation with notable success where it enables safer therapy, increased motivation of the patients, and enhanced engagement and interaction with the therapists [2]. Real-time functionality of the virtual environment is a critical feature since it engages the patient's biofeedback and provides augmented information that may be included in the visual feedback loop, which can lead to improvement of the overall rehabilitation outcome.

Computer model for visualizing the kinematical structure is based on a geometrical model. Real-time visualization of the human body involves measurements whose difficulty depends on the complexity of the model and accuracy requirements. It is not uncommon in human modelling that acquiring the adequate measurements and proper assessment turn out to be among the most challenging aspects of the process. In biomechanics, the use of special cameras in conjunction with passive or active markers is a well-established and commonly used technique. A large number of markers are required for a mathematically complete description of the configuration of kinematic structure as complex as human body; however, this number can be significantly reduced by presuming a set of geometric constraints and introducing the corresponding simplifications while retaining the kinematic versatility of the model. As a consequence, a complete description in certain configurations of the kinematic structure is not computable – the problem occurs in singular poses of the consecutive segments of the body.

In the paper we will present a kinematic model of the human body, a procedure to assess the complete configuration of the structure from optical measurements using a reduced set of markers, and a method to minimize the effects of indefiniteness in the singular poses of consecutive segments by adaptively suppressing the redundant degrees of freedom (DOF). The applicability of the model and the method will be demonstrated in a real-time virtual environment.

2 Human Body Kinematics

The presented kinematic model is designed primarily for visualization of the human figure in a virtual environment on a whole-body scale. The complexity is governed by the requirements of real-time operation on one side, whereas preserving the necessary number of DOF for convincing representation of the natural human-like motion on the other side. Proximal joints (hips and shoulders) are represented as close approximations of the real joints, while distal joints (fingers, wrists, ankles) are simplified accordingly. The model features hinge and spherical joints whose configuration is given by one and three parameters, respectively.

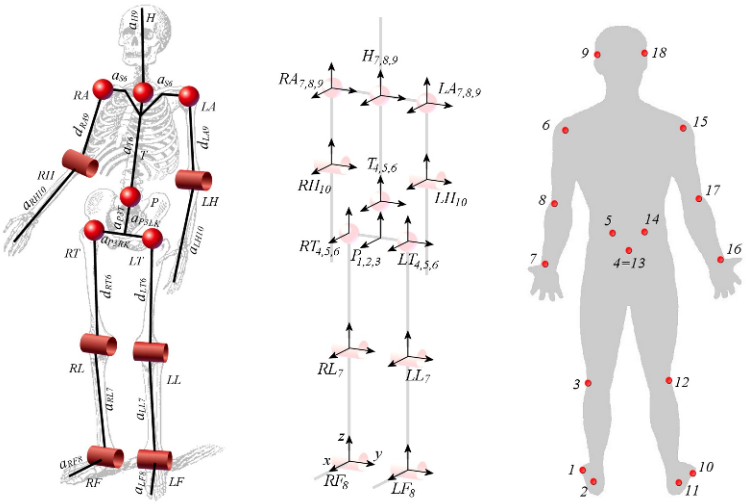


Fig. 1 Kinematic model (left), zero-configuration (middle), and marker placement (right).

2.1 Kinematic Model

Human body is represented as a mechanism with a tree-structured kinematic chain shown in Fig. 1 (left), comprised of 13 rigid segments representing the feet, shanks, thighs, pelvic segment, torso, upper arms, forearms, and head. The ankles, knees, and elbows are represented by 1-DOF hinge joints. The hips, shoulders, and joints connecting the pelvic segment with torso and torso with the head, respectively, are represented by 3-DOF spherical joints. The base of the mechanism is represented by the pelvic segment which can move in space freely and therefore has 6 DOF. The kinematic model has 24 articulated rotational DOF in total, with addition of the 3 rotational DOF and 3 translational DOF of the pelvic segment which define the position and orientation of the body with regard to the reference coordinate system. We used a method based on vector parameters to compute direct kinematics of the presented mechanism [3]. The placement of the coordinate systems and zero-configuration of the mechanism are shown in Fig. 1 (middle).

2.2 Motion Assessment

The assessment of angles of the adjoining segments must be provided in order to transfer the motion of the body to the virtual environment. In theory, at least 3 fixed points should be defined on each rigid segment determining its exact position and orientation in space. By presuming the underlying kinematic structure, the number of fixed points can be reduced substantially if chosen in accordance with geometric

constraints of the model. 18 points were defined representing the anatomic landmarks on the human body as shown in Fig. 1 (right), enabling the computation of joint variables for all DOF of the model.

The positions of these points were determined by measuring the positions of active markers placed accordingly, using the OPTOTRAK system (Northern Digital Inc.) with a 70-Hz sample rate. The position and orientation of the pelvic segment was determined from the three markers placed over the posterior superior iliac spines (PSIS) and lower edge of the sacrum. In addition, the positions of the PSIS and sacral markers were used to calculate the centers of the hip joints [4]. The midpoint between the hip joints was considered as the base point. One marker was placed on the skin at the approximate center of rotation of each metatarsophalangeal (MTP) joint, knee, ankle, wrist, elbow, and shoulder in order to determine the positions of the underlying joints, respectively. Two markers were placed symmetrically above the ears, determining the pose of the head.

The position of the body was represented by the position coordinates of the pelvic base point. Vector cross-product operations were applied to the consecutive body-segment vectors to obtain the segment coordinate systems, and the corresponding axes and angles of joint rotations.

2.3 Singularity Handling

Using the proposed angle-determination method is applicable in most configurations of the normal ranges of motion of human joints except in full extension of the elbow and knee. Full extension of the joint corresponds to the singular pose of the model which causes computational issues; the model behaves as having one articulated DOF less. The direction of the vector product of two collinear vectors representing consecutive segments in the singular pose is not defined unambiguously, making it impossible to calculate the direction of the resulting rotation axis. Full extension of the elbows and knees is common in the activities of daily living; hence, the issue with singular poses should not be neglected. We will first demonstrate the procedure for handling the singularity in the elbow joint which is simpler and then expand the solution to knee and ankle joints. The calculations are presented for the left arm and leg.

The elbow angle is defined by the coordinate vectors representing the upper arm *LA* and forearm *LH*, determining the axis of rotation $\boldsymbol{y}_{LH}$ via their vector product as shown in Fig. 2 (left, top). The magnitude of this vector product is used as a measure of proximity of the singular pose:

$$|\boldsymbol{z}_{LH} \times \boldsymbol{z}_{LA}| = |\boldsymbol{z}_{LH}||\boldsymbol{z}_{LA}| \sin\theta. \tag{1}$$

Since both coordinate vectors are of magnitude 1 by definition, the magnitude of their vector product amounts to $\sin\theta$. As the angle approaches the value $k\pi$ (singularity) the magnitude of vector product approaches 0, whereas the maximal value of

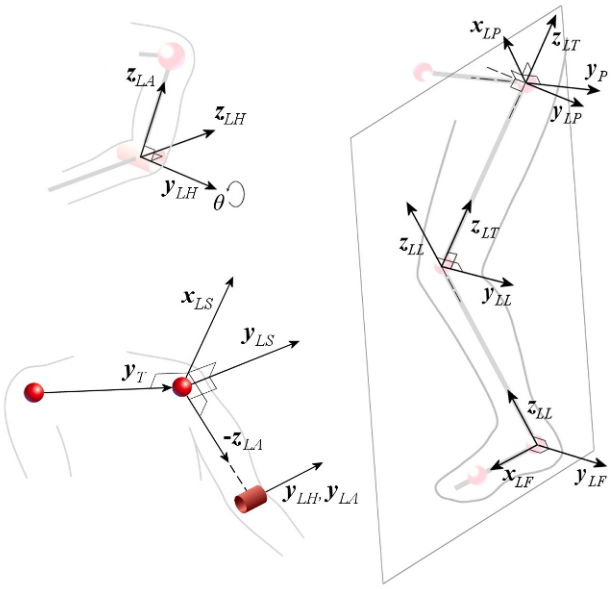


Fig. 2 Directions of rotation axes: arm (left) and leg (right).

1 is reached at angle values of $\theta = k\pi + \frac{\pi}{2}$. The direction of the $\boldsymbol{y}$ axis is therefore indefinite in the singular poses. We define a new supplementary axis $\boldsymbol{y}_{LS}$:

$$\boldsymbol{x}_{LS} = \boldsymbol{y}_T \times \boldsymbol{z}_{LA}, \tag{2}$$

$$\boldsymbol{y}_{LS} = \boldsymbol{z}_{LA} \times \boldsymbol{x}_{LS}. \tag{3}$$

The $\boldsymbol{y}_{LS}$ axis points in the same direction as the $\boldsymbol{y}$ axis of the shoulder joint would have, if the rotation about the $\boldsymbol{z}_{LA}$ axis of the upper arm was not possible, i.e. in case of only 2 DOF shoulder joint, as defined by the position of the elbow marker and the orientation of the torso ($\boldsymbol{y}_T$), shown in Fig. 2 (left, bottom). Next, two coefficients are defined:

$$\boldsymbol{f}_{LA1} = |\boldsymbol{y}_{LA}|, \tag{4}$$

$$\boldsymbol{f}_{LA2} = 1 - \boldsymbol{f}_{LA1}. \tag{5}$$

The coefficient f_{LA1} equals the magnitude of the vector product of the $\boldsymbol{z}$ coordinate vectors of the upper arm and forearm, amounting to a value between 0 and 1. A value of 1 indicates the best case where the angle is $\pi/2$, whereas value 0 indicates the singularity. The two defined coefficients are then used as the weights to the vectors $\boldsymbol{y}_{LA}$ and $\boldsymbol{y}_{LS}$ of which the first indicates the direction of rotation axis defined by the upper arm and forearm, whereas the second indicates the direction axis defined by the upper arm and torso. The new alternate $\boldsymbol{y}'_{LA}$ axis is obtained as a weighed sum:

$$\boldsymbol{y}'_{LA} = f_{LA1}\hat{\boldsymbol{y}}_{LA} + f_{LA2}\hat{\boldsymbol{y}}_{LS}. \tag{6}$$

This new vector is then used both as the axis of $\boldsymbol{y}$ rotation of the shoulder and the $\boldsymbol{y}$ rotation of the elbow since the arm itself, according to the model, is a planar mechanism ($\boldsymbol{y}_{LH} = \boldsymbol{y}_{LA}$). In general it is not critical to use unit vectors of magnitude 1 when calculating directions of the resulting vector products; however it is essential to use unit vectors $\hat{\boldsymbol{y}}_{LA}$ and $\hat{\boldsymbol{y}}_{LS}$ in Eq. (6) in order to ensure that the weights bear proper significance. The remaining directional vectors must only be normalized prior to composing the rotational matrices.

The described procedure yields an alternate direction of rotation axis as a weighed sum of two coordinate vectors; the first coordinate vector is weighed by a factor inversely proportional with the proximity of the singularity, whereas the second vector is weighed proportionately with it. In this way, the shoulder joint features 3 DOF in regions of good definiteness while the significance of the third DOF (rotation about the $\boldsymbol{z}_{LH}$) decreases continuously nearing the singularity and disappears completely in full extension of the elbow. The same approach is used for the lower extremities, except that three joints instead of two can be used to calculate the alternate direction of the rotation axes. The direction of the knee rotation axis ($\boldsymbol{y}_{LL}$) is obtained by performing cross-product operations upon the thigh and shank unit vectors as shown in Fig. 2 (right):

$$\boldsymbol{y}_{LL} = \boldsymbol{z}_{LT} \times \boldsymbol{z}_{LL}. \tag{7}$$

According to our simplified model, the direction of the ankle rotation axis is the same as the knee axis, and is obtained from the shank and foot unit vectors:

$$\boldsymbol{y}_{LF} = \boldsymbol{z}_{LL} \times \boldsymbol{x}_{LF}. \tag{8}$$

The third knee-axis calculation assumes that the hip joint has only two DOF, with the longitudinal rotation about the thigh $\boldsymbol{z}$ axis omitted ($\boldsymbol{y}_P$ is a unit vector connecting both hip joints):

$$\boldsymbol{x}_{LP} = \boldsymbol{y}_P \times \boldsymbol{z}_{LT}, \tag{9}$$

$$\boldsymbol{y}_{LP} = \boldsymbol{z}_{LT} \times \boldsymbol{x}_{LP}. \tag{10}$$

For each of the three $\boldsymbol{y}$ axis vectors, a weighting factor is considered:

$$f_{LL} = |\boldsymbol{y}_{LL}|, \tag{11}$$

$$f_{LF} = \begin{cases} (1 - |\boldsymbol{y}_{LL}|)|\boldsymbol{y}_{LF}| \; ; \; (1 - |\boldsymbol{y}_{LL}|) > 0, \\ 0 \qquad\qquad ; \; (1 - |\boldsymbol{y}_{LL}|) \leq 0, \end{cases} \tag{12}$$

$$f_{LP} = \begin{cases} 1 - (f_{LL} + f_{LF}) \; ; \; (1 - (f_{LL} + f_{LF})) > 0, \\ 0 \qquad\qquad ; \; (1 - (f_{LL} + f_{LF})) \leq 0. \end{cases} \tag{13}$$

Finally, the new axis of knee and ankle rotation is obtained as a weighted sum of unit vectors:

$$\boldsymbol{y}'_{LL} = f_{LL}\hat{\boldsymbol{y}}_{LL} + f_{LF}\hat{\boldsymbol{y}}_{LF} + f_{LP}\hat{\boldsymbol{y}}_{LP}. \tag{14}$$

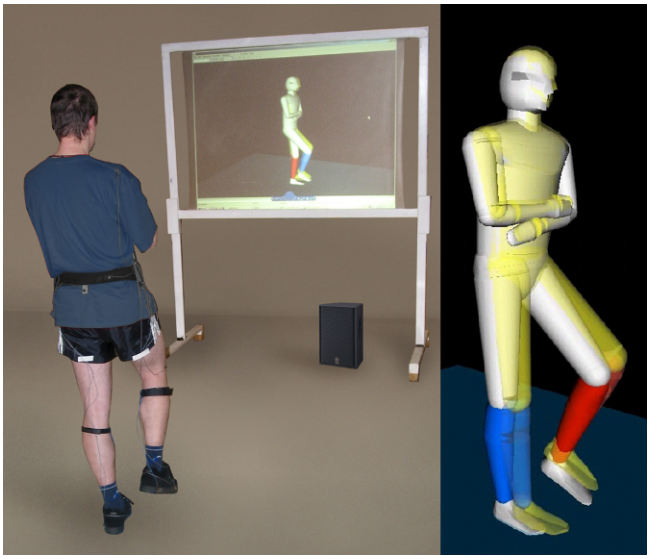


Fig. 3 Virtual mirror (left) and virtual figure enlarged (right).

In poses where the knee rotation axis is well pronounced, the significance of the first weighting factor is prevalent in Eq. (8). Near the singularity of the knee joint, the first weighting factor moves towards zero whereas the second factor increases if the ankle axis is well pronounced. In the worst case, when the ankle is also approaching singularity (in the case of strong plantar flexion atop of the fully extended knee), the third weighting factor gains significance. In this case, the hip joint is considered as having only two DOF, thus preventing any longitudinal rotation about the thigh axis z_{LT}.

2.4 Virtual Mirror

Kinematic data, which were calculated from the OPTOTRAK measurements, were used to animate the motion of the human model in VR shown in Fig. 3 (right). The virtual environment was presented on a large screen in front of which the subject performed the desired movements and observed the virtual figure executing the same movements in real-time, thereby creating the impression of a virtual mirror, shown in Fig. 3 (left). The human figure matched the kinematic model, and imitated the shape of the human body [5]. The ratios between the segment lengths were based on statistical anthropometry [6]. The movements of the figure corresponded to the movements of the subject in real time at a 35 Hz refresh rate without detectable lag. We used VRML 2.0 (Virtual Reality Modeling Language) to visualize the movements of the figure.

3 Conclusions

The presented kinematic model for use in VR visualization had 30 DOF. Simplifications were made by omitting DOF of shorter distal body segments and representing the ankles, knees, and elbows as one DOF joints which is not uncommon in human modeling [7]. As such, the motion of the virtual figure proved to be a convincing representation of the actual movements on the whole-body scale. The computational issues caused by the singular poses in the knee and elbow joints were solved effectively by the singularity handling procedure, resulting in smooth and natural-appearing motion. Without exploiting this simple procedure, the model exhibited sudden knee-axis and elbow-axis shifts and inadequate poses in an unpredictable manner when approaching singularities. On the down side, considering hips and shoulders as 2-DOF joints generally results in unsatisfactory model behavior in terms of the ability to realize the actual poses of the subjects in normal range of motion. In our case, this limitation was reflected as inability to visualize the longitudinal rotation of arms or legs when fully stretched although this is anatomically possible. Furthermore, the method does not resolve the case when all four (arm) or five (leg) markers concerned are collinear. The complexity level of the presented kinematic model incorporates compromise of the substantially reduced number of markers, the desired fidelity of the motion and smooth real-time performance of the VR model. As such it can be used in conjunction with the virtual mirror as a basic platform in medical studies requiring visual feedback of body motion [8].

Acknowledgements The authors acknowledge the financial support from the Slovenian Research Agency.

References

1. Medved, V.: *Measurement of Human Locomotion*. CRC press, Boca Raton (2001).
2. Holden, M.K.: Virtual environments for motor rehabilitation: review. *Cyberpsychol. Behav.* **8**(3), 187–211 (2005).
3. Lenarčič, J.: Kinematics. In: Dorf R.C., Nof S.Y. (Eds.), *International Encyclopedia of Robotics*. John Wiley, New York (1988).
4. Frigo, C. and Rabuffetti, M.: Multifactorial estimation of hip and knee joint centres for clinical application and gait analysis. *Gait Posture* **8**(2), 91–102 (1998).
5. Zatsiorsky, V.M.: *Kinematics of Human Motion*. Human Kinetics Publishers, Champaign (1998).
6. De Leva, P.: Adjustments to Zatsiorsky–Seluyanov's segment inertia parameters. *J. Biomech.* **29**, 1223–1230 (1995).
7. Zhao, W., Wu, G., and Sommer, H.J.: Closed form kinematics for a spatial closed-chain mechanism modeling biped stance. *Mech. Mach. Theory* **33**(4), 379–387 (1998).
8. Koritnik T., Bajd T., and Munih, M.: Virtual environment for lower-extremities training. *Gait Posture* **27**(2), 323–330 (2008).

PART 7

On the Development of Low Mass Shaking Force Balanced Manipulators

V. van der Wijk[1,2] and J.L. Herder[1,2]

[1] *University of Twente, Enschede, The Netherlands*
[2] *Delft University of Technology, Delft, The Netherlands;*
e-mail: v.vanderwijk@kineticart.nl, j.l.herder@tudelft.nl

Abstract. Whenever mechanisms have to run at high speeds, dynamic balancing (shaking force and shaking moment balancing) becomes an important issue to reduce machine vibrations. This article considers Fischer's method of principal vectors for shaking force balancing with low mass addition. A method based on linear momentum equations and Equivalent Linear Momentum Systems is proposed to calculate the principal vectors for all mechanism elements having an arbitrary mass distribution. A planar serial chain of three bodies, and a planar serial chain of four bodies are studied. The theory is verified experimentally for a planar serial chain of five bodies.

Key words: Shaking force balancing, linear momentum, center of mass, machine dynamics.

1 Introduction

Whenever mechanisms (i.e. robots, manipulators) have to run at high speeds, dynamic balancing of the shaking forces and shaking moments becomes an important issue. Due to mass and inertia of the moving elements, shaking forces and shaking moments are exerted on the base. This results in vibrations which have a negative effect on the level of noise, wear, and fatigue [1]. Dynamically balanced mechanisms do not exert vibrations on the base and environment, since both the linear momentum and angular momentum of the moving components are constant for any motion of the mechanism. Therefore dynamically balanced mechanisms can have an increased accuracy and shorter cycle times since no waiting times are needed to have vibrations die out [2]. For hand tools, dynamic balancing improves the ergonomics and reduces the risks of injuries [3] while for moving objects (at the ground or in space), dynamic balance is important for maintaining the orientation [4].

A general and serious problem with dynamic balancing is that often a considerable amount of mass and inertia is added [5, 6]. The addition of mass is mainly due to the application of counter-masses, which are used to make the linear momentum of the mechanism constant by having the center of mass (CoM) of the mechanism be stationary [7].

This article studies Fischer's method of principal vectors [1, 8] for physically tracing the CoM of a collection of rigid bodies for the purpose of (shaking) force

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_44,

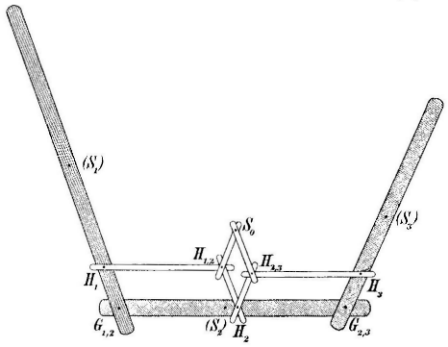


Fig. 1 Fischer's mechanism to trace the CoM of three bodies S_0 by additional links, source [8].

balancing. It will be shown that by using Fischer's method, force balance is achieved by the kinematic design of the mechanism instead of by adding counter-masses. This reduces the addition of mass considerably.

Figure 1 shows Fischer's method for a planar mechanism of three rigid bodies. By additional parallelogram links with joints at H_1, H_2, and H_3 (the principal points), the CoM of bodies 1, 2, and 3 is traced by point S_0. When point S_0 is considered to be a joint at the base, then the CoM is stationary with respect to the base and the mechanism is force balanced. If this joint is the only connection with the base, then the mechanisms is a two degrees of freedom (DoF) force balanced manipulator.

The idea of having S_0 be a stationary point for the purpose of force balancing was proposed by Fischer in [8]. To have S_0 be a joint at the base and including also the mass of the parallelogram links was first considered by Agrawal et al. in [9]. Although they demonstrated experimentally that including these masses is possible, they did not solve it theoretically.

The purpose of this article is to provide a method, based on linear momentum equations, to calculate the dimensions of the parallelogram links such that the CoM of all moving bodies, including the added parallelogram links, can be made stationary with respect to the base.

It is assumed that every element has an arbitrary mass-distribution. The article is limited to planar mechanisms. First the mechanism of Fig. 1 is studied, which is referred to as a serial chain of three bodies. Consequently an extended serial chain of four bodies is considered, followed by validation of the theory by a mechanism model of a five-body serial chain.

2 Force Balanced Manipulator with Three Degrees of Freedom

Figure 2 shows Fischer's mechanism of Fig. 1 in which each element has an arbitrary mass distribution. The masses of the three bodies are m_1, m_2, and m_3, and are located at positions M_1, M_2, and M_3, respectively. The masses m_{ij} of the parallelogram links

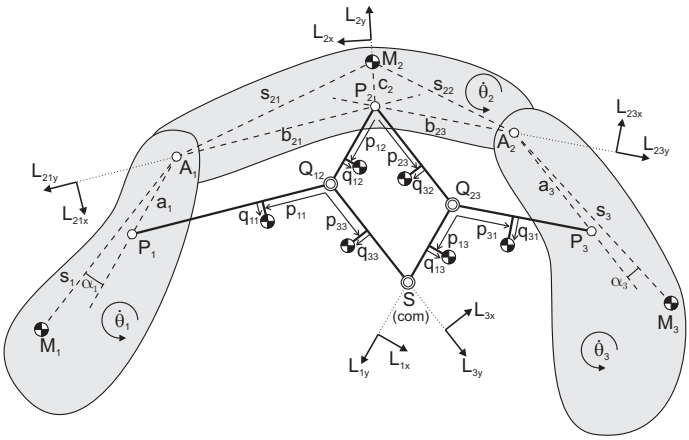


Fig. 2 S is the CoM of three bodies in series and additional links with arbitrary mass distribution.

are positioned at distance p_{ij} from the indicated joint along the link, and at distance q_{ij} normal to the respective link. The angular velocities of the three bodies are $\dot{\theta}_1$, $\dot{\theta}_2$, and $\dot{\theta}_3$, respectively.

The purpose is to calculate the lengths of the parallelogram links which depend on the positions of the principle points P_1, P_2, and P_3, i.e. the points where the parallelogram links are jointed to the mechanism. These points are determined by distances a_1, a_3, b_{21}, and b_{23}, and angles α_1 and α_3.

For the calculation of these parameters each DoF is considered independently. This means that each of the parallelograms is considered immovable once at a time. First consider parallelogram $A_2P_2Q_{23}P_3$ to be immovable, which means that $\dot{\theta}_2$ and $\dot{\theta}_3$ are zero and the other two parallelograms are movable with one DoF. To have the CoM of the mechanism be at S at any time, the linear momentum of the moving masses must be equal to the linear momentum of the total mass of the mechanism m_{tot} at S. For an instantaneous position of the mechanism, the directions of the linear momentum L_{1x} and L_{1y} can be chosen to be perpendicular and parallel to the line a_1, respectively, as indicated in Fig. 2. Then the following two linear momentum equations are obtained.

$$L_{1x} = (m_1 s_1 \cos\alpha_1 + (m_{11} + m_{33})a_1 + m_{12}p_{12} + m_{13}p_{13})\dot{\theta}_1 = m_{tot}a_1\dot{\theta}_1 \quad (1)$$

$$L_{1y} = (m_1 s_1 \sin\alpha_1 - m_{12}q_{12} - m_{13}q_{13})\dot{\theta}_1 = 0 \quad (2)$$

with

$$m_{tot} = m_1 + m_2 + m_3 + m_{11} + m_{12} + m_{13} + m_{31} + m_{32} + m_{33} \quad (3)$$

When the masses and their positions are known, from Eqs. (2) and (1), the equations for α_1 and a_1 become, respectively

$$\alpha_1 = \sin^{-1}\left(\frac{m_{12}q_{12} + m_{13}q_{13}}{m_1 s_1}\right) \tag{4}$$

$$a_1 = \frac{m_1 s_1 \cos\alpha_1 + m_{12}p_{12} + m_{13}p_{13}}{m_{tot} - m_{11} - m_{33}} \tag{5}$$

Similarly α_3 and a_3 are calculated. Therefore parallelogram $A_1P_1Q_{12}P_2$ is considered to be immovable, by which the other two parallelograms are movable with one DoF ($\dot{\theta}_1$ and $\dot{\theta}_2$ are zero). When for an instantaneous position of the mechanism the directions of the linear momentum L_{3x} and L_{3y} are chosen to be perpendicular and parallel to line a_3, respectively, as indicated in Fig. 2, the following equations for the linear momentum are obtained.

$$L_{3x} = (m_3 s_3 \cos\alpha_3 + (m_{31} + m_{13})a_1 + m_{32}p_{32} + m_{33}p_{33})\dot{\theta}_3 = m_{tot}a_3\dot{\theta}_3 \tag{6}$$

$$L_{3y} = (-m_3 s_3 \sin\alpha_3 + m_{32}q_{32} + m_{33}q_{33})\dot{\theta}_3 = 0 \tag{7}$$

which result for α_3 and a_3 in

$$\alpha_3 = \sin^{-1}\left(\frac{m_{32}q_{32} + m_{33}q_{33}}{m_3 s_3}\right) \tag{8}$$

$$a_3 = \frac{m_3 s_3 \cos\alpha_3 + m_{32}p_{32} + m_{33}p_{33}}{m_{tot} - m_{31} - m_{13}} \tag{9}$$

For the calculation of lengths b_{21} and b_{23}, the parallelogram $P_2Q_{12}SQ_{23}$ is considered immovable, while the other parallelograms are movable with one DoF. Bodies 1 and 3 then have solely translational motion ($\dot{\theta}_1$ and $\dot{\theta}_3$ are zero) while body 2 solely rotates about P_2. Since the CoM at S now is immovable, the linear momentum of the mechanism with respect to P_2 must be zero.

The linear momentum of the moving elements is calculated with respect to various reference frames and summed. Three reference frames are defined. The linear momentum of m_1 and m_{11} is calculated with respect to a reference frame that is perpendicular and parallel to line b_{21}, which are L_{21x} and L_{21y} as shown in Fig. 2, respectively. The linear momentum of m_3 and m_{31} is calculated with respect to a reference frame that is perpendicular and parallel to line b_{23}, which are L_{23x} and L_{23y}, respectively. The linear momentum of m_2 is calculated with respect to a reference frame that is perpendicular and parallel to line c_2, which are L_{2x} and L_{2y}, respectively. The linear momentum equations in each reference frame result in

$$\begin{aligned} L_{21x} &= (m_1 b_{21} + m_{11}p_{11})\dot{\theta}_2 & L_{21y} &= (-m_{11}q_{11})\dot{\theta}_2 \\ L_{23x} &= (m_3 b_{23} + m_{31}p_{31})\dot{\theta}_2 & L_{23y} &= (m_{31}q_{31})\dot{\theta}_2 \\ L_{2x} &= (m_2 c_2)\dot{\theta}_2 & L_{2y} &= 0 \end{aligned} \tag{10}$$

To facilitate the summing of the linear momentum equations, it is proposed to project the moving masses on body 2 such that the linear momentum equations of this projection are equal to the linear momentum equations of the mechanism. This projection is referred to as the *Equivalent Linear Momentum System* (ELMS), and

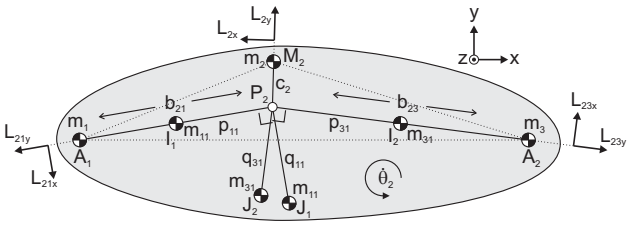


Fig. 3 Equivalent Linear Momentum System w.r.t. body 2 to find P_2; This is the projection of the moving masses on body 2 to result in equivalent linear momentum equations.

has the advantage that the relations among the reference frames do not need to be formulated. The ELMS in this case is shown in Fig. 3 and has masses m_1, m_2, and m_3 positioned at A_1, M_2, and A_2, respectively, a mass m_{11} positioned at both I_1 and J_1, and a mass m_{31} positioned at both I_2 and J_2.

An useful feature of the ELMS is that, in order to have a zero linear momentum, P_2 must be the CoM of all the projected masses. When m_{11} and m_{31} are not zero, it is difficult to find an algebraic equation for P_2 because of the dependency of positions I_1, I_2, J_1, and J_2 and lengths b_{21}, b_{23}, and c_2 on P_2. Numerically, P_2 can be found by solving the equations of the summed linear momentum $\overrightarrow{L}$, which can be written from the ELMS of Fig. 3 in $[x\ y\ z]^T$ notation as

$$\begin{aligned}\overrightarrow{L} = (&u_1 \cdot \overrightarrow{A_1P_2} \times \hat{z} + v_1 \cdot \overrightarrow{A_1P_2} + u_2 \cdot \overrightarrow{A_2P_2} \times \hat{z} \\ &- v_2 \cdot \overrightarrow{A_2P_2} + u_3 \cdot \overrightarrow{M_2P_2} \times \hat{z})\dot{\theta}_2 = \overrightarrow{0}\end{aligned} \tag{11}$$

with unit vector $\hat{z} = [0\ 0\ 1]^T$ and

$$\begin{aligned} u_1 &= m_1 + \frac{m_{11}p_{11}}{b_{21}} \qquad u_2 = m_3 + \frac{m_{31}p_{31}}{b_{23}} \qquad u_3 = m_2 \\ v_1 &= \frac{m_{11}q_{11}}{b_{21}} \qquad\qquad v_2 = \frac{m_{31}q_{31}}{b_{23}} \end{aligned} \tag{12}$$

in which $b_{21} = |\overrightarrow{A_1P_2}|$ and $b_{23} = |\overrightarrow{A_2P_2}|$. The only unknown in the equation is P_2. The cross product with unit vector $\hat{z}$ is used to calculate perpendicular directions within the xy-plane.

If distances q_{11} and q_{31} are zero and the ratios $p_{11}/b_{21} = \lambda_1$ and $p_{31}/b_{23} = \lambda_2$ are known, for instance if m_{11} is halfway length b_{21} by which $\lambda_1 = 1/2$, an algebraic solution for P_2 can be found, which is

$$\overrightarrow{P_2} = \frac{u_1\overrightarrow{A_1} + u_2\overrightarrow{A_2} + u_3\overrightarrow{M_2}}{u_1 + u_2 + u_3} \tag{13}$$

with

$$u_1 = m_1 + m_{11}\lambda_1 \qquad u_2 = m_3 + m_{31}\lambda_2 \qquad u_3 = m_2 \tag{14}$$

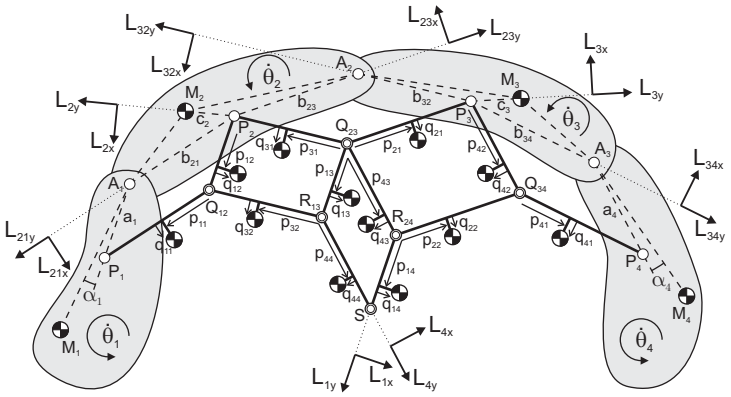


Fig. 4 S is the CoM of four bodies in series and additional links with arbitrary mass distribution.

In fact, this solution implies that P_2 is the CoM of masses with values u_1, u_2, and u_3 positioned at positions A_1, A_2, and M_2, respectively.

3 Force Balanced Manipulator with Four Degrees of Freedom

In this section, the principal points are calculated of a planar serial chain of four bodies and parallelogram links with arbitrary mass distribution. It will be shown that the calculation of the principal points of extended serial chains is similar to the case of three bodies in series and that each principle point can be obtained individually.

Figure 4 shows Fischer's mechanism for four bodies in series, extended with an arbitrary mass distribution for each element. The masses of the four bodies are m_1, m_2, m_3, and m_4, located at positions M_1, M_2, M_3, and M_4, respectively. The masses m_{ij} of the parallelogram links are positioned at distance p_{ij} from the indicated joint along the link and at distance q_{ij} normal to the respective link. The angular velocities of the four bodies are $\dot{\theta}_1$, $\dot{\theta}_2$, $\dot{\theta}_3$, and $\dot{\theta}_4$, respectively.

P_1 is found in an equivalent way as was done for three bodies in series. To find P_1, which is defined by a_1 and α_1, the parallelograms of which the motion does not depend on $\dot{\theta}_1$ ($A_2P_2Q_{23}P_3$, $P_3Q_{23}R_{24}Q_{34}$, and $A_3P_3Q_{34}P_4$) are considered to be immovable. The other parallelograms then are movable with one DoF. The reference frame for the linear momentum L_{1x} and L_{1y} is chosen to be perpendicular and parallel to line a_1, respectively. To have S be the CoM of the mechanism, the linear momentum of the moving masses must equal the linear momentum of the total mass considered at S. The linear momentum equations in each direction are written as

$$\begin{aligned} L_{1x} &= (m_1 s_1 \cos\alpha_1 + (m_{11} + m_{32} + m_{44})a_1 + m_{12}p_{12} \\ &\quad + m_{13}p_{13} + m_{14}p_{14})\dot{\theta}_1 = m_{tot}a_1\dot{\theta}_1 \qquad (15) \\ L_{1y} &= (m_1 s_1 \sin\alpha_1 - m_{12}q_{12} - m_{13}q_{13} - m_{14}q_{14})\dot{\theta}_1 = 0 \qquad (16) \end{aligned}$$

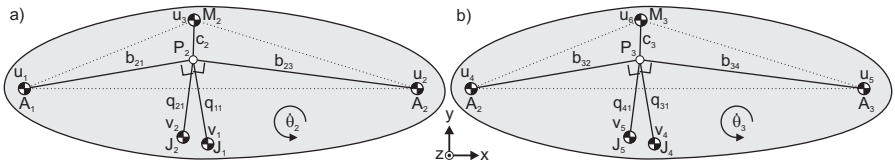


Fig. 5 Equivalent Linear Momentum Systems w.r.t. (a) body 2 to find P_2 (b) body 3 to find P_3.

with

$$\begin{aligned} m_{tot} = {} & m_1 + m_2 + m_3 + m_4 + m_{11} + m_{12} + m_{13} + m_{14} \\ & + m_{21} + m_{22} + m_{31} + m_{32} + m_{41} + m_{42} + m_{43} + m_{44} \end{aligned} \tag{17}$$

With the masses and their positions on the mechanism being known, the equations for α_1 and a_1 become

$$\alpha_1 = \sin^{-1}\left(\frac{m_{12}q_{12} + m_{13}q_{13} + m_{14}q_{14}}{m_1 s_1}\right) \tag{18}$$

$$a_1 = \frac{m_1 s_1 \cos\alpha_1 + m_{12}p_{12} + m_{13}p_{13} + m_{14}p_{14}}{m_{tot} - m_{11} - m_{32} - m_{44}} \tag{19}$$

Equivalently P_4 is found, which is determined by a_4 and α_4. Therefore the parallelograms of which the motion does not depend on $\dot{\theta}_4$ ($A_1P_1Q_{12}P_2$, $P_2Q_{12}R_{13}Q_{23}$, and $A_2P_2Q_{23}P_3$) are considered to be immovable. For the reference frame of the linear momentum L_{4x} and L_{4y} being perpendicular and parallel to line a_4, respectively, the linear momentum equations become

$$\begin{aligned} L_{4x} = {} & (m_4 s_4 \cos\alpha_4 + (m_{41} + m_{22} + m_{14})a_4 + m_{42}p_{42} \\ & + m_{43}p_{43} + m_{44}p_{44})\dot{\theta}_4 = m_{tot}a_4\dot{\theta}_4 \end{aligned} \tag{20}$$

$$L_{4y} = (-m_4 s_4 \sin\alpha_4 + m_{42}q_{42} + m_{43}q_{43} + m_{44}q_{44})\dot{\theta}_4 = 0 \tag{21}$$

The equations for α_4 and a_4 then become

$$\alpha_4 = \sin^{-1}\left(\frac{m_{42}q_{42} + m_{43}q_{43} + m_{44}q_{44}}{m_4 s_4}\right) \tag{22}$$

$$a_4 = \frac{m_4 s_4 \cos\alpha_1 + m_{42}p_{42} + m_{43}p_{43} + m_{44}p_{44}}{m_{tot} - m_{41} - m_{22} - m_{14}} \tag{23}$$

For the calculations of P_2 and P_3 the ELMS with respect to bodies 2 and 3 are used, which are shown in Fig. 5. To calculate P_2, parallelograms $P_2Q_{12}R_{13}Q_{23}$ and $Q_{23}R_{13}SR_{24}$ are considered to be immovable, for which the mechanism is movable with one DoF. Bodies 1, 3, and 4 then solely translate ($\dot{\theta}_1$, $\dot{\theta}_3$, and $\dot{\theta}_4$ are zero) and body 2 solely rotates. Figure 5a shows the ELMS with respect to body 2 in which masses with values u_1, u_2, u_3, v_1, and v_2 are placed at positions A_1, A_2, M_2, J_1, and

J_2 respectively. The values of these masses are

$$\begin{aligned} u_1 &= m_1 + \frac{m_{11}p_{11}}{b_{21}} & v_1 &= m_{11} \\ u_2 &= m_3 + m_4 + m_{41} + m_{42} + \frac{m_{21}p_{21} + m_{22}p_{22}}{b_{23}} & v_2 &= m_{21} + \frac{m_{22}q_{22}}{q_{21}} \\ u_3 &= m_2 \end{aligned} \tag{24}$$

and can be obtained, similarly as in Section 2, by calculating the linear momentum with respect to various reference frames. The linear momentum of m_1 and m_{11} is calculated in directions L_{21x} and L_{21y}, the linear momentum of m_2 is calculated in directions L_{2x} and L_{2y}, and the linear momentum of m_3, m_4, m_{41}, m_{42}, m_{21}, and m_{22}, is calculated in directions L_{23x} and L_{23y}. For the ease of drawing, all the masses that are along the same line are scaled and concentrated in a single point. For instance mass u_1 at A_1 is a combination of m_1 located at A_1 and m_{11} located along line A_1P_2 at a distance p_{11} from P_2. For an equivalent linear momentum of m_{11}, a mass $m_{11}p_{11}/b_{21}$ can be located at A_1. To find P_2, the equation of the summed linear momentum must be solved which is

$$\begin{aligned} \overrightarrow{L} = (&u_1 \cdot \overrightarrow{A_1P_2} \times \hat{z} + v_1 \cdot \overrightarrow{A_1P_2} + u_2 \cdot \overrightarrow{A_2P_2} \times \hat{z} \\ &- v_2 \cdot \overrightarrow{A_2P_2} + u_3 \cdot \overrightarrow{M_2P_2} \times \hat{z})\dot{\theta}_2 = \overrightarrow{0} \end{aligned} \tag{25}$$

in which P_2 is the only unknown.

For the calculation of P_3, the parallelograms $P_3Q_{23}R_{24}Q_{34}$ and $Q_{23}R_{13}SR_{24}$ are considered to be immovable, for which the mechanism is movable with one DoF. In this case bodies 1, 2, and 4 solely translate ($\dot{\theta}_1$, $\dot{\theta}_2$, and $\dot{\theta}_4$ are zero) and body 3 solely rotates. Figure 5b shows the ELMS with respect to body 3 in which masses with values u_4, u_5, u_6, v_4, and v_5 are placed at positions A_2, A_3, M_3, J_4, and J_5, respectively. The values of these masses are

$$\begin{aligned} u_4 &= m_1 + m_2 + m_{11} + m_{12} + \frac{m_{31}p_{31} + m_{32}p_{32}}{b_{32}} & v_4 &= m_{31} + \frac{m_{32}q_{32}}{q_{31}} \\ u_5 &= m_4 + \frac{m_{41}p_{41}}{b_{34}} & v_5 &= m_{41} \\ u_6 &= m_3 \end{aligned} \tag{26}$$

and can be obtained from the calculations of the linear momentum of m_1, m_2, m_{11}, m_{12}, m_{31}, and m_{32} in directions L_{32x} and L_{32y}, the linear momentum of m_3 in directions L_{3x} and L_{3y} and the linear momentum of m_4, and m_{41} in directions L_{34x} and L_{34y}. Hence to find P_3, the equation of the summed linear momentum must be solved which is written as

$$\begin{aligned} \overrightarrow{L} = (&u_4 \cdot \overrightarrow{A_2P_3} \times \hat{z} + v_4 \cdot \overrightarrow{A_2P_3} + u_5 \cdot \overrightarrow{A_3P_3} \times \hat{z} \\ &- v_5 \cdot \overrightarrow{A_3P_3} + u_6 \cdot \overrightarrow{M_3P_3} \times \hat{z})\dot{\theta}_3 = \overrightarrow{0} \end{aligned} \tag{27}$$

in which P_3 is the only unknown.

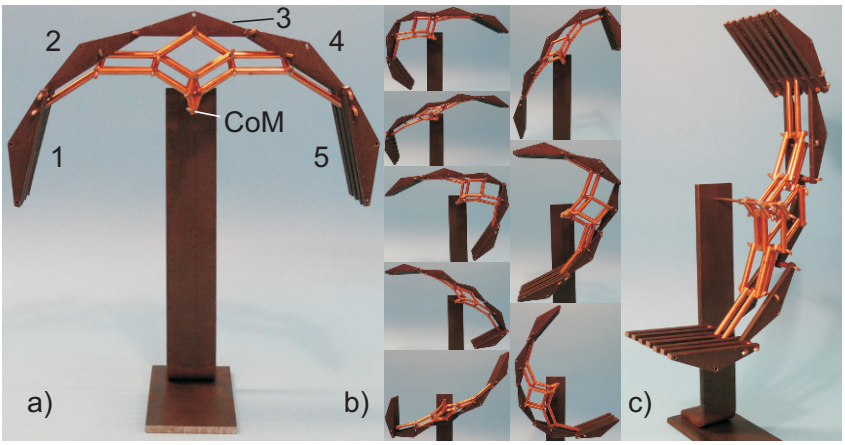


Fig. 6 Experiment with a mechanism model of a planar chain of five bodies.

4 Evaluation and Experiments

To verify the calculations, the mechanism model of Fig. 6 was developed. This mechanism has a serial chain of five bodies, of which the parameters are calculated equivalently to three or four bodies in series. The chain of five bodies consists of triangular elements of 4 mm thick steel, connected together at 60 mm distance from each other. For the ease of production, the parallelograms are made of Ø4 mm welding rod and each axis is made of Ø2 mm welding rod. For each rod the mass is then located halfway the joints.

The mass of each body 1 to 5 in Fig. 6a was chosen to be 120.61, 34.46, 17.23, 34.46 and 120.61 grams, respectively, but they were produced with errors of -1.27, 0.12, -0.12, -0.11, and -1.29 grams, respectively. Various mass values were chosen for each body to keep the dimensions of the parallelograms practical (not too small). For rigidity most of the rods are placed in couples (Fig. 6c).

The total mass of all moving elements is 428.21 grams. Since the five bodies of the serial chain weigh together 327.37 grams, all the parallelogram links and all axes then have increased the total mass by only 30%, which is remarkably low with respect to conventional balancing techniques using counter-masses that easily lead to an increase of at least 300% [5–7].

The mechanism remains steady in every pose. The stick-slip friction about the axis at the base was determined with measurement weights, being 0.00045 Nm maximally. Since the mechanism is steady in every pose this means that the CoM has a maximal error with the joint at the base of $0.00045\ \mathrm{Nm}/0.42821\ \mathrm{kg}/9.81\ \mathrm{ms}^{-2} = 0.11$ mm. Since this error is relatively small, the CoM will be very near to being stationary for which the mechanisms is force balanced. Figure 6b shows a selection of mechanism poses.

5 Conclusion

A method based on linear momentum equations and the use of Equivalent Linear Momentum Systems was proposed for the calculations of the principal points of planar serial chains with multiple bodies. Planar serial chains of three and four bodies were studied theoretically. The theory was validated by a mechanism model of a planar serial chain of five bodies that is steady in every pose. It was shown that the mass addition for balancing with Fischer's method is relatively low.

References

1. Lowen, G.G., Berkof, R.S.: Survey of investigations into the balancing of linkages. *Journal of Mechanisms* **3**, 221–231 (1968).
2. Raaijmakers, R.: Besi zoekt snelheidslimiet pakken en plaatsen op (Besi attacks the speedlimit for pick and place motion). *Mechatronica Nieuws* 26–31 (2007) [in Dutch].
3. Ishida, K., Matsuda, T.: Performance characteristics and working comfortableness of forest workers of a new non-vibrating chain saw utilizing perfectly balanced rotation-reciprocation device. In *Proceedings of the Fifth World Congress of Theory of Machines and Mechanisms*, ASME, pp. 951–954 (1979).
4. Brown, G.W.: Suspension system for supporting and conveying equipment, such as a camera, Patent No. US-4710819 (1987).
5. Van der Wijk, V., Herder, J.L., Demeulenaere, B.: Comparison of various dynamic balancing principles regarding additional mass and additional inertia. *Journal of Mechanisms and Robotics* **1**(4), 04 1006 (2009).
6. Van der Wijk, V., Herder, J.L., Demeulenaere, B., Gosselin, C.M.: Comparative analysis and optimization for low-mass and low-inertia dynamic balancing of a rotatable link balanced by a counter-rotary counter-mass. *Mechanisms and Robotics* (accepted) (2009).
7. Van der Wijk, V., Herder, J.L.: Guidelines for low mass and low inertia dynamic balancing of mechanisms and robotics. In Kröger and Wahl (Eds.), *Advances in Robotics Research, Proceedings of the German Workshop on Robotics*. Springer, pp. 21–30 (2009).
8. Fischer, O.: *Theoretische Grundlagen Für eine Mechanik der Lebenden Körper*. Teubner, Leipzig (1906).
9. Agrawal, S.K., Gardner, G., Pledgie, S.: Design and fabrication of an active gravity balanced planar mechanism using auxiliary parallelograms. *Journal of Mechanical Design* **123**, 525–528 (2001).

Singularity-Invariant Leg Rearrangements in Stewart–Gough Platforms

Júlia Borràs, Federico Thomas and Carme Torras

Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Llorens Artigas 4–6, 08028 Barcelona, Spain; e-mail: {jborras, fthomas, ctorras}@iri.upc.edu

Abstract. This work presents a necessary and sufficient condition to define a singularity-invariant leg rearrangement, based on an affine relation between the squared leg lengths before and after the rearrangement. This condition is then specified for four rigid components that can occur in Stewart–Gough platforms, leading to the characterization of singularity-invariant leg rearrangements on all of them.

Key words: Stewart–Gough platform, forward kinematics, singularities.

1 Introduction

The Stewart–Gough platform is defined as a 6-DoF parallel mechanism with six identical SPS legs [1, 2]. It has remained one of the most widely studied because, despite its geometric simplicity, its analysis translates into challenging mathematical problems. One important part of this analysis corresponds to the characterization of its singularities, which has only been completely solved for some specializations (for example, designs in which some spherical joints coalesce to form multiple spherical joints [3, 4]).

Finding leg rearrangements in a given Stewart–Gough platform that leave the singularity locus invariant does not solve the problem of characterizing singularities, but it provides a lot of insight that proves useful in several ways. For example, such leg rearrangements permit simplifying the platforms geometry to ease the task of obtaining the sought characterization of its singularity locus. On the other hand, if this locus is already characterized, modifying the placement of legs permits improving some platform characteristics (such as stiffness, avoidance of leg collisions or elimination of multiple spherical joints) without altering such locus. Eventually, by analyzing all possible leg rearrangements, one could identify all equivalent platforms.

In addition, singularity-invariant leg rearrangements provide a straightforward characterization of architectural singularities, promising a common framework to the extensive literature on this topic [5–9].

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_45,

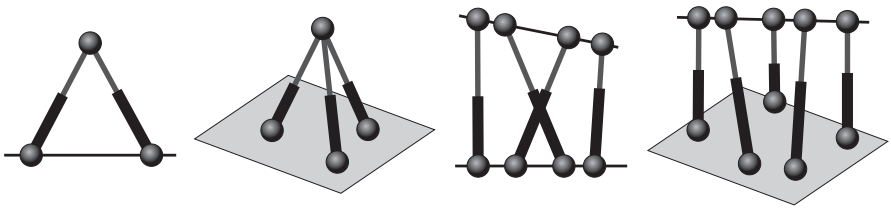


Fig. 1 The four possible rigid components involving linear geometric elements in Stewart–Gough platforms.

Now, let us suppose that we would like to apply a singularity-invariant leg rearrangement limited to a subset of legs. Clearly, this is only possible if this subset of legs defines a rigid subassembly. Kong and Gosselin refer to these subassemblies as *components* [10]. The simplest component arises when two legs share an attachment. The result is called the *Point-Line component*. Similarly, the three other components involving linear geometric entities (points, lines and planes) are the *Point-Plane*, *Line-Line* and *Line-Plane* components (Fig. 1).

Leg rearrangements were previously proposed by the authors for each of these components [11–13]. In this work a common framework for them all is provided in this way: all are shown to satisfy the same necessary and sufficient condition, confirming, through a unifying approach, that such rearrangements are indeed singularity-invariant.

This paper is organized as follows: Section 2 introduces the condition for singularity invariance. Then, in Sections 3 to 6, this condition is applied to each of the components in Fig. 1. Finally, Section 7 summarizes the main results and points out some future research directions.

2 Condition for Singularity Invariance

For a general Stewart–Gough platform, the linear actuators' velocities, $\dot{l}_1, \dot{l}_2, \ldots, \dot{l}_6$, can be expressed in terms of the platform velocity vector $(\mathbf{v}, \boldsymbol{\Omega})$ as follows:

$$\operatorname{diag}(l_1, \ldots, l_6) \begin{pmatrix} \dot{l}_1 \\ \dot{l}_2 \\ \vdots \\ \dot{l}_6 \end{pmatrix} = \mathbf{J} \begin{pmatrix} \mathbf{v} \\ \boldsymbol{\Omega} \end{pmatrix}, \tag{1}$$

where $\mathbf{J}$ is the matrix of normalized Plücker coordinates of the six leg lines [1]. The parallel singularities of the platform are those configurations in which $\det(\mathbf{J}) = 0$ [14].

Now, let us change the location of the leg attachments so that the lengths of the legs in their new locations, say $d_1, d_2, \ldots, d_6$, are related to those of the original legs,

$l_1, l_2, \ldots, l_6$, through the relation:

$$\begin{pmatrix} d_1^2 \\ d_2^2 \\ \vdots \\ d_6^2 \end{pmatrix} = \mathbf{A} \begin{pmatrix} l_1^2 \\ l_2^2 \\ \vdots \\ l_6^2 \end{pmatrix} + \mathbf{b} \tag{2}$$

where $\mathbf{A}$ and $\mathbf{b}$ are a constant matrix and a constant vector, respectively. Differentiating this equation with respect to time and substituting (1) in the result, we get

$$\operatorname{diag}(d_1, \ldots, d_6) \begin{pmatrix} \dot{d}_1 \\ \dot{d}_2 \\ \vdots \\ \dot{d}_6 \end{pmatrix} = \mathbf{A}\mathbf{J} \begin{pmatrix} \mathbf{v} \\ \boldsymbol{\Omega} \end{pmatrix}. \tag{3}$$

Then, the singularities of the platform after the leg rearrangement leading to (2) are those configurations in which $\det(\mathbf{A}\mathbf{J}) = \det(\mathbf{A})\det(\mathbf{J}) = 0$. If $\det(\mathbf{A}) \neq 0$, the leg rearrangement is said to be *singularity-invariant*. If $\det(\mathbf{A}) = 0$, the rearrangement introduces an *architectural singularity*, i.e., the resulting platform is always in a singularity independently of its leg lengths [5].

Since lengths are assumed to be positive magnitudes, Eq. (2) defines a one-to-one relationship between leg lengths before and after a singularity-invariant leg rearrangement. As a consequence, this kind of transformations leaves not only the singularities of the platform unaltered, but also the nature and number of its assembly modes.

3 Point-Line Component

A general leg rearrangement on the Point-Line component consists in the substitution of any leg by another one going from the point to the line (gray leg in Fig. 2, left). Considering the new leg length d, the following relation was proved in [12] using Heron's tetrahedron volume formula:

$$d^2 = \frac{nl_1^2 + ml_2^2 - mn}{m+n}. \tag{4}$$

As this is an affine relation, following Section 2 we can state that any leg rearrangement within a Point-Line component leaves singularities invariant.

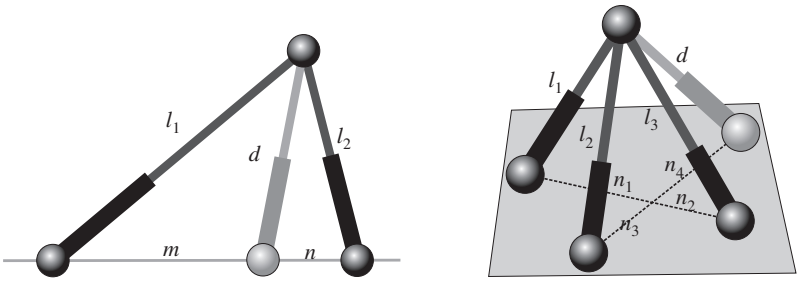


Fig. 2 A Point-Line component (left) and a Point-Plane component (right).

4 Point-Plane Component

Proceeding similarly, let us substitute a leg of the Point-Plane component by another one going from the vertex of the tripod to any point on the base plane (Fig. 2, right). The tripod contains three Point-Line components, so Eq. (4) can be used twice to obtain the following affine relation between the new leg length d and l_1, l_2 and l_3:

$$d^2 = \frac{n_2(n_3+n_4)}{n_3(n_1+n_2)}l_1^2 - \frac{n_4}{n_3}l_2^2 + \frac{n_1(n_3+n_4)}{n_3(n_1+n_2)}l_3^2 - \frac{n_1n_2(n_3+n_4)}{n_3} + n_4(n_3+n_4). \quad (5)$$

As a result, we can state that any leg rearrangement within a Point-Plane component leaves singularities invariant.

5 Line-Line Component

Following the notation introduced in Fig. 3 (left), suppose that we want to compute the length of a new leg between a point on the base line $\mathbf{a} = (x,0,0)$ and a point on the platform line $\mathbf{b} = \mathbf{p} + z\mathbf{i}$.

Taking the distance equations of the four legs $l_i^2 = \|\mathbf{b}_i - \mathbf{a}_i\|^2$, for $i = 1,\ldots,4$, together with that of the new leg $d^2 = \|\mathbf{b} - \mathbf{a}\|^2$, the subtraction of the equation $u^2 + v^2 + w^2 = 1$ cancels all quadratic terms in u, v and w, yielding

$$z_i\, t - x_i p_x - x_i p_z u + \frac{1}{2}(p_x^2 + p_y^2 + p_z^2 + x_i^2 + z_i^2 - l_i^2) = 0, \text{ for } i = 1,\ldots,4$$
$$z\, t - x\, p_x - x\, p_z u + \frac{1}{2}(p_x^2 + p_y^2 + p_z^2 + x^2 + z^2 - d^2) = 0,$$

where $t = \mathbf{p}\cdot\mathbf{i}$. In addition, subtracting the first equation from the others, quadratic terms in p_x, p_y and p_z cancel too, and the system becomes linear:

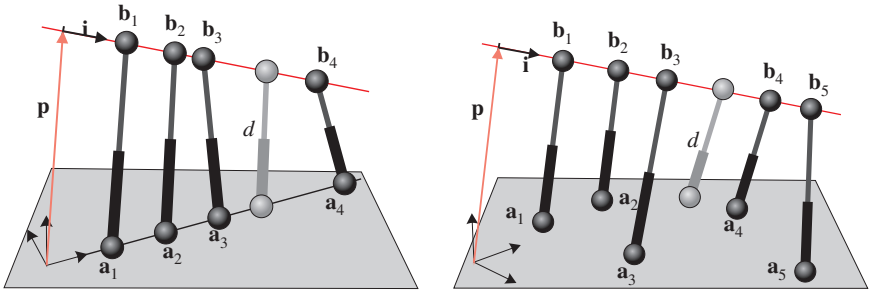


Fig. 3 On the base reference frame, local coordinates of the base attachments are $\mathbf{a}_i = (x_i, 0, 0)$ on the Line-Line component (left), and $\mathbf{a}_i = (x_i, y_i, 0)$ on the Line-Plane component (right). In both cases, the pose of the upper line with respect to the base plane can be described by the position vector $\mathbf{p} = (p_x, p_y, p_z)^T$ and the unit director vector of the line $\mathbf{i} = (u, v, w)^T$. Thus, the coordinates of the leg attachments in the platform line, expressed in the base reference frame, can be written as $\mathbf{b}_i = \mathbf{p} + z_i\mathbf{i}$, for $i = 1, \ldots, 4$ (left) and $i = 1, \ldots, 5$ (right).

$$\begin{pmatrix} z_1 - z_2 & x_2 - x_1 & x_2 z_2 - x_1 z_1 & 0 \\ z_1 - z_3 & x_3 - x_1 & x_3 z_3 - x_1 z_1 & 0 \\ z_1 - z_4 & x_4 - x_1 & x_4 z_4 - x_1 z_1 & 0 \\ z_1 - z & x - x_1 & xz - x_1 z_1 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} t \\ p_x \\ u \\ d^2 \end{pmatrix} = \begin{pmatrix} N_2 \\ N_3 \\ N_4 \\ N \end{pmatrix}, \tag{6}$$

where $N_i = 1/2(x_i^2 + z_i^2 - l_i^2 - x_1^2 - z_1^2 + l_1^2), i = 1, \ldots, 4$, and $N = 1/2(x^2 + z^2 - x_1^2 - z_1^2 + l_1^2)$ are constant. Now the expression for d^2 can be obtained by solving the system using Cramer's rule:

$$d^2 = \frac{\begin{vmatrix} -z_1 & x_1 & x_1 z_1 & x_1^2 + z_1^2 - l_1^2 & 1 \\ -z_2 & x_2 & x_2 z_2 & x_2^2 + z_2^2 - l_2^2 & 1 \\ -z_3 & x_3 & x_3 z_3 & x_3^2 + z_3^2 - l_3^2 & 1 \\ -z_4 & x_4 & x_4 z_4 & x_4^2 + z_4^2 - l_4^2 & 1 \\ -z & x & xz & x^2 + z^2 & 1 \end{vmatrix}}{\begin{vmatrix} z_1 - z_2 & x_2 - x_1 & x_2 z_2 - x_1 z_1 \\ z_1 - z_3 & x_3 - x_1 & x_3 z_3 - x_1 z_1 \\ z_1 - z_4 & x_4 - x_1 & x_4 z_4 - x_1 z_1 \end{vmatrix}}. \tag{7}$$

For any non-architecturally singular Line-Line component, the denominator is different from zero (in accordance with the condition found in [11]). Expanding the determinants involved in Eq. (7) leads to the affine relation

$$d^2 = c_1 l_1^2 + c_2 l_2^2 + c_3 l_3^2 + c_4 l_4^2 + c_0, \tag{8}$$

where all the coefficients depend on known constant coordinates. Thus, we can also state that any leg rearrangement within a Line-Line component leaves singularities invariant.

6 Line-Plane Component

Finally, let us consider the 5-legged parallel platform appearing in Fig. 3 (right). We proceed in a similar way as for the Line-Line component following the notation on the figure. The system derived from leg lengths has now 5 equations plus the one for the additionally introduced leg. After the same simplifications, the following linear system is obtained:

$$\begin{pmatrix} x_2-x_1 & y_2-y_1 & x_2z_2-x_1z_1 & y_2z_2-y_1z_1 & 0 \\ x_3-x_1 & y_3-y_1 & x_3z_3-x_1z_1 & y_3z_3-y_1z_1 & 0 \\ x_4-x_1 & y_4-y_1 & x_4z_4-x_1z_1 & y_4z_4-y_1z_1 & 0 \\ x_5-x_1 & y_5-y_1 & x_5z_5-x_1z_1 & y_5z_5-y_1z_1 & 0 \\ x-x_1 & y-y_1 & xz-x_1z_1 & yz-y_1z_1 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} p_x \\ p_y \\ u \\ v \\ d^2 \end{pmatrix} = \begin{pmatrix} (z_2-z_1)t+N_2 \\ (z_3-z_1)t+N_3 \\ (z_4-z_1)t+N_4 \\ (z_5-z_1)t+N_5 \\ (z-z_1)t+N \end{pmatrix}, \quad (9)$$

where now $N_i = 1/2(x_i^2+y_i^2+z_i^2-l_i^2-x_1^2-y_1^2-z_1^2+l_1^2), i=1,\ldots,5$, and $N = 1/2(x^2+y^2+z^2-x_1^2-y_1^2-z_1^2+l_1^2)$. We can always find a system matrix with a non-zero determinant for any non-architecturally singular Line-Plane component (see [13] for details).

Thus, using Cramer's rule again yields

$$d^2 = \frac{2(rt+s)}{C} \quad (10)$$

where C is the determinant of the matrix in (9) and r and s are the determinants

$$r = \begin{vmatrix} x_2-x_1 & y_2-y_1 & x_2z_2-x_1z_1 & y_2z_2-y_1z_1 & z_2-z_1 \\ x_3-x_1 & y_3-y_1 & x_3z_3-x_1z_1 & y_3z_3-y_1z_1 & z_3-z_1 \\ x_4-x_1 & y_4-y_1 & x_4z_4-x_1z_1 & y_4z_4-y_1z_1 & z_4-z_1 \\ x_5-x_1 & y_5-y_1 & x_5z_5-x_1z_1 & y_5z_5-y_1z_1 & z_5-z_1 \\ x-x_1 & y-y_1 & xz-x_1z_1 & yz-y_1z_1 & z-z_1 \end{vmatrix}, \quad (11)$$

and

$$s = \begin{vmatrix} x_2-x_1 & y_2-y_1 & x_2z_2-x_1z_1 & y_2z_2-y_1z_1 & N_2 \\ x_3-x_1 & y_3-y_1 & x_3z_3-x_1z_1 & y_3z_3-y_1z_1 & N_3 \\ x_4-x_1 & y_4-y_1 & x_4z_4-x_1z_1 & y_4z_4-y_1z_1 & N_4 \\ x_5-x_1 & y_5-y_1 & x_5z_5-x_1z_1 & y_5z_5-y_1z_1 & N_5 \\ x-x_1 & y-y_1 & xz-x_1z_1 & yz-y_1z_1 & N \end{vmatrix}. \quad (12)$$

Note that this is not an affine relation because it depends on t. However, if we impose $r=0$, then the resulting expression can be rewritten as

$$d^2 = \frac{1}{C} \begin{vmatrix} x & y & xz & yz & x^2+y^2+z^2 & 1 \\ x_1 & y_1 & x_1z_1 & y_1z_1 & x_1^2+y_1^2+z_1^2-l_1^2 & 1 \\ x_2 & y_2 & x_2z_2 & y_2z_2 & x_2^2+y_2^2+z_2^2-l_2^2 & 1 \\ x_3 & y_3 & x_3z_3 & y_3z_3 & x_3^2+y_3^2+z_3^2-l_3^2 & 1 \\ x_4 & y_4 & x_4z_4 & y_4z_4 & x_4^2+y_4^2+z_4^2-l_4^2 & 1 \\ x_5 & y_5 & x_5z_5 & y_5z_5 & x_5^2+y_5^2+z_5^2-l_5^2 & 1 \end{vmatrix}. \quad (13)$$

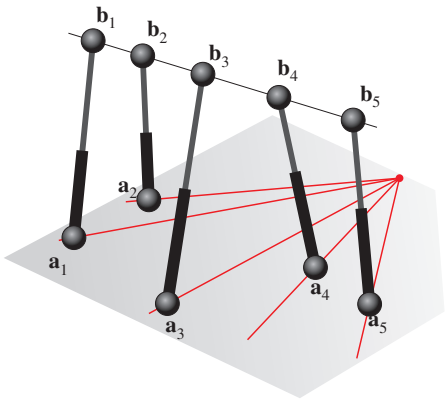


Fig. 4 Singularity-invariant leg rearrangements in the Line-Plane component must satisfy equation $r = 0$, where r is defined in Eq. (11).

After Laplace expansion by the elements of the 5th column, (13) leads to the affine relation

$$d^2 = c_1 l_1^2 + c_2 l_2^2 + c_3 l_3^2 + c_4 l_4^2 + c_5 l_5^2 + c_0, \tag{14}$$

where again all the coefficients depend on known constant coordinates.

In conclusion, the Line-Plane component is the first for which a general leg rearrangement is not necessarily singularity-invariant. To be so, the new leg attachments $\mathbf{a} = (x, y, 0)$ and $\mathbf{b} = \mathbf{p} + z\mathbf{i}$ must satisfy the equation $r = 0$.

Specific geometric rules to perform singularity-invariant leg rearrangements can be obtained from the equation $r = 0$. Indeed, it defines a one-to-one correspondence between points on the platform line and lines of a pencil on the base plane (Fig. 4). Thus, base attachments can always be moved within their corresponding lines. Furthermore, the vertex of the base pencil plays an important role in the characterization of the kinematics and singularities of the Line-Plane component (see [13, 15] for details).

7 Conclusions and Future Work

In this paper, a necessary and sufficient condition for a leg rearrangement in a Stewart–Gough platform to preserve its singularity locus has been derived. As long as an affine relation holds between the squared leg lengths before and after the rearrangement, the platform singularities remain unchanged. In other words, a leg rearrangement is singularity-invariant if, and only if, it induces an affine mapping in the joint space of the manipulator.

It has been shown that all leg rearrangements in the Point-Line, Point-Plane and Line-Line components of a Stewart–Gough platform (excluding those leading to an architecturally-singular configuration) satisfy this condition, therefore all being

singularity-invariant. In the case of the Line-Plane component, the condition holds only for a subset of leg rearrangements that has a neat geometric interpretation as previously shown. Thus, the present paper provides a common framework where the different singularity-preserving leg transformations proposed by the authors in previous works are viewed in a unified way.

As regards to future research, work on the double-planar Stewart–Gough platform is currently under way, suggesting interesting results for the general classification of all Stewart–Gough platforms.

It has also been briefly shown how the condition $\det(\mathbf{A}) = 0$ in Section 2 can characterize architectural singularities, thus work in this direction is in progress.

Acknowledgements This work has been partially supported by the Spanish Ministry of Education and Science, under the I+D project DPI2007-60858, and the Generalitat de Catalunya through the VALTEC program, cofinanced by FEDER funds.

References

1. Merlet, J.-P.: *Parallel Robots*. Springer (2000).
2. Dasguptaa, B., Mruthyunjayab, T.: The Stewart platform manipulator: A review. *Mech. Mach. Theory* **35**, 15–40 (2000).
3. Downing, D., Samuel, A., Hunt, K.: Identification of the special configurations of the octahedral manipulator using the pure condition. *Int. J. Robot. Res.* **21**, 147–159 (2002).
4. Alberich-Carramiñana, M., Thomas, F., Torras, C.: Flagged parallel manipulators. *IEEE Trans. Robot.* **23**(5), 1013–1023 (2007).
5. Ma, O., Angeles, J.: Architecture singularities of platform manipulators. In: *IEEE Int. Conf. Robot. Automat.*, Vol. 2, pp. 1542–1547 (1991).
6. Karger, A.: Architecture singular planar parallel manipulators. *Mech. Mach. Theory* **38**, 1149–1164 (2003).
7. Husty, M., Karger, A.: Architecture singular parallel manipulators and their self-motions. In: *Int. Sym. Adv. Robot Kinematics*, pp.355–364 (2000).
8. Wohlhart, K.: Synthesis of architecturally mobile double-planar platforms. In: *Int. Sym. Adv. Robot Kinematics*, pp. 473–482 (2002).
9. Rschel, O. Mick, S.: Characterisation of architecturally shaky platforms. In: *Int. Sym. Adv. Robot Kinematics*, pp. 465–474 (1998).
10. Kong, X., Gosselin, C.: Classification of 6-SPS parallel manipulators according to their components. In: *Proc. ASME Des. Eng. Tech. Conf.*, DETC2000/MECH-14105 (2000).
11. Borràs, J., Thomas, F., Torras, C.: Architecture singularities in flagged parallel manipulators. In: *IEEE Int. Conf. Robot. Automat.*, pp. 3844–3850 (2008).
12. Borràs, J., Thomas, F., Torras, C.: On Δ-transforms. *IEEE Trans. Robot.* **25**(6), 1225–1236 (2009).
13. Borràs, J., Thomas, F.: Kinematics of the line-plane subassembly in Stewart platforms. In: *IEEE Int. Conf. Robot. Automat.*, pp. 4094–4099 (2009).
14. Gosselin, C. Angeles, J.: Singularity analysis of closed-loop kinematic chains. *IEEE Trans. Robot.* **6**(3), 281–290 (1990).
15. Borràs, J., Thomas, F., Torras, C.: A family of quadratically-solvable 5-UPS parallel robots. In: *IEEE Int. Conf. Robot. Automat.*, to appear (2010).

Geometric Kinematics of Rigid Bodies with Point Contact

L. Cui and J.S. Dai

King's College London, United Kingdom; e-mail: {lei.cui, jian.dai}@kcl.ac.uk

Abstract. This paper applies Darboux frame method to developing geometric kinematics of sliding-spin-rolling motion of rigid objects with point contact. For the first time, the geodesic curvatures, normal curvatures and geodesic torsions of both the sliding motion and rolling motion are derived in terms of known geometric entities. The geometric kinematics of the moving object is represented with geometric invariants. Effect of the relative curvatures and torsion on sliding-spin-rolling kinematics is explicitly presented.

Key words: Kinematics, point contact, sliding-spin-rolling motion, Darboux frame, coordinate-invariant.

1 Introduction

A general point contact motion is a combination of sliding, spin, and rolling motion which can be decomposed into a translational sliding motion at the contact point and a rotational spin-rolling motion about the contact point. Kinematics of point contact has attracted many kinematicians. Cai and Roth [1] investigated kinematics of rigid objects in point contact in general spatial motion and concentrated on two special motions including sliding and pure-rolling motion. Montana [2] studied the kinematics of contact from a geometric point of view and derived the equations of contact. Chen [3] developed a purely geometric form of motion representation. Ghafoor, Dai, and Duffy [4] simplified kinematics of rolling contact to line contact. Cui and Dai [5, 6] developed coordinate-invariant instantaneous kinematics of rolling contact and discussed the implications for trajectory planning.

Previous literature on sliding-spin-rolling motion of rigid bodies with point contact suffers from some drawbacks. First, previous formulations were local, which means that once the origin or orientation of the coordinates changes, the whole formulation has to be re-established. Secondly, there is a misconception about spin motion. Though spin motion does not contribute to rolling contact trajectory curves, it is subject to geometric constraints. Thirdly, the formulations can only be differentiated to a certain order, usually one or two.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_46,

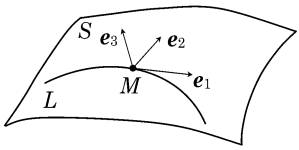


Fig. 1 A Darboux frame at point *M*.

This paper starts with sliding motion and proceeds to sliding-spin-rolling motion. The geometric entities of rolling and sliding motions are for the first time derived. The paper does not take time into consideration, so that the paper can have a broader scope. The proposed formulation is coordiante-invariant and can be differentiated to any order.

2 Darboux Frame and Darboux Vector

Here a brief introduction is given concerning some basic concepts of differential geometry. Details can be found in [7–9].

Let L be an oriented curve traced on an oriented surface S in E^3, Darboux frame $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ is a right-handed orthonormal frame associated with each point M on curve L, where $\mathbf{e}_1$ is the unit tangent vector to emphL; $\mathbf{e}_3$ is the unit normal to S; $\mathbf{e}_2$ is tangential to S as in Fig. 1.

The equations of motion of the Darboux frame are

$$\begin{aligned} \frac{d\mathbf{M}}{ds} &= \mathbf{e}_1 \\ \begin{pmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{pmatrix} &= \begin{pmatrix} 0 & k_g & k_n \\ -k_g & 0 & \tau_g \\ -k_n & -\tau_g & 0 \end{pmatrix} \begin{pmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{pmatrix} \end{aligned} \tag{1}$$

where k_g, k_n, and τ_g are called geodesic curvature, normal curvature, and geodesic torsion respectively. The second equation in Eq. (1) can also be written in matrix form as

$$\frac{d\mathbf{E}}{ds} = \mathbf{TE} \tag{2}$$

The three vectors $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$ in Eq. (1) are contravariant vectors, or type (1, 0) tensors. They do not have any intrinsic coordinate system. The components of these contravariant vectors will transform in a certain way passing from one coordinate system to another. The four scalars s, k_g, k_n and τ_g in Eq. (1) are geometric invariants which remain identical regardless of coordinate transformations.

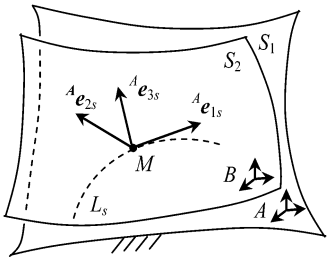


Fig. 2 The sliding contact trajectory curve L_s on the surface S_1.

3 Sliding Motion

Since this paper is to study the relative motion between rigid object A and object B, Object A can be assumed to be fixed and object B to be moving. Set up fixed coordinate frame $\{A\}$ on surface S_1 of object A and $\{B\}$ on surface S_2 of object B, a pre-subscript A or B will be used to denote in which coordinate system a vector is expressed. Sliding motion is a translational motion with two degrees of freedom. It occurs in the common tangent plane of the two surfaces. Sliding motion between two rigid bodies produces one sliding trajectory curve L_s on the fixed surface S_1. Set up a moving frame $\{{}^A\mathbf{e}_{1s}, {}^A\mathbf{e}_{2s}, {}^A\mathbf{e}_{3s}\}$ at the contact point M of L_s with ${}^A\mathbf{e}_{1s}$ being the unit tangent vector to curve L_s and ${}^A\mathbf{e}_{3s}$ the unit normal vector to surface S_1 as in Fig. 2.

Let s_s denote the arc length of the sliding contact trajectory L_s, then it follows from Eq. (1) that

$$\frac{d^A\mathbf{M}}{ds_s} = {}^A\mathbf{e}_{1s} \tag{3}$$

4 Spin-Rolling Motion

Kinematics of spin-rolling motion has been reported in [5, 6]. Some results are presented here for the completeness of this paper. Spin-rolling motion produces one contact trajectory curve L'_r on the surface S_2 and one contact trajectory curve L_r on the surface S_1. Set up moving frame $\{{}^A\mathbf{e}_{1r}, {}^A\mathbf{e}_{2r}, {}^A\mathbf{e}_{3r}\}$ of the curve L_r and $\{{}^B\mathbf{e}_{1r}, {}^B\mathbf{e}_{2r}, {}^B\mathbf{e}_{3r}\}$ of the curve L'_r at the contact point M as in Fig. 3. Due to the constraints of rolling contact, these two frames can be made to coincide at any time.

The geometric angular velocity of the moving object in terms of the arc length s_r can be obtained as

$${}^A\omega_r = -\tau^{*A}_{gr}\mathbf{e}_{1r} + k^{*A}_{nr}\mathbf{e}_{2r} - k^{*A}_{gr}\mathbf{e}_{3r} \tag{4}$$

where

$$k^*_{gr} = k'_{gr} - k_{gr}, \quad k^*_{nr} = k'_{nr} - k_{nr}, \quad \tau^*_{gr} = \tau'_{gr} - \tau_{gr} \tag{5}$$

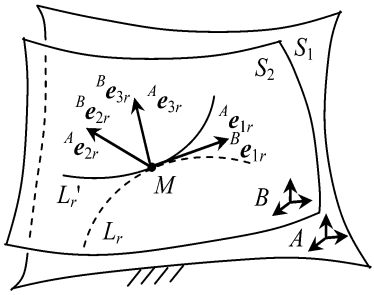


Fig. 3 Contact trajectory curves of spin-rolling motion.

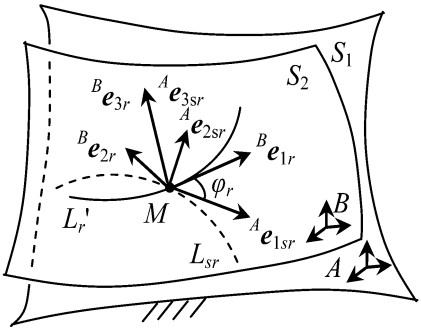


Fig. 4 The contact curves of sliding-spin-rolling motion.

While k'_{gr}, k'_{nr}, τ'_{gr} are the respective geodesic curvature, normal curvature, and geodesic torsion at point M of the contact trajectory curve L'_r on the surface S_2, k_{gr}, k_{nr}, τ_{gr} are the respective geodesic curvature, normal curvature, and geodesic torsion at point M of the contact trajectory curve L_r on the surface S_1.

5 Sliding-Spin-Rolling Motion

Sliding-spin-rolling motion is a general spatial motion of five degrees of freedom subject to geometric constraints of point contact. The contact point M traces one rolling trajectory curve L'_r on the moving surface S_2 and one sliding-rolling contact trajectory curve L_{sr} on the fixed surface S_1 as in Fig. 4.

The trajectory curve L_{sr} is produced by both sliding and rolling motion. Hence at any instant the curve L_{sr} can be regarded as a combination of one instantaneous sliding curve L_s of sliding motion and one instantaneous rolling curve L_r of rolling motion at point M of the surface S_1 as in Fig. 5. Note that the two instantaneous curves L_s and L_r do not exist globally, since the two curves meet at every contact point M of curve L_{sr}.

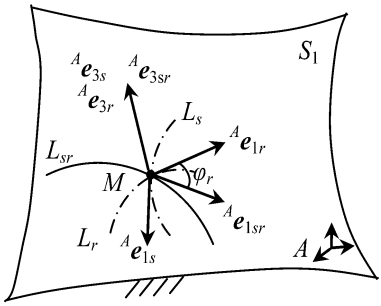


Fig. 5 Contact trajectory curves of sliding-spin-rolling motion on the surface S_1.

The only known geometric entities are from the two known curves L_{sr} and L'_r, for example, through using tactile sensors. It is necessary to derive other geometric entities of the two instantaneous curves L_s and L_r as in Fig. 5 from the known entities.

To obtain the unknown geometric entities, set moving frame $\{{}^A\mathbf{e}_{1sr}, {}^A\mathbf{e}_{2sr}, {}^A\mathbf{e}_{3sr}\}$ of the curve L_{sr} and moving frame $\{{}^A\mathbf{e}_{1s}, {}^A\mathbf{e}_{2s}, {}^A\mathbf{e}_{3s}\}$ of the curve L_s and moving frame $\{{}^A\mathbf{e}_{1r}, {}^A\mathbf{e}_{2r}, {}^A\mathbf{e}_{3r}\}$ of the curve L_r at the contact point M on the surface S_1, where the normal vectors ${}^A\mathbf{e}_{3sr}$, ${}^A\mathbf{e}_{3r}$, and ${}^A\mathbf{e}_{3s}$ are perpendicular to the surface S_1 and thus they coincide. The vectors ${}^A\mathbf{e}_{1sr}$, ${}^A\mathbf{e}_{1s}$ and ${}^A\mathbf{e}_{1r}$ are tangent to the respective curve L_{sr}, L_s and L_r as in Fig. 5.

The rolling contact trajectory curve L'_r on the surface S_2 is solely produced by rolling motion. Set up moving frame $\{{}^B\mathbf{e}_{1r}, {}^B\mathbf{e}_{2r}, {}^B\mathbf{e}_{3r}\}$ at the contact point M of the curve L'_r on the surface S_2, where the vector ${}^B\mathbf{e}_{3r}$ is perpendicular to the surface S_2 and ${}^B\mathbf{e}_{1r}$ is tangent to the curve L'_r as in Fig. 4.

Due to rolling constraints, the moving frame $\{{}^A\mathbf{e}_{1r}, {}^A\mathbf{e}_{2r}, {}^A\mathbf{e}_{3r}\}$ of the instantaneous curve L_r on the surface S_1 can be always made to coincide with the moving frame $\{{}^B\mathbf{e}_{1r}, {}^B\mathbf{e}_{2r}, {}^B\mathbf{e}_{3r}\}$ of the curve L'_r on the surface S_2. Since sliding motion and rolling motion are independent, so there generally exists an angle φ_r between the curves L_{sr} and L'_r, and subsequently between the vectors ${}^A\mathbf{e}_{1sr}$ and ${}^B\mathbf{e}_{1r}$ as in Fig. 4. It can be deduced that the angle between ${}^A\mathbf{e}_{1sr}$ and ${}^A\mathbf{e}_{1r}$ is also φ_r as in Fig. 5, since moving frames $\{{}^A\mathbf{e}_{1r}, {}^A\mathbf{e}_{2r}, {}^A\mathbf{e}_{3r}\}$ and $\{{}^B\mathbf{e}_{1r}, {}^B\mathbf{e}_{2r}, {}^B\mathbf{e}_{3r}\}$ coincide.

5.1 Derivation of Geometric Invariants of Rolling Motion

From Fig. 5, it can be seen that the moving frame $\{{}^A\mathbf{e}_{1r}, {}^A\mathbf{e}_{2r}, {}^A\mathbf{e}_{3r}\}$ can be obtained from $\{{}^A\mathbf{e}_{1sr}, {}^A\mathbf{e}_{2sr}, {}^A\mathbf{e}_{3sr}\}$ by a rotation of angle φ_r about ${}^A\mathbf{e}_{3sr}$. It follows that

$$
\begin{aligned}
{}^{A}\mathbf{e}_{1r} &= \cos\varphi_r^{A}\mathbf{e}_{1sr} + \sin\varphi_r^{A}\mathbf{e}_{2sr} \\
{}^{A}\mathbf{e}_{2r} &= -\sin\varphi_r^{A}\mathbf{e}_{1sr} + \cos\varphi_r^{A}\mathbf{e}_{2sr} \\
{}^{A}\mathbf{e}_{3r} &= {}^{A}\mathbf{e}_{3sr}
\end{aligned} \tag{6}
$$

Eq. (6) can be written in matrix form as

$$
\mathbf{E}_r = \mathbf{R}_r\mathbf{E}_{sr} \tag{7}
$$

where

$$
\mathbf{E}_r = \begin{pmatrix} {}^{A}\mathbf{e}_{1r} \\ {}^{A}\mathbf{e}_{2r} \\ {}^{A}\mathbf{e}_{3r} \end{pmatrix}, \quad \mathbf{R}_r = \begin{pmatrix} \cos\varphi_r & \sin\varphi_r & 0 \\ -\sin\varphi_r & \cos\varphi_r & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{E}_{sr} = \begin{pmatrix} {}^{A}\mathbf{e}_{1sr} \\ {}^{A}\mathbf{e}_{2sr} \\ {}^{A}\mathbf{e}_{3sr} \end{pmatrix} \tag{8}
$$

From Eq. (2), differentiating the left side of Eq. (7) with respect to s_r yields

$$
\frac{d\mathbf{E}_r}{ds_r} = \mathbf{T}_r\mathbf{E}_r = \mathbf{T}_r\mathbf{R}_r\mathbf{E}_{sr} \tag{9}
$$

where $\mathbf{T}_r$ contains unknown geometric entities k_{gr}, k_{nr}, and τ_{gr} of the instantaneous rolling curve L_r. Let s_{sr} denote the arc length of the trajectory curve L_{sr}, differentiating the right side of Eq. (7) with respect to s_r yields

$$
\frac{d}{ds_r}(\mathbf{R}_r\mathbf{E}_{sr}) = \lambda\left(\frac{d\mathbf{R}_r}{ds_{sr}} + \mathbf{R}_r\mathbf{T}_{sr}\right)\mathbf{E}_{sr} \tag{10}
$$

where $\lambda = ds_{sr}/ds_r$ is the sliding ratio and $\mathbf{T}_{sr}$ contains known geometric entities k_{gsr}, k_{nsr}, and τ_{gsr} of the contact curve L_{sr}. From Eqs. (9) and (10), it can be seen that

$$
\mathbf{T}_r\mathbf{R}_r = \lambda\left(\frac{d\mathbf{R}_r}{ds_{sr}} + \mathbf{R}_r\mathbf{T}_{sr}\right) \tag{11}
$$

Right-multiplying both the left and right side of Eq. (11) with $\mathbf{R}_r^{-1}$ yields

$$
\mathbf{T}_r = \lambda\left(\frac{d\mathbf{R}_r}{ds_{sr}} + \mathbf{R}_r\mathbf{T}_{sr}\right)\mathbf{R}_r^{-1} \tag{12}
$$

The unknown geometric entities in $\mathbf{T}_r$ can be obtained by computing the right side of Eq. (12) as

$$
\begin{aligned}
k_{gr} &= \lambda\left(\frac{d\varphi_r}{ds_{sr}} + k_{gsr}\right), \quad k_{nr} = \lambda(\cos\varphi_r k_{nsr} + \sin\varphi_r\tau_{gsr}), \\
\tau_{gr} &= \lambda(-\sin\varphi_r k_{nsr} + \cos\varphi_r\tau_{gsr})
\end{aligned} \tag{13}
$$

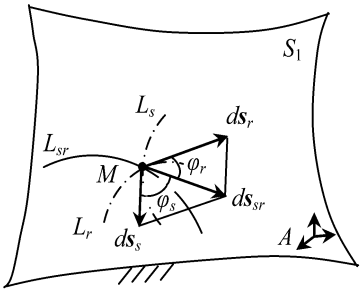


Fig. 6 Relationship between sliding motion and spin-rolling-sliding motion.

5.2 Derivation of Geometric Invariants of Sliding Motion

Let s_s denote the arc length of the instantaneous sliding curve L_s. The variation of arc lengths s_r of L_r, s_s of L_s, and s_{sr} of L_{sr} satisfy the geometric constraint as in Figure 6, where φ_s denote the angle between $d\mathbf{s}_{sr}$ and $d\mathbf{s}_s$.

It follows that

$$d\mathbf{s}_s = d\mathbf{s}_{sr} - d\mathbf{s}_r \tag{14}$$

The variation of arc length s_s can be obtained as

$$ds_s = \sqrt{ds_{sr}^2 + ds_r^2 - 2ds_{sr}ds_r\cos\varphi_r} \tag{15}$$

where s_{sr}, s_r and φ_r are known geometric entities. The sine rule from trigonometry formulae gives the angle φ_s between ds_{sr} and ds_s as

$$\varphi_s = \arcsin\left(\frac{ds_r}{ds_s}\sin\varphi_r\right) = \arcsin\left(\frac{\sin\varphi_r}{\sqrt{\lambda^2+1-2\lambda\cos\varphi_r}}\right) \tag{16}$$

The geometric invariants of L_s can be obtained in a similar way to the derivation of the geometric invariants of the instantaneous rolling curve L_r.

5.3 Geometric Velocity of Sliding-Spin-Rolling Motion

After all the geometric entities are derived, the geometric velocity of sliding-spin-rolling motion can be obtained in the frame $\{{}^A\mathbf{e}_{1sr}, {}^A\mathbf{e}_{2sr}, {}^A\mathbf{e}_{3sr}\}$ on the fixed surface S_1. From Eq. (3) the sliding velocity of the contact point M in terms of arc length s_{sr} is

$$\mathbf{v}_M = \frac{d^A\mathbf{M}}{ds_{sr}} = \frac{ds_s}{ds_{sr}}\frac{d^A\mathbf{M}}{ds_s} = \frac{\sqrt{\lambda^2+1-2\lambda\cos\varphi_r}}{\lambda}(\cos\varphi_s^A\mathbf{e}_{1sr} - \sin\varphi_s^A\mathbf{e}_{2sr}) \tag{17}$$

The rolling motion gives the angular velocity, which can be obtained from Eq. (4) as

$$\begin{aligned}\omega &= -\tau_{gr}^*(\cos\varphi_r {}^A\mathbf{e}_{1sr} + \sin\varphi_r {}^A\mathbf{e}_{2sr}) + k_{nr}^*(-\sin\varphi_r {}^A\mathbf{e}_{1sr} + \cos\varphi_r {}^A\mathbf{e}_{2sr}) - k_{nr}^* {}^A\mathbf{e}_{3sr} \\ &= (-\tau_{gr}^*\cos\varphi_r - k_{nr}^*\sin\varphi_r) {}^A\mathbf{e}_{1sr} + (-\tau_{gr}^*\sin\varphi_r + k_{nr}^*\cos\varphi_r) {}^A\mathbf{e}_{1sr} - k_{gr}^* {}^A\mathbf{e}_{3sr}\end{aligned} \tag{18}$$

The above two equations give the instantaneous geometric velocity of the moving object with point contact. It can be seen that the velocity formulations are expressed in contravariant vectors and scalars. Thus, the formulations are coordinate-invariant and can be differentiated to any order.

6 Conclusions

This paper investigated geometric kinematics of sliding-spin-rolling motion of rigid bodies with point contact. Without taking time into consideration, the paper had a broader scope to discuss the effects of curvatures and torsions on sliding-spin-rolling motion. For the first time geometric entities of sliding and rolling motion were derived in terms of arc lengths, curvatures, and torsions of two known contact trajectory curves. Finally geometric velocity of the moving object was expressed in terms of geometric invariants and contravariant vectors. The results were coordinate-invariant and can be generalized to arbitrary curves and surfaces.

References

1. Cai, C., Roth, B.: On the spatial motion of rigid bodies with point contact. In: *IEEE Conference on Robotics and Automation*, pp. 686–695 (1987).
2. Montana, D.J.: The kinematics of contact and grasp. *The International Journal of Robotics Research* **7**, 17–32 (1988).
3. Chen, C.H.: Conjugation form of motion representation and its conversion formulas. *Mechanism and Machine Theory* **32**, 765–774 (1997).
4. Ghafoor, A., Dai, J.S., Duffy J.: Fine motion control based on constraint criteria under preloading configurations. *Journal of Robotic Systems* **17**, 171–185 (2000).
5. Cui, L., Dai, J.S.: A coordinate-free approach to instantaneous kinematics of two rigid objects with rolling contact and its implications for trajectory planning. In: *IEEE Conference on Robotics and Automation*, pp. 612–617 (2009).
6. Cui, L., Dai, J.S.: A Darboux frame based formulation of spin-rolling motion of rigid objects with point contact. *IEEE Transactions on Robotics* **26**, (2010).
7. Carmo, M.P.: *Differential Geometry of Curves and Surfaces*. Prentice-Hall, New York (1976).
8. Cartan, E.: *Riemannian Geometry in an Orthogonal Frame*. World Scientific Press, Singapore (2001).
9. Cartan, H.: *Differential Forms*. Dover Publisher, New York (1996).

Singularity Locus of 6–4 Fully-Parallel Manipulators

R. Di Gregorio

Department of Engineering, University of Ferrara, Ferrara, Italy;
e-mail: raffaele.digregorio@unife.it

Abstract. This paper presents a new analytic expression for the singularity locus of 6–4 fully-parallel manipulators (FPMs) with two double spherical pairs. The new expression allows the separation of the geometric constant parameters from the configuration dependent ones, and it is specially useful for designing manipulators referable to this architecture. The presented expression is deduced from a singularity-locus expression previously proposed by the author for the general Gough–Stewart platform, and it highlights that the shape of the singularity locus depends on the properties of a 3×3 matrix.

Key words: Parallel manipulator, instantaneous kinematics, singular configuration, singularity locus.

1 Introduction

Parallel manipulators (PMs) feature a mobile platform connected to the frame (base) through a number of kinematic chains (limbs). Their architecture, when correctly exploited, allows the design of fast, stiff and precise manipulators that control the platform pose (position and orientation) in a workspace which, in general, is small with respect to the overall size of the machine.

A particular family of six degrees-of-freedom (dof) PMs is constituted by fully-parallel manipulators (FPMs) [1]. FPMs are characterized by six limbs of type U<u>P</u>S (U, P, and S stand for universal joint, prismatic pair and spherical pair, respectively; the underscoring indicates the actuated joint). The universal-joint (spherical-pair) centers are fixed points of the base (of the platform), and, hereafter, they will be called attachment points. In each limb, the actuated prismatic pair controls the distance between the limb's attachment points, and the combined action of all the six actuated pairs makes it possible to control the relative pose between platform and base.

Different FPM architectures are obtained by making two or more attachment points coincide either in the platform or in the base, and they can be collected into subfamilies named *n–m*, where *n* and *m* stand for the number of distinct attachment

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_47,

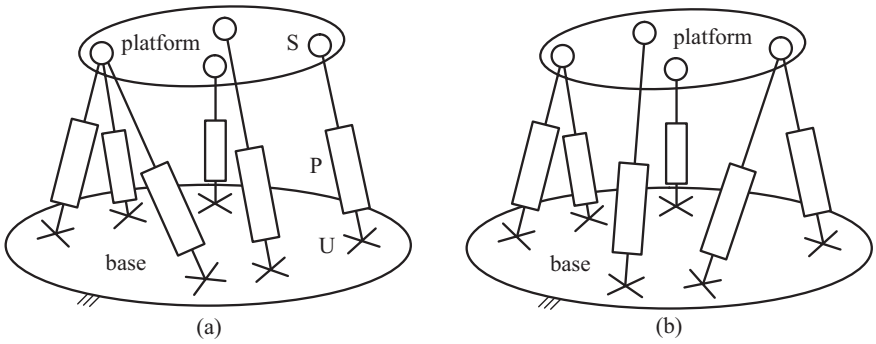


Fig. 1 6–4 FPM architectures: (a) spherical Stewart platform, (b) architecture with two double and two single spherical pairs.

points in the base and in the platform, respectively. The most general architecture is the 6–6 one, also named the general Gough–Stewart platform [2]. All the analytic relationships that hold for 6–6 FPMs can be adapted for *n*–*m* FPMs by suitably changing the geometric constant parameters that appear in them. Nevertheless, in general, the simple variation of these parameters yields expressions that do not highlight the peculiarities of the particular architecture, and specific analyses and/or rearrangements of the deduced formulas are necessary to understand how the architecture behaves.

6–4 FPMs refer to two different architectures (Fig. 1). One architecture [3], also named spherical Stewart platform [4], exhibits one triple and three single spherical pairs (Fig. 1a), whereas the other [5,6] exhibits two double and two single spherical pairs (Fig. 1b).

Designing PMs involves the solution of problems that are specific of closed-loop mechanisms. The identification of the singular configurations (parallel singularities) of direct instantaneous kinematics is one of these problems. Parallel singularities are manipulator configurations where the platform can perform instantaneous motions even though the actuators are locked, which involves that the platform pose cannot be controlled. At a parallel singularity, infinite generalized torques applied by the actuators are necessary to equilibrate some types of external loads exerted on the platform (i.e., the manipulator breaks down). These control and static troubles justify their identification during design and their avoidance during operation.

The singularity loci of FPMs have been studied by many researchers (see, for instance [2,4,7–10]). Most of them refer to 6–6 FPMs [2,8–10], others are devoted to particular *n*–*m* FPMs [4,7].

Regarding the two 6–4 FPM architectures, the singularity locus of the spherical Stewart platform was presented in [4] and, successively, deduced from the general expression of 6–6 FPMs' singularity locus in [8]. The singularity locus of the other 6–4 architecture was not investigated in depth. It will be studied in this paper.

Here, a new analytic expression will be presented for this locus. This expression separates the geometric constant parameters from the configuration dependent

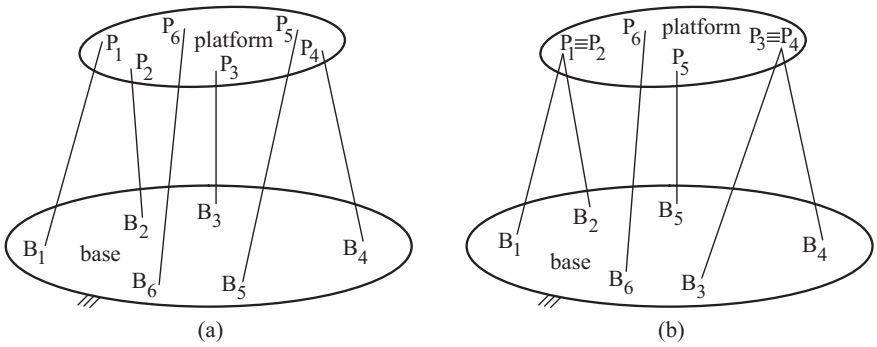


Fig. 2 Notations: (a) 6–6 FPM, (b) 6–4 FPM with two double and two single spherical pairs.

ones, and it is specially useful for designing manipulators referable to this architecture. The analysis of this expression will show that the locus' shape depends on the properties of a 3×3 matrix. In Section 2, an intermediate expression is deduced from a singularity-locus expression previously proposed by the author for the 6–6 FPMs [8]. In Section 3, the intermediate expression will be manipulated till to the final expression. Finally, Sections 4 and 5 will discuss the deduced expression and will offer the concluding remarks.

2 Notations and Preliminary Computations

Let B_i and P_i, $i = 1,\ldots,6$, be the attachment points of the ith limb in the base and the platform, respectively (Fig. 2). Hereafter, the length of segment $\overline{B_iP_i}$ and the line through B_i and P_i will be called limb length and limb axis, respectively, of the ith limb.

The following vectors and scalars are defined:

$$\mathbf{u}_i = \mathbf{P}_i - \mathbf{B}_i, \qquad i = 1,\ldots,6; \tag{1a}$$

$$\mathbf{v}_l = (\mathbf{P}_l - \mathbf{P}_1) \times \mathbf{u}_l, \qquad l = 2,\ldots,6; \tag{1b}$$

$$\mathrm{u}_{ijk} = \mathbf{u}_i \cdot \mathbf{u}_j \times \mathbf{u}_k, \qquad i,j,k \in \{1,\ldots,6\}; \tag{1c}$$

$$\mathrm{v}_{lmn} = \mathbf{v}_l \cdot \mathbf{v}_m \times \mathbf{v}_n, \qquad l,m,n \in \{2,\ldots,6\}. \tag{1d}$$

By using a suitable Laplace expansion [11] of the Jacobian determinant, these notations and definitions made it possible (see [8]) to put the singularity-locus equation of 6–6 FPMs (Fig. 2a) in the form:

$$\begin{aligned}\mathrm{u}_{123}\mathrm{v}_{456} - \mathrm{u}_{124}\mathrm{v}_{356} + \mathrm{u}_{125}\mathrm{v}_{346} - \mathrm{u}_{126}\mathrm{v}_{345} + \mathrm{u}_{134}\mathrm{v}_{256} - \mathrm{u}_{135}\mathrm{v}_{246} \\ + \mathrm{u}_{136}\mathrm{v}_{245} + \mathrm{u}_{145}\mathrm{v}_{236} - \mathrm{u}_{146}\mathrm{v}_{235} + \mathrm{u}_{156}\mathrm{v}_{234} = 0.\end{aligned} \tag{2}$$

Equation (2) can be used to deduce singularity-locus equations of particular *n–m* FPMs by introducing the specific values of their geometric constant parameters.

6–4 FPMs with the architecture of Fig. 2b are characterized by the coincidence of P_1 with P_2 and of P_3 with P_4. The coincidence of P_1 with P_2 makes $\mathbf{v}_2$ vanish. As a consequence, all the v_{lmn} with at least one of the right subscripts, l, m, and n, equal to 2 vanish, too. The introduction of these values into Eq. (2) yields the following singularity-locus equation, for 6–4 FPMs with two double and two single spherical pairs:

$$\det(\mathbf{J}) = 0, \tag{3}$$

with

$$\det(\mathbf{J}) = \mathrm{u}_{123}\mathrm{v}_{456} - \mathrm{u}_{124}\mathrm{v}_{356} + \mathrm{u}_{125}\mathrm{v}_{346} - \mathrm{u}_{126}\mathrm{v}_{345}. \tag{4}$$

3 New Expression of the Singularity-Locus Equation

All the four monomials that appear at the left-hand side, $\det(\mathbf{J})$, of Eq. (3) contain a factor of type u_{12k}, $k \in \{3,4,5,6\}$. This fact makes it possible to rearrange the expression of $\det(\mathbf{J})$ as follows:

$$\det(\mathbf{J}) = \mathbf{n}_{12} \cdot \mathbf{s}_{34} \tag{5}$$

where

$$\mathbf{n}_{12} = \mathbf{u}_1 \times \mathbf{u}_2, \tag{6a}$$

$$\mathbf{s}_{34} = \mathbf{u}_3\mathrm{v}_{456} - \mathbf{u}_4\mathrm{v}_{356} + \mathbf{u}_5\mathrm{v}_{346} - \mathbf{u}_6\mathrm{v}_{345}. \tag{6b}$$

Vector $\mathbf{n}_{12}$ is perpendicular to the plane the two intersecting axes of the limbs 1 and 2 (Fig. 2b) lie on, whereas $\mathbf{s}_{34}$ is a vector whose magnitude and direction depend on the platform geometry and on the axes of the limbs $3,4,5$, and 6. If either these two vectors are perpendicular to one another, or at least one of them is a null vector, Eq. (3) will be satisfied and the manipulator will be at a singularity.

Definition (6b) can be further elaborated to separate the effects of the platform geometry and of the limb axes. The first two terms that appear in (6b) can be transformed as follows (see Appendix A)

$$\mathbf{u}_3\mathrm{v}_{456} - \mathbf{u}_4\mathrm{v}_{356} = \mathbf{h} \times \mathbf{n}_{34} \tag{7}$$

with

$$\mathbf{h} = (\mathbf{v}_5 \times \mathbf{v}_6) \times (\mathbf{P}_3 - \mathbf{P}_1) \equiv (\mathbf{v}_5 \times \mathbf{v}_6) \times (\mathbf{P}_4 - \mathbf{P}_1), \tag{8a}$$

$$\mathbf{n}_{34} = \mathbf{u}_3 \times \mathbf{u}_4; \tag{8b}$$

whereas the algebraic manipulation of the last two terms yields (see Appendix B)

$$\mathbf{u}_5 v_{346} - \mathbf{u}_6 v_{345} = \mathbf{q}\,[(\mathbf{P}_3 - \mathbf{P}_1)\cdot\mathbf{n}_{34}] \equiv \mathbf{q}\,[(\mathbf{P}_4 - \mathbf{P}_1)\cdot\mathbf{n}_{34}] \tag{9}$$

with

$$\begin{aligned}\mathbf{q} &= \mathbf{u}_5[\mathbf{v}_6\cdot(\mathbf{P}_3 - \mathbf{P}_1)] - \mathbf{u}_6[\mathbf{v}_5\cdot(\mathbf{P}_3 - \mathbf{P}_1)] \\ &\equiv \mathbf{u}_5[\mathbf{v}_6\cdot(\mathbf{P}_4 - \mathbf{P}_1)] - \mathbf{u}_6[\mathbf{v}_5\cdot(\mathbf{P}_4 - \mathbf{P}_1)].\end{aligned} \tag{10}$$

The introduction of relationships (7) and (9) into relationship (5) yields

$$\det(\mathbf{J}) = \mathbf{n}_{12}\cdot\{\mathbf{h}\times\mathbf{n}_{34} + \mathbf{q}\,[(\mathbf{P}_3 - \mathbf{P}_1)\cdot\mathbf{n}_{34}]\}, \tag{11}$$

which can be transformed in matrix form as follows (here, a vector symbol, say $\mathbf{a}$, denotes the associated 3×1 matrix, $\widetilde{\mathbf{a}}$ denotes the skew-symmetric matrix associated to $\mathbf{a}$, and $\mathbf{a}^{\mathrm{T}}$ denotes the transpose of $\mathbf{a}$)

$$\det(\mathbf{J}) = \mathbf{n}_{12}^{\mathrm{T}}\,\mathbf{A}\,\mathbf{n}_{34} \tag{12}$$

where $\mathbf{A}$ is the 3×3 matrix

$$\mathbf{A} = \widetilde{\mathbf{h}} + [\mathbf{q}(\mathbf{P}_3 - \mathbf{P}_1)^{\mathrm{T}}]. \tag{13}$$

Matrix $\mathbf{A}$ depends only on the platform geometry and on the limb vectors $\mathbf{u}_5$ and $\mathbf{u}_6$. The introduction either of relationship (11) or of relationship (12) into Eq. (3) yields the sought-after new expression of the singularity-locus equation.

4 Discussion

The platform geometry of the studied 6–4 architecture is characterized by:

(i) $(\mathbf{P}_2 - \mathbf{P}_1) = \mathbf{0}$;
(ii) $(\mathbf{P}_3 - \mathbf{P}_1) \equiv (\mathbf{P}_4 - \mathbf{P}_1)$;
(iii) $(\mathbf{P}_3 - \mathbf{P}_1) \neq \mathbf{0}$;
(iv) $(\mathbf{P}_5 - \mathbf{P}_1) \neq \mathbf{0}$ and $(\mathbf{P}_5 - \mathbf{P}_1) \neq (\mathbf{P}_3 - \mathbf{P}_1)$;
(v) $(\mathbf{P}_6 - \mathbf{P}_1) \neq \mathbf{0}$ and $(\mathbf{P}_6 - \mathbf{P}_1) \neq (\mathbf{P}_3 - \mathbf{P}_1)$;
(vi) $(\mathbf{P}_5 - \mathbf{P}_1) \neq (\mathbf{P}_6 - \mathbf{P}_1)$.

Conditions (i) and (ii) have been already exploited during the deduction of expressions (11) and (12). All the other conditions must be considered in this discussion. In particular, condition (iii) implies that the coincidence of P_3 and P_1 must not be considered among the ones that make matrix $\mathbf{A}$ become a null matrix. Condition (iv) (condition (v)) implies that vector $\mathbf{v}_5$ ($\mathbf{v}_6$) becomes a null vector if and only if $\mathbf{u}_5$ ($\mathbf{u}_6$) is a null vector, which implies that the leg length of the 5th (6th) limb vanishes. Eventually, condition (*vi*) practically implies that $\mathbf{u}_5 = \mathbf{u}_6$ is never sufficient to make $\mathbf{h}$ and $\mathbf{q}$ simultaneously null vectors (i.e., to make matrix $\mathbf{A}$ a null matrix).

The geometric conditions that make expression (12) equal to zero (i.e., that make the 6–4 architecture singular) can be exhaustively classified as follows: (a) the ones

that make either $\mathbf{n}_{12}$ ($\mathbf{n}_{34}$) a null vector or matrix $\mathbf{A}$ a null matrix, and (b) the ones that make expression (12) zero even though $\mathbf{n}_{12}$, $\mathbf{n}_{34}$, and $\mathbf{A}$ are not null.

Regarding the singularity conditions of type (a), vector $\mathbf{n}_{12}$ ($\mathbf{n}_{34}$), according to definition (6a) (definition (8b)), becomes a null vector: (a.1) if either of $\mathbf{u}_1$ ($\mathbf{u}_3$) and $\mathbf{u}_2$ ($\mathbf{u}_4$) is a null vector (i.e., at least one limb length of limbs 1 and 2 (3 and 4) vanishes), or (a.2) if the axes of limbs 1 and 2 (3 and 4) are parallel (i.e., with reference to Fig. 2b, the triangle $B_1B_2P_1$ ($B_3B_4P_3$) degenerates into a segment). Matrix $\mathbf{A}$, over the conditions excluded by the platform geometry, becomes a null matrix (see definitions (8a), (10), and (13)): (a.3) if either of $\mathbf{u}_5$ and $\mathbf{u}_6$ is a null vector (i.e., at least one limb length of limbs 5 and 6 vanishes), or (a.4) if $(\mathbf{P}_3 - \mathbf{P}_1)$ is perpendicular both to $\mathbf{v}_5$ and to $\mathbf{v}_6$ (i.e., is parallel to $(\mathbf{v}_5 \times \mathbf{v}_6)$).

Singularities of type (b) can be identified by analyzing the eigenvalues and the eigenvectors of matrix $\mathbf{A}$. This matrix is so simple that its eigenvalues and eigenvectors can be analytically determined through an algebraic manipulator. The eigenvalues of $\mathbf{A}$ are

$$\lambda_1 = 0,$$
$$\lambda_{2,3} = \frac{1}{2}\left\{(\mathbf{P}_3 - \mathbf{P}_1)\cdot\mathbf{q} \mp \sqrt{[(\mathbf{P}_3 - \mathbf{P}_1)\cdot\mathbf{q}]^2 - 4[(\mathbf{v}_5 \times \mathbf{v}_6) - \mathbf{q}]\cdot[(\mathbf{P}_3 - \mathbf{P}_1)\times\mathbf{h}]}\right\}. \tag{14}$$

The right (left) eigenvector, $\mathbf{e}_1$ ($\mathbf{g}_1$), associated to λ_1 are

$$\mathbf{e}_1 = \mathbf{h} \tag{15a}$$
$$\mathbf{g}_1 = [\mathbf{q} - (\mathbf{v}_5 \times \mathbf{v}_6)] \times (\mathbf{P}_3 - \mathbf{P}_1), \tag{15b}$$

whereas the right (left) eigenvectors, $\mathbf{e}_2$ and $\mathbf{e}_3$ ($\mathbf{g}_2$ and $\mathbf{g}_3$), associated to λ_2 and λ_3, respectively, have more cumbersome analytic expressions that are not reported here for the sake of brevity.

Since λ_1 is zero, expression (12) is equal to zero: (b.1) if $\mathbf{n}_{34}$ is parallel to $\mathbf{e}_1$, which is a special case of coplanarity of $(\mathbf{P}_3 - \mathbf{P}_1)$, $\mathbf{u}_3$, and $\mathbf{u}_4$, or (b.2) if $\mathbf{n}_{12}$ is parallel to $\mathbf{g}_1$, which is a special case of coplanarity of $(\mathbf{P}_3 - \mathbf{P}_1)$, $\mathbf{u}_1$, and $\mathbf{u}_2$.

Moreover, since the two sets $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ and $\{\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\}$ constitute a bi-orthogonal system, the two vectors $\mathbf{n}_{12}$ and $\mathbf{n}_{34}$ can always be put in the form

$$\mathbf{n}_{12} = \sum_{i=1,3} c_i\, \mathbf{g}_i, \qquad \mathbf{n}_{34} = \sum_{i=1,3} d_i\, \mathbf{e}_i, \tag{16}$$

where the scalar coefficients c_i and d_i can be easily computed through simple formulas of linear algebra.

The introduction of (16) into relationship (12), yields

$$\det(\mathbf{J}) = \lambda_2 c_2 d_2 + \lambda_3 c_3 d_3. \tag{17}$$

Over the previously discussed zeroing conditions, expression (17) is equal to zero: (b.3) if $\mathbf{n}_{12}$ has no component along $\mathbf{g}_2$ and $\mathbf{n}_{34}$ has no component along $\mathbf{e}_3$

Table 1 Singularity conditions.

Type (a) conditions	Type (b) conditions
(a.1), (a.3): $\exists i \in \{1,\ldots,6\} \vert \mathbf{u}_i = 0$	(b.1), (b.2): $(\mathbf{n}_{34} \Vert \mathbf{e}_1)$ or $(\mathbf{n}_{12} \Vert \mathbf{g}_1)$
(a.2): $(\mathbf{u}_1 \Vert \mathbf{u}_2)$ or $(\mathbf{u}_3 \Vert \mathbf{u}_4)$	(b.3): $(\mathbf{n}_{34} \perp \mathbf{g}_3)$ & $(\mathbf{n}_{12} \perp \mathbf{e}_2)$
(a.4): $(\mathbf{P}_3 - \mathbf{P}_1) \Vert (\mathbf{v}_5 \times \mathbf{v}_6)$	(b.4): $(\mathbf{n}_{34} \perp \mathbf{g}_2)$ & $(\mathbf{n}_{12} \perp \mathbf{e}_3)$
	(b.5): $\mathbf{n}_{34}$ and $\mathbf{n}_{12}$ satisfy (18)

(i.e., $c_2 = d_3 = 0 \Leftrightarrow (\mathbf{n}_{34} \perp \mathbf{g}_3)$ & $(\mathbf{n}_{12} \perp \mathbf{e}_2)$), or (b.4) if $\mathbf{n}_{12}$ has no component along $\mathbf{g}_3$ and $\mathbf{n}_{34}$ has no component along $\mathbf{e}_2$ (i.e., $c_3 = d_2 = 0 \Leftrightarrow (\mathbf{n}_{34} \perp \mathbf{g}_2)$ & $(\mathbf{n}_{12} \perp \mathbf{e}_3)$), or, eventually, (b.5) if the following relationship among components is satisfied

$$\frac{\lambda_2 c_2}{\lambda_3 c_3} = -\frac{d_3}{d_2} \,. \tag{18}$$

All the singularity conditions are summarized in Table 1.

5 Conclusions

A new analytic expression for the singularity locus of 6–4 fully-parallel manipulators with two double and two single spherical pairs has been presented. It has been deduced from a singularity-locus expression previously proposed by the author for the general Gough–Stewart platform.

The proposed expression contains a particular 3×3 matrix. The role that this matrix plays in the geometric identification of the singularity conditions has been discussed till to give an exhaustive list of these conditions.

The proposed expression allows the separation of the geometric constant parameters from the configuration dependent ones, and it is specially useful for designing manipulators referable to this architecture.

Appendix A

Definitions (1) and the coincidence of P_3 and P_4 (see Fig. 2b) make it possible to write

$$\begin{aligned} \mathbf{u}_3 \mathrm{v}_{456} - \mathbf{u}_4 \mathrm{v}_{356} = \mathbf{u}_3 \{ [(\mathbf{P}_3 - \mathbf{P}_1) \times \mathbf{u}_4] \cdot (\mathbf{v}_5 \times \mathbf{v}_6) \} \\ - \mathbf{u}_4 \{ [(\mathbf{P}_3 - \mathbf{P}_1) \times \mathbf{u}_3] \cdot (\mathbf{v}_5 \times \mathbf{v}_6) \}, \end{aligned} \tag{A.1}$$

which, by using the identity $\mathbf{a} \times \mathbf{b} \cdot \mathbf{c} \equiv \mathbf{c} \times \mathbf{a} \cdot \mathbf{b}$, becomes

$$\begin{aligned}\mathbf{u}_3 v_{456} - \mathbf{u}_4 v_{356} = \mathbf{u}_3\{[(\mathbf{v}_5 \times \mathbf{v}_6) \times (\mathbf{P}_3 - \mathbf{P}_1)] \cdot \mathbf{u}_4\} \\ - \mathbf{u}_4\{[(\mathbf{v}_5 \times \mathbf{v}_6) \times (\mathbf{P}_3 - \mathbf{P}_1)] \cdot \mathbf{u}_3\}.\end{aligned} \tag{A.2}$$

Eventually, the identity $\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) \equiv [\mathbf{b}(\mathbf{a} \cdot \mathbf{c}) - \mathbf{c}(\mathbf{a} \cdot \mathbf{b})]$ and definitions (8) allow the transformation of formula (A.2) into relationship (7).

Appendix B

Definitions (1) and the coincidence of P_3 and P_4 (see Figure 2(b)) make it possible to write

$$\mathbf{u}_5 v_{346} - \mathbf{u}_6 v_{345} = \mathbf{u}_5[\mathbf{v}_6 \cdot (\mathbf{v}_3 \times \mathbf{v}_4)] - \mathbf{u}_6[\mathbf{v}_5 \cdot (\mathbf{v}_3 \times \mathbf{v}_4)], \tag{B.1}$$

where (the identities $\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) \equiv [\mathbf{b}(\mathbf{a} \cdot \mathbf{c}) - \mathbf{c}(\mathbf{a} \cdot \mathbf{b})]$ and $\mathbf{a} \times \mathbf{b} \cdot \mathbf{c} \equiv \mathbf{a} \cdot \mathbf{b} \times \mathbf{c}$ have been used to obtain the second and, then, the third expressions, respectively)

$$\begin{aligned}\mathbf{v}_3 \times \mathbf{v}_4 &= [(\mathbf{P}_3 - \mathbf{P}_1) \times \mathbf{u}_3] \times [(\mathbf{P}_3 - \mathbf{P}_1) \times \mathbf{u}_4] \\ &= (\mathbf{P}_3 - \mathbf{P}_1)\{[(\mathbf{P}_3 - \mathbf{P}_1) \times \mathbf{u}_3] \cdot \mathbf{u}_4\} \\ &= (\mathbf{P}_3 - \mathbf{P}_1)[(\mathbf{P}_3 - \mathbf{P}_1) \cdot (\mathbf{u}_3 \times \mathbf{u}_4)],\end{aligned} \tag{B.2}$$

The introduction of the third expression (B.2) into the right-hand side of relationship (B.1) yields relationship (9) after the factor $(\mathbf{P}_3 - \mathbf{P}_1) \cdot (\mathbf{u}_3 \times \mathbf{u}_4)$ has been collected.

Acknowledgements This work was supported by the contribution of Regione Emilia Romagna (District Councillorship for Productive Assets, Economic Development, Telematic Plan), PRRIITT misura 3.4 azione A, to InterMech (Division Acoustics and Vibrations – LAV), and by UNIFE funds.

References

1. Innocenti, C. and Parenti-Castelli, V.: Exhaustive enumeration of fully-parallel kinematic chains. In: *Proc. of the 1994 ASME Int. Winter Annual Meeting*, Chicago, USA, DSC-Vol. 55-2, pp. 1135–1141 (1994).
2. St-Onge, B.M. and Gosselin, C.M.: Singularity analysis and representation of the general Gough–Stewart platform. *Int. J. Robotics Research* **19**(3), 271–288 (2000).
3. Innocenti, C. and Parenti-Castelli, V.: Direct kinematics of the 6–4 fully parallel manipulator with position and orientation uncoupled. In: *Proc. of the European Robotics and Intelligent Systems Conference*, Corfù, Greece, pp. 3–10 (1991).
4. Wohlhart, K.: Displacement analysis of the general spherical Stewart platform. *Mechanism and Machine Theory* **29**(4), 581–589 (1994).
5. Innocenti, C.: Direct kinematics in analytical form of the 6–4 fully-parallel mechanism. *ASME J. of Mechanical Design* **117**, 89–95 (1995).
6. Liao, Q.Z., Senevirantne, L.D., and Earles, S.W.E.: Forward positional analysis for the general 4–6 in-parallel platform. *Proc. Inst. Mech. Eng. – Part C: J. Mech. Eng. Sci.* **209**, 55–67 (1995).

7. Di Gregorio, R.: Analytic formulation of the 6–3 fully-parallel manipulator's singularity determination. *Robotica* **19**, 663–667 (2001).
8. Di Gregorio, R.: Singularity-locus expression of a class of parallel mechanisms. *Robotica* **20**, 323–328 (2002).
9. Li, H., Gosselin, C.M., Richard, M.J., and St-Onge, B.M.: Analytic form of the six-dimensional singularity locus of the general Gough–Stewart platform. *ASME J. of Mechanical Design* **128**(1), 279–287 (2006).
10. Ben-Horin, P. and Shoham, M.: Singularity analysis of a class of parallel robots based on Grassmann-Cayley algebra. *Mechanism and Machine Theory* **41**(8), 958–970 (2006).
11. Eberly, D.: The Laplace expansion theorem: computing the determinants and inverses of matrices. Geometric Tools, LLC (2008). http://www.geometrictools.com/Documentation/LaplaceExpansionTheorem.pdf

The Pre-Stereographic Model of the General Three-System of Screws

D. Zlatanov

PMAR Laboratory, DIMEC, University of Genoa, Italy;
e-mail: zlatanov@dimec.unige.it

Abstract. The general three-system is represented by a sphere through the origin. The highest- and lowest-pitch screws meet it at two antipodes, defining the *pitch axis*. A line from the origin, parallel to each screw, intersects the sphere at an image point. Its projection on the pitch axis measures the screw's pitch; a translation equal to the point's velocity under unit clockwise rotation about the pitch axis, positions it on the screw's axis.

Key words: Screw systems, visualization, projective spaces, unilateral wrenches.

1 Introduction

In many robotic mechanisms, a rigid body is constrained to move with fewer than six degrees of freedom or is subject to wrenches that span a less than six-space. Then, the possible instantaneous motions of the body, or the systems of forces acting on it, are described by a subspace of the six-dimensional vector space of twists, $se(3)$, or wrenches, $se(3)^*$. Such linear subspaces, or their underlying projective spaces, are referred to as *screw systems* [1,2,5,6].

It is useful not only to know the exact type of the system present in each case, but to have an understanding and a visual reference of the geometric pattern of the twists or wrenches it contains. For more complex systems, it is helpful to have a geometric model of the system, i.e., a representation with a common geometric set, identifying each screw with an element, enabling the visualization of a global image of the system.

The perfect example is Ball's representation of the general two-system with a circle [1,5]. Unfortunately, this tool is not widely known, in part because it has been unclear how to generalize it for systems of higher dimension.

In [8], which studied convex cones within various systems, some special three-systems, with all screw parallel to a plane, were modeled by a pitch-to-elevation coordinate plane, in which circles represent Ball diagrams of two-subsystems. The present paper presents an analogous model for the more general three-systems, where screws can have arbitrary directions.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_48,

2 Screws and Projective Spaces

2.1 Physical and Geometrical Screws

With respect to a given reference frame, a twist (or a wrench) is given by a pair of non-commensurate physical vectors, $(\boldsymbol{\omega}, \mathbf{v}) \in se(3)$, the body's angular velocity and the linear velocity at the origin (or $(\mathbf{f}, \mathbf{m}) \in se(3)^*$, the resultant force and moment with respect to the origin).

A physical element of either $se(3)$ or $se(3)^*$ is associated with a *screw* about which the body twists or the wrench is applied. A screw is a line in space, $\ell(\boldsymbol{\xi})$ (the screw axis of $\boldsymbol{\xi}$), with an associated metric quantity, $h(\boldsymbol{\xi})$ (the pitch). Infinite-pitch screws are pure directions (free vectors).

The pitch, h, and the axis point, $\mathbf{d}$, closest to the origin, of a finite-pitch screw underlying a twist $\boldsymbol{\xi} = (\boldsymbol{\omega}, \mathbf{v})$ are given by

$$h = \frac{\boldsymbol{\omega} \cdot \mathbf{v}}{\boldsymbol{\omega} \cdot \boldsymbol{\omega}}, \quad \mathbf{d} = \frac{\boldsymbol{\omega} \times \mathbf{v}}{\boldsymbol{\omega} \cdot \boldsymbol{\omega}} \tag{1}$$

Conversely, $\boldsymbol{\xi} = (\boldsymbol{\omega}, \mathbf{v}) = (\boldsymbol{\omega}, h\,\boldsymbol{\omega} + \mathbf{r} \times \boldsymbol{\omega})$, for any $\mathbf{r} \in \ell(\boldsymbol{\xi})$. An infinite-pitch screw corresponds to a translation $(0, \mathbf{v})$ (or a force couple, $(0, \mathbf{m})$).

The geometric screw becomes an element of a physical linear space when multiplied by a dimensioned scalar, such as the twist amplitude, ω, or the wrench intensity, f. The *directed* geometric screw is a *normalized* twist, i.e., either it has a unit angular-velocity component, $|\boldsymbol{\omega}| = 1$, or is a translation with unit velocity, $\boldsymbol{\tau} = (0, \mathbf{v})$, $|\mathbf{v}| = 1$. Equivalently, each screw is identified with a class, $[\boldsymbol{\xi}]$, of twists obtained from each other by (real-number) scalar multiplication, i.e., it is an element of the five-dimensional *real projective space*, $[\boldsymbol{\xi}] \in P(se(3))$, generated by $se(3)$.

2.2 Models of Projective Space

Real projective n-space, $P(\mathbb{R}^{n+1}) = \mathbb{RP}^n$, is defined by imposing the equivalence relation $\boldsymbol{x} \sim \lambda \boldsymbol{x}$, $\lambda \neq 0$, on $\mathbb{R}^{n+1} - \{0\}$, identifying vectors that are scalar multiples. An element of $\mathbb{RP}^n$ is a class $[\boldsymbol{x}]$ of equivalent $(n+1)$-vectors and can be represented by interpreting the coordinate array of any one of them, $\boldsymbol{x} = (x_1, \ldots, x_{n+1})$, as homogenous coordinates, $[\boldsymbol{x}] = (x_1 : \ldots : x_{n+1})$. This representation is sometimes called the *analytic model* of $\mathbb{RP}^n$ [7].

The elements of $\mathbb{RP}^n$ can be thought of as lines through the origin in $\mathbb{R}^{n+1}$. (Lines in $\mathbb{RP}^n$ become 2-planes through the origin in $\mathbb{R}^{n+1}$, etc.) This is called the *homogeneous model* (or sometimes the vector-space model).

It is often desirable to visualize the elements of projective space as points of an n-dimensional manifold. This can be done via geometric models obtained by normalization, i.e., by selecting a representative point from each line.

The familiar representation of $\mathbb{RP}^n$ by $\mathbb{R}^n$ augmented with the hyperplane at infinity is called the *straight model* by Stolfi [7]. If we intersect each line through the origin with the unit n-sphere in $\mathbb{R}^{n+1}$, S^n, we can represent each element of $\mathbb{RP}^n$ as a pair of antipodal points. This is the *spherical model*.

More generally a geometric model is obtained by projection on a quadric hypersurface from a point either *away* from the quadric (as above), or *on* it. Instead of a sphere centered at the origin, we can use a sphere, $S^n(\boldsymbol{s})$, with antipodal points O and some $\boldsymbol{s} \in \mathbb{R}^{n+1}$. For a line through O not tangent to $S^n(\boldsymbol{s})$, we take the second intersection point as the image. The tangents at O comprise a copy of $\mathbb{PR}^{n-1}$:

$$\begin{aligned} \mathbb{RP}^n &\cong \left(S^n(\boldsymbol{s}) - \{O\}\right) \sqcup \mathbb{RP}^{n-1} \\ [\boldsymbol{x}] &\mapsto \begin{cases} (\boldsymbol{x}\cdot\boldsymbol{s})\,\dfrac{\boldsymbol{x}}{\|\boldsymbol{x}\|} & \text{if} \quad \boldsymbol{x}\cdot\boldsymbol{s} \neq 0 \\ [\boldsymbol{x}] & \text{if} \quad \boldsymbol{x}\cdot\boldsymbol{s} = 0 \end{cases} \end{aligned} \tag{2}$$

The image is the sphere with a point removed and replaced by a copy of $\mathbb{RP}^{n-1}$ in the tangent hyperplane. (This procedure is known as blowing up the point.) This representation maps to the straight model via stereographic projection, hence we call it the *pre-stereographic model*. It can be shown that the Ball circle diagram is such an image of the general two-system. In this paper we construct a pre-stereographic model of the general three system.

3 A Model of the General Three-System

3.1 The Pre-Stereographic Sphere

A *general* three-system, $\mathcal{P}^2 = P(\mathcal{A}^3)$, is one generated by screws with different finite pitches and mutually perpendicular concurrent axes. The coordinate axes are chosen along the principal screws, $\boldsymbol{\xi}_x = (\mathbf{i}, h_x\mathbf{i})$, $\boldsymbol{\xi}_y = (\mathbf{j}, h_y\mathbf{j})$, $\boldsymbol{\xi}_z = (\mathbf{k}, h_y\mathbf{k})$, with $h_x > h_y > h_z$. For any $\boldsymbol{\xi} \in \mathcal{A}^3$,

$$\boldsymbol{\xi} = \omega_x\boldsymbol{\xi}_x + \omega_y\boldsymbol{\xi}_y + \omega_z\boldsymbol{\xi}_z = (\omega_x\mathbf{i} + \omega_y\mathbf{j} + \omega_z\mathbf{k},\, h_x\omega_x\mathbf{i} + h_y\omega_y\mathbf{j} + h_z\omega_z\mathbf{k}). \tag{3}$$

We will assume throughout $\omega^2 = \omega_x^2 + \omega_y^2 + \omega_z^2 = 1$. The pitch and the location of $\boldsymbol{\xi}$ are obtained from (1) and (3):

$$p = H_x\omega_x^2 - H_z\omega_z^2, \quad \mathbf{d} = -H_z\omega_y\omega_z\mathbf{i} + H\omega_z\omega_x\mathbf{j} - H_x\omega_x\omega_y\mathbf{k}\,. \tag{4}$$

where relative pitch values are used: $p = h - h_y$, $H_x = h_x - h_y$, $H_z = h_y - h_z$. We denote by $H = H_x + H_z = h_x - h_z$ the largest relative pitch.

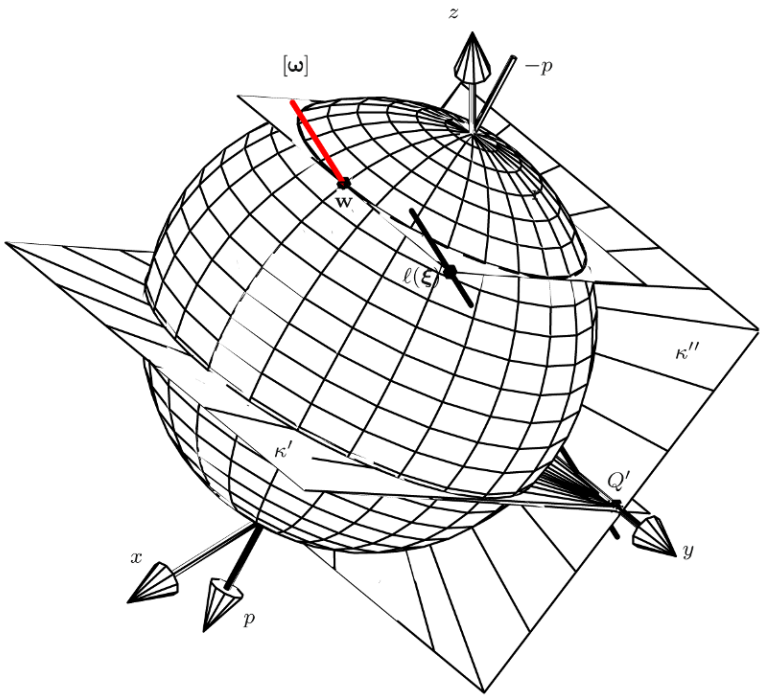


Fig. 1 The sphere, $\mathcal{S}$, with the two planes: κ', normal to the pitch axis; and κ'', tangent to $\mathcal{S}$ at O. The pencils, $\mathcal{B}'$ and $\mathcal{B}''$, of h_y-screws are visible in the planes.

We select $\mathbf{s} = s_x\mathbf{i} + s_z\mathbf{k}$, $s_x = \sqrt{HH_x}$, $s_z = \sqrt{HH_z}$, so $|\mathbf{s}| = H$. We use the sphere, $S^2(\mathbf{s})$, with antipodal points O and $\mathbf{s}$, Fig. 1. First we translate all screws of $\mathcal{P}^2$ to the origin, then we intersect their axes with $S^2(\mathbf{s})$:

$$\begin{array}{ccccc} \mathcal{P}^2 & \longrightarrow & \mathbb{RP}^2 & \longrightarrow & \left\{S^2(\mathbf{s}) - \{O\}\right\} \sqcup \mathbb{RP}^1 \\ [\boldsymbol{\xi}] & \longmapsto & [\boldsymbol{\omega}] & \longmapsto & \begin{cases} (\mathbf{s}\cdot\boldsymbol{\omega})\boldsymbol{\omega} & \text{if} \quad \boldsymbol{\omega}\cdot\mathbf{s} \neq 0 \\ [\boldsymbol{\omega}] & \text{if} \quad \boldsymbol{\omega}\cdot\mathbf{s} = 0 \end{cases} \end{array} \tag{5}$$

where $(\boldsymbol{\omega}, \mathbf{v})$ is a normalized twist in $[\boldsymbol{\xi}]$. As in (2), the image is $S^2(\mathbf{s})$ with the origin blown up and replaced by a copy of $\mathbb{RP}^1$ in the tangent plane at O, κ''. We denote the image set $\mathcal{S} = S^2(\mathbf{s})$, and the image point $\mathbf{w}(\boldsymbol{\xi}) = \mathbf{w}_\mathbf{s}(\boldsymbol{\xi})$.

In the general three-system the h_y-screws form two planar pencils in two planes intersecting in $\boldsymbol{\xi}_y$ [5]. The sphere is constructed so that it is tangent to one of these pencils, $\mathcal{B}''$ at $Q'' \in \kappa''$, Fig. 1. The second pencil, $\mathcal{B}'$, is in the plane $\kappa' \perp \mathbf{e}_p$ at Q'. (The coordinates of Q' and Q'' are $(0, \pm H/2, 0)$.)

All $[\boldsymbol{\xi}] \notin \mathcal{B}''$ are faithfully represented by their image points, $\mathbf{w}(\boldsymbol{\xi}) = (\mathbf{s}\cdot\boldsymbol{\omega})\boldsymbol{\omega} \in \mathcal{S}$; elements of $\mathcal{B}''$ are only translated to O, they have the same image point $\mathbf{w} = \mathbf{0}$ and so are represented by a line through O tangent to $\mathcal{S}$.

For a given set of principal axes and relative-pitch constants, there are four possible choices for $\mathcal{S}$, corresponding to the four possible choices of the coordinate directions of a right-handed frame along the given axes.

3.2 The Pitch Axis

The line from $\mathbf{w}_z = s_z\mathbf{k}$ to $\mathbf{w}_x = s_x\mathbf{i}$, is the *pitch axis* of $\mathcal{S}$.

Proposition 1 *The projection of the pre-stereographic image vector of any screw on the pitch axis is equal to its relative pitch, $p = h - h_y = \mathbf{w} \cdot \mathbf{e}_p$.*

Proof. Indeed,

$$\mathbf{w} \cdot \mathbf{e}_p = (\mathbf{s} \cdot \boldsymbol{\omega})\boldsymbol{\omega} \cdot \mathbf{e}_p = \left[(s_x\mathbf{i} + s_z\mathbf{k}) \cdot \boldsymbol{\omega}\right] \boldsymbol{\omega} \cdot \frac{1}{H}(s_x\mathbf{i} - s_z\mathbf{k})$$

$$= \frac{1}{H}(s_x\omega_x + s_z\omega_z)(s_x\omega_x - s_z\omega_z) = \frac{1}{H}\left(s_x^2\omega_x^2 - s_z^2\omega_z^2\right) = H_x\omega_x^2 - H_z\omega_z^2 = p$$

This means that under $\mathbf{w} : \mathcal{P}^2 \to \mathcal{S}$ the screws of a given pitch $h = h_y + p$, $p \neq 0$, map bijectively onto a circle $\sigma_p = \mathcal{S} \cap \pi_p$, where $\pi_p \perp \mathbf{e}_p$ is a plane at a distance p from O. In other words, the pitch isograms on $\mathcal{S}$ are the "parallels" on the sphere. The "north" and "south" "poles" are $\mathbf{w}_x$ and $\mathbf{w}_z$. (Indeed, it is well known that $\left[\boldsymbol{\xi}_x\right]$ and $\left[\boldsymbol{\xi}_z\right]$ are the only screws in the system with the extreme pitches h_x and h_z, otherwise $h_z < h < h_x$.)

3.3 The Screw Axis Location

Here we relate "longitude" on $\mathcal{S}$ and screw-axis location.

Proposition 2 *On the axis of any screw, $[\boldsymbol{\xi}]$, there is a point, $\mathbf{r}$, which is displaced from the screw's image point, $\mathbf{w}(\boldsymbol{\xi})$, by a vector, $\mathbf{r} - \mathbf{w} = \rho(\mathbf{w})\mathbf{n}(\mathbf{w})$, with magnitude the distance, $\rho(\mathbf{w})$, between the image point and the pitch axis, and direction as the oriented area of the triangle, $\triangle\mathbf{w}\mathbf{w}_x\mathbf{w}_z$, with vertices the image points of the screw and the principal screws of highest and lowest pitch.*

Proof. It is sufficient to show that $(\mathbf{r} - \mathbf{d}) \times \boldsymbol{\omega} = 0$. Using the expression $\mathbf{r} = \mathbf{w} + (\mathbf{w} - \mathbf{w}_z) \times \mathbf{e}_p = \mathbf{w} + \mathbf{w} \times \mathbf{e}_p - \sqrt{H_x H_z}\mathbf{j}$, as well as (4) for $\mathbf{d}$, it is not difficult to compute $\mathbf{r} \times \boldsymbol{\omega}$ and $\mathbf{d} \times \boldsymbol{\omega}$ and show that they are equal.

The vector $\mathbf{r} - \mathbf{w}$ can be seen as the velocity of point $\mathbf{w}$ under unit clockwise rotation about $\mathbf{e}_p$. It is along the tangent "due west" from $\mathbf{w}$, Fig. 2.

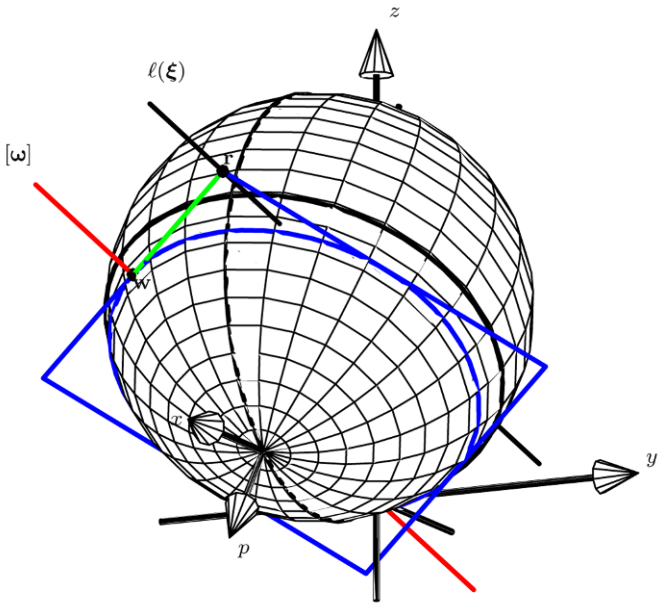


Fig. 2 The location of the screw axis for a given image point **w**. A square through **w** circumscribes its "parallel". The vertex, **r**, to the "west" of **w** is on the screw axis, $\ell(\boldsymbol{\xi})$.

4 Possible Applications of the Pre-Stereographic Model

General three-systems arise commonly in robotics. Examples include the freedom and constraint systems of 3-dof manipulators (e.g., some types of 3-*RPS* [4] and 3-*UPU*, and more generally 3-*RER* [9] parallel mechanisms), as well as the wrenches generated by varying the magnitudes of three forces applied to a body on three given lines (e.g. by a three-finger grasp of a generic smooth surface, or by three skew cables). Moreover, screw systems, and $\mathcal{P}^2$ in particular, can be used to describe certain two-dimensional sets of finite displacements, such as those of the end link of an *RR*-chain or the different ways to move a line between two given locations [3].

It is difficult to display $\mathcal{P}^2$ both accurately and clearly, as its screw axes pass through every point in space and a direct visualization is impossible. Hence the desire to model the system with a two-dimensional set. A representation using the straight [1, 2] or spherical models of $\mathbb{RP}^2$ conceals the pitch and location of the screws. In contrast, modeling with $\mathcal{S}$ provides a simple and compact visual image conveying readily accurate quantitative information. When studying a CAD model of a 3-dof parallel manipulator it would be useful to see the possible instantaneous motions of the end-effector at each simulated configuration. This can be accomplished by integrating into the CAE software a visualization algorithm able to generate a 3D plot of the pre-stereographic sphere, as in Fig. 2. (All figures have been generated by parametric 3D plotting algorithms written in Maple.)

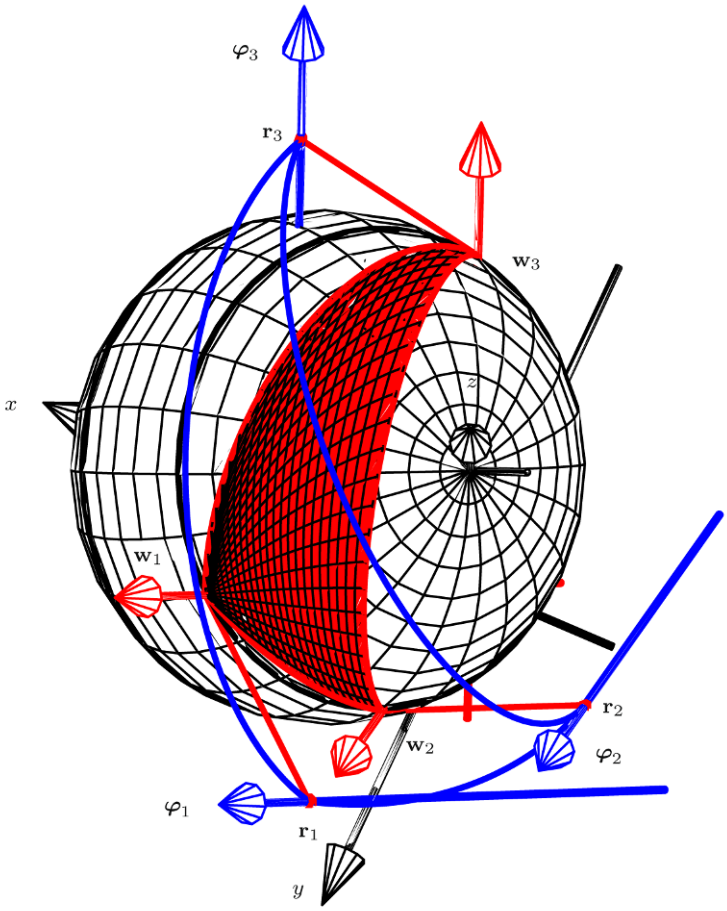


Fig. 3 The convex cone generated by three skew forces, $\boldsymbol{\varphi}_1 = (.4302418364\mathbf{i} + .4590955140\mathbf{j} + .7772536724\mathbf{k}, 7.314111219\mathbf{i} + 3.672764112\mathbf{j} - 6.218029379\mathbf{k})$, $\boldsymbol{\varphi}_2 = (.009467261665\mathbf{i} + 0.7070077476\mathbf{j} + 0.7071424300\mathbf{k}, .1609434483\mathbf{i} + 5.656061981\mathbf{j} - 5.657139440\mathbf{k})$, and $\boldsymbol{\varphi}_3 = (.2587986556\mathbf{i} - .6287677134\mathbf{j} + .7332628577\mathbf{k}, 4.399577145\mathbf{i} - 5.030141707\mathbf{j} - 5.866102862\mathbf{k})$. The image points of the screws in the cone form a triangle, $\mathbf{w}_1\mathbf{w}_2\mathbf{w}_3$, on $\mathcal{S}$. The corresponding screw-axis points form a triangle with vertices $\mathbf{r}_1 = (1.385157564, 15.93045683, 11.03886818)$, $\mathbf{r}_2 = (-7.944365468, 4.26855305, 4.041725896)$, $\mathbf{r}_3 = (-7.944365468, 4.26855305, 4.041725896)$ with elliptical sides on a spheroid.

Even more difficult to visualize than a screw system is a convex cone within it, yet such sets are ubiquitous in grasping, fixturing, and cable robotics, where forces can vary in intensity but not in direction. In the geometric model of the system, such a cone is represented by a *characteristic simplex* [8]. For $\mathcal{P}^2$ the simplex can be a spherical triangle on $\mathcal{S}$. Figure 3 illustrates the resultant wrenches generated by three unidirectional forces. These can be, the possible reaction wrenches generated by three contact forces. Thus, it is evident from the figure that these three forces

cannot keep in equilibrium an unconstrained body in a gravity field, as the resultant cannot be a pure force. Algorithms creating such 3D plots can be useful in the design and analysis of mechanical devices for the application of unidirectional wrenches.

5 Conclusions

As a basis for visualization techniques, the proposed pre-stereographic model of the three-system can be helpful for analysis and design in a variety of areas of robotics, such as grasping, fixturing, and cable-driven or parallel manipulators. The screws of the general three-system, passing through every point in space, are much more difficult to envision than the cylindroid, which now can be drawn in 3D on every laptop. Thus, a sophisticated representation, such as the one proposed, is arguably more important in the three-dimensional case than for the general two-system, elegantly represented by the Ball circle.

References

1. Ball, R.S.: *A Treatise on the Theory of Screws*. Cambridge University Press (1900).
2. Gibson, C.G., Hunt, K.H.: Geometry of screw systems. *Mechanism and Machine Theory* **25**, 1–27 (1990).
3. Huang, C., Wang, J.: The finite screw system associated with the displacement of a line. *Journal of Mechanical Design* **125**, 105–109 (2003).
4. Huang, Z., Wang, J.: Identification of principal screws of 3-DOF parallel manipulators by quadric degeneration. *Mechanisms and Machine Theory* **36**, 893–911 (2001).
5. Hunt, K.H.: *Kinematic Geometry of Mechanisms*. Clarendon Press (1978).
6. Selig, J.M.: *Geometric Fundamentals of Robotics*. Springer (2005).
7. Stolfi, J.: *Oriented Projective Geometry*. Academic Press (2005).
8. Zlatanov, D., Agrawal, S., Gosselin, C.M.: Convex cones in screw spaces. *Mechanism and Machine Theory* **40**, 710–727 (2006).
9. Zlatanov, D., Gosselin, C.M.: On the kinematic geometry of 3-RER parallel mechanisms. In: *Proceedings of the 11th World Congress in Mechanism and Machine Science*, Tianjin (2004).

Difficulty of Kinematic Synthesis of Usable Constrained Planar 6R Robots

J. Michael McCarthy and Jerome Choe

Robotics and Automation Laboratory, University of California, Irvine, Irvine, CA 92697, e-mail: {jmmccart, jichoe}@uci.edu

Abstract. In this paper we consider the difficulty of finding tasks for kinematic synthesis that result in usable constrained 6R planar serial chain. Kinematic synthesis consists of solving the constraint equations for articulated systems to determine the dimensions of the device the reaches a set of specified task positions. Our initial results show that the probability that an inventor finds a task that yields a usable constrained nR chain is approximately $P = (1/36)^{n-1}$. For a constrained 6R chain this is one out of 60 million randomly selected tasks. We consider this to be a difficult kinematic synthesis problem.

Key words: Kinematic synthesis, robot design, constrained serial chains, planar 6R robot.

1 Introduction

In this paper, we study the synthesis procedure introduced by Soh and McCarthy [1] to constrain a planar nR serial chain into a single degree-of-freedom 2n-bar linkage that has $3n-2$ revolute joints. This procedure is equivalent to the kinematic synthesis of a sequence of $n-1$ RR constraints. Soh and McCarthy used this procedure to design a constrained 6R chain that guided its end-effector though five specified task positions (Fig. 1).

An RR serial chain that is constrained by a second RR chain forms a four-bar linkage with two assemblies. If we constrain an nR chain with $n-1$ RR chains, we obtain a sequence of $n-1$ four-bar chains, with 2^{n-1} assemblies (Fig. 2). In this paper we determine experimentally the probability that all five of the task positions of a constrained 2R chain lie on one assembly, and use this to estimate the equivalent probability for a constrained 6R chain.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_49, © Springer Science+Business Media B.V. 2010

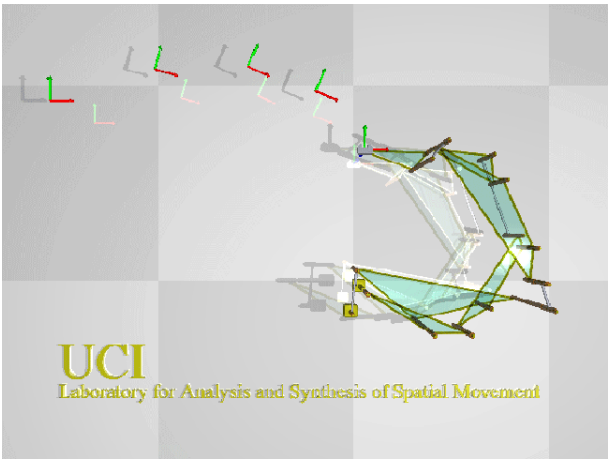


Fig. 1 This 12-bar linkage is obtained by constraining the links of planar 6R serial chain so that it guides the end-effector through the same five task positions as the unconstrained robot.

2 Literature Review

Kinematic synthesis computes the dimensions of a mechanical constraint, such as a planar RR chain, such that the end-link of the chain passes through a specified set of task positions – R refers to a revolute or hinged joint. Introduced by Burmester [3], and developed further by Hartenberg and Denavit [2], research in kinematic synthesis has extended the planar theory to spherical and spatial task positions [4–6] for a wide variety of mechanical constraints. Also see the books by Suh and Radcliffe [7], Sandor and Erdman [8], and McCarthy [9]. Recent results can be found in [10, 11].

The addition of an RR chain to a linkage introduces two modes of assembly. Failure of the task positions to lie on the same assembly is called a circuit, or branch, defect. A linkage with this defect generates trajectories that pass through a portion of the task positions in one assembly, and the remainder of the task positions in the second assembly, and is not useful.

The elimination of branch defects in linkage synthesis was addressed by Waldron [12] for kinematic synthesis of RR constraints defined by four task positions. Mirth and Chase [13] studied the problem for six-bar linkages. However, no work has considered the branching problem for RR synthesis through five task positions, and there is little if any research on branching for other planar, spherical or spatial open chains used as mechanical constraints.

Our work in the kinematic synthesis of constrained serial chains is inspired by Krovi et al. [14], who derived synthesis equations for planar nR planar serial chains in which the n joints are constrained by a cable drive. They obtained a "single degree-of-freedom coupled serial chain" used to design an assistive device. Also see [15] who developed a synthesis theory for 3R chains.

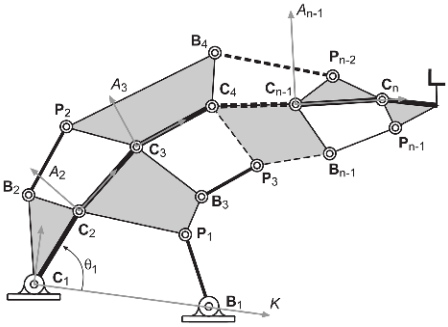


Fig. 2 This shows our conventions for the analysis of a mechanically constrained nR serial chain.

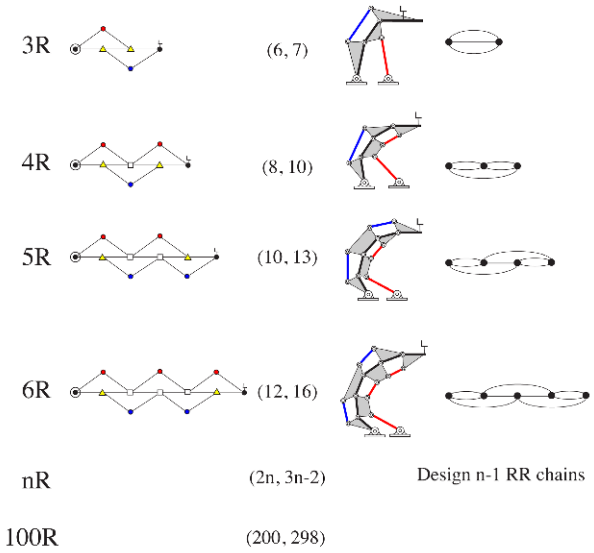


Fig. 3 This shows the kinematic structure of mechanically constrained serial chains. The linkage graph is on the left, which has each link as a node and each R joint as an edge. The contracted graph on the right shows only links with three or four edges as nodes. This shows that the structure extends to any length of nR robot.

3 Kinematics Equations of a Planar nR Chain

Let the configuration of an nR serial chain be defined by the coordinates, $\mathbf{C}_i = (x_i, y_i)$, $i = 1, \ldots, n$, that locate its n revolute joints. The distances $a_{i,i+1} = |\mathbf{C}_{i+1} - \mathbf{C}_i|$ define the lengths of each link. Attach a frame A_i to each of these links so so its origin is located at $\mathbf{C}_i$ and its x axis is directed toward $\mathbf{C}_{i+1}$. The joints $\mathbf{C}_1$ and $\mathbf{C}_n$ are the attachments to the base frame $F = A_0$ and the moving frame $M = A_n$, respectively, and we assume they are the origins of these frames. The joint angles θ_i define the relative rotation about the joints $\mathbf{C}_i$.

Introduce a world frame G and task frame H so the kinematics equations of the nR chain are given by

$$[D] = [G][Z(\theta_1)][X(a_{12})][Z(\theta_2)][X(a_{23})]\dots[X(a_{n-1,n})][Z(\theta_n)][H], \tag{1}$$

where $[Z(\theta_i)]$ and $[X(a_{i,i+1})]$ are the 3×3 homogeneous transforms that represent a rotation about the z-axis by θ_i, and a translation along the x-axis by $a_{i,i+1}$, respectively. The transformation $[G]$ defines the position of the base of the chain relative to the world frame, and $[H]$ locates the task frame relative to the end-effector frame. The matrix $[D]$ defines the coordinate transformation from the world frame G to the task frame H.

Given five task positions $[T_j], j = 1,\dots,5$ of the end-effector of this chain, we can solve the equations

$$[D] = [T_j], \quad j = 1,\dots,5, \tag{2}$$

to determine the joint parameter vectors $\theta_j = (\theta_{1,j}, \theta_{2,j}, \dots, \theta_{n,j})$. Because there are three independent constraints in this equation, we have free parameters when $n > 3$.

We assume that the free configuration variables in the nR chain have been chosen to provide the desired configuration of the nR chain in each task position. The result is that alternating pairs of links have five specified relative task positions that can be used to design an RR constraint, see Fig. 3. Soh [1] was able to solve these constraints to obtain a working constrained 6R (Fig. 1).

4 Synthesis of RR Constraints

An RR constraint has five design parameters, the coordinates of the base pivot $\mathbf{B} = (u, v)$ in the base frame, the coordinates of the moving pivot $\mathbf{P} = (x, y)$ in the moving frame, and the length R of the link that connects the pivots. Five task positions yield five quadratic constraint equations that can be solved to obtain these parameters. This process can be applied to alternating pairs of links in an nR chain to obtain a sequence of four-bar linkages (Fig. 2).

The inverse kinematics problem for an nR chain that reaches the five specified task $[T_j], j = 1,\dots,5$, yields the joint parameter vectors θ_j for each task position. This determines five positions for each link frame A_{k+1} relative to the link frame A_{k-1}. The result is $n-1$ sets of five relative positions, each of which is used to synthesize an RR chain that connects link A_{k+1} to A_{k-1}.

Let $[V_{k,j}]$, $j = 1,\dots,5$ denote the five positions of A_{k+1} relative to A_{k-1}. Then the coordinates of the moving pivot $\mathbf{P}_k$ attached to A_{k+1} and the fixed pivot $\mathbf{B}_k$ attached to A_{k-1} must satisfy the equations,

$$([V_{k,j}]\mathbf{P}_k - \mathbf{B}_k) \cdot ([V_{k,j}]\mathbf{P}_k - \mathbf{B}_k) = R_k, \quad j = 1,\dots,5. \tag{3}$$

These are the equations can be solved algebraically to determine as many as four solutions for the coordinates of $\mathbf{P}_k$ and $\mathbf{B}_k$, [8, 9]. It is important to note that the

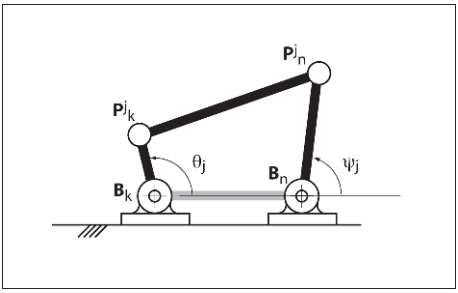


Fig. 4 The synthesis results for a four-bar linkage are used to compute the input and output angles θ_i and ψ_i.

existing RR chain $\mathbf{C}_k\mathbf{C}_{k+1}$ satisfies these design equations, which guarantees the existence of second real solution.

The RR design equations allow us to constrain an nR chain to reach five task positions. Figure 3 lists the planar linkages that this procedure allows us to design. Notice that in each case, we obtain a sequence of four-bar linkages extending from the base G to the end-effector frame H. Furthermore, while the base and end-effector frames are binary links having only two revolute joints, the links A_1 and A_{n-1} are ternary links, and the remaining links are quaternary.

5 Identification of Usable Four-bar Linkages

In this section, we show how to determine if a four-bar linkage has a circuit defect, and use this to test if the set of five task positions yield a usable four-bar linkage.

The constraint equations (3) are solved to obtain four sets of values $\sigma_k = (u,v,x,y,R)_k$, $k = 1,2,3,4$, which appear as: (i) four imaginary solutions; (ii) two imaginary and two real solutions; or (iii) four real solutions. Two real solutions are used to define the quadrilateral that forms a four-bar linkage, therefore case (ii) yields one four-bar linkage, and case (iii) yields six four-bar linkages.

If the set of task positions results in four imaginary solutions to the design equations, case (i), then the task fails to yield a usable four bar linkage.

For cases (ii) and (iii), choose two real solutions, σ_k and σ_n, and use the segment $\mathbf{B}_k\mathbf{P}_k$ as the input crank and the segment $\mathbf{B}_n\mathbf{P}_n$ as the output crank (see Fig. 4). In each of the five task positions, we evaluate the input angles θ_j and output angles ψ_j of the quadrilateral $\mathscr{Q}_{km} : (\mathbf{B}_k, \mathbf{P}_k, \mathbf{P}_n, \mathbf{B}_n)$ using the formulas,

$$\theta_j = \arctan\left(\frac{\det[\mathbf{B}_n - \mathbf{B}_k, \mathbf{P}_k^j - \mathbf{B}_k]}{(\mathbf{B}_n - \mathbf{B}_k)\cdot(\mathbf{P}_k^j - \mathbf{B}_k)}\right), \text{and}$$

$$\psi_j = \arctan\left(\frac{\det[\mathbf{B}_n - \mathbf{B}_k, \mathbf{P}_n^j - \mathbf{B}_n]}{(\mathbf{B}_n - \mathbf{B}_k)\cdot(\mathbf{P}_n^j - \mathbf{B}_n)}\right), j = 1,\ldots,5, \tag{4}$$

where $\mathbf{P}^j = [V_j]\mathbf{P}$ are the five coordinate vectors defining the moving pivots in each task position.

Position analysis of the four-bar quadrilateral $\mathscr{Q}_{km}$ yields an equation of the form [9]:

$$A(\theta)\cos\psi + B(\theta)\sin\psi + C(\theta) = 0. \tag{5}$$

This equation is solved to obtain

$$\psi^{(+,-)} = \arctan\left(\frac{B}{A}\right) \pm \arccos\left(\frac{-C}{\sqrt{A^2+B^2}}\right). \tag{6}$$

The plus and minus signs in (6) define the two modes of assembly of the quadrilateral $\mathscr{Q}_{km}$.

In order to determine if a given task, $[V_j], j = 1,\ldots,5$, yields a usable four-bar linkage, we do the following:

1. Select pairs of real solutions σ_k and σ_n and construct a quadrilateral, or quadrilaterals, $\mathscr{Q}_{km}$;
2. Use the formulas (4) to evaluate the input angles $\Theta = (\theta_1, \theta_2, \ldots, \theta_5)$, and the output angles $\Psi = (\psi_1, \psi_2, \ldots, \psi_5)$ for each $\mathscr{Q}_{km}$;
3. Let mode$(\psi_j) = 1$, if ψ_j in Ψ equals ψ_j^+ obtained from (6), similarly let mode$(\psi_j) = 0$, if $\psi_j$equals ψ_j^- in (6), and evaluate the array $\mu = (\text{mode}(\psi_1), \text{mode}(\psi_1), \ldots, \text{mode}(\psi_5))$;
4. Return the array μ.

If the array μ consists of all ones or all zeros then the four-bar linkage is usable because all the task positions lie on a single assembly mode.

6 Probability of a Usable Four-bar Linkage

We can estimate the probability that a randomly selected set of five task positions will yield a usable four-bar linkage via kinematic synthesis. To do this we assume that the imaginary and real solutions to the kinematics equations are uniformly distributed among the four combinations: (1) four complex solutions, (2) two complex solutions and two real solutions), (3) two real solution and two imaginary solutions, and (4) four real solutions. This implies that 3/4 of randomly selected task positions yield at least one real four-bar linkages.

The probability that a real four-bar linkage is usable can be estimated by considering the placement of 1 or 0 in five bins. There are 2^5 combinations, with two cases that are useful. This yields $P = 1/2^4$. Thus, the probability that a randomly selected task yields a usable linkage is $P = (3/4)(1/2^4) = 0.047$. Another way of saying this is that 1 task out of 21.3 randomly selected tasks yields a usable four-bar linkage.

Table 1 presents five numerical experiments that evaluate the probability that a random task yields usable four-bar linkages. We call these usable tasks. This table

Table 1 Results of numerical experiments to determine the distribution of randomly selected tasks that yield usable four-bar linkages.

Random Tasks	*10,000*	*50,000*	*50,000*	*100,000*	*1,000,000*
Four complex solutions	2282	22642	11594	23083	230766
% of solutions	*22.8*	*23.3*	*23.2*	*23.1*	*23.1*
Two complex, Two real	5533	27442	27537	55281	551269
% of solutions	*55.3*	*54.9*	*55.1*	*55.3*	*55.1*
Four real solutions	2185	10916	10869	221636	217965
% of solutions	*21.8*	*21.8*	*21.7*	*22.2*	*21.8*
Tasks with usable four-bars	399	2080	2178	4262	42172
Tasks per usable task	25.1	24.0	23.0	23.5	23.7

shows that our assumption of uniform distributions are off slighly, and the probability is approximately 1 out of 24 tasks.

7 Probability of Usable Constrained 2R Robots

The synthesis procedure for a constrained nR chain presented by Soh and McCarthy [1] begins with five task configurations for the entire chain. Solution of the synthesis equations yield an RR constraint for alternating pairs of links. This is a generalization of the synthesis of an RR constraint for 2R robot.

A 2R robot that has its end-effector constrained by an RR crank forms a four-bar linkage. When formulated in this way the synthesis equations always have a real solution, because the pivots of the initial 2R robot satisfy the synthesis equations.

If we assume that the 2R linkage has its base pivot at the origin and we identify a random value for its length, then we can identify a set of five random task positions for its end-effector by identifying five random values for the pair of joint angles (θ_1, θ_2). Each random task must have at least one real RR constraint that yields a four-bar linkage. We expect the probability that a random task yields a usable four-bar linkage to be $P = 1/2^4 = 0.062$. This suggests that one out of 16 random tasks will yield a usable linkage, which we find is not the case.

Table 2 presents the results from 10 numerical experiments that evaluate 10,000 random tasks for a 2R chain with a randomly assigned length for each experiment. We see that the probability of a usable task is approximately one out of 38 random tasks.

Table 2 Repeated evaluation of 10,000 random tasks to identify usable constrained 2R robots.

Test No.	*Usable tasks*	*2 Real*	*4 Real*	*Tasks per usable task*
1	227	5993	4007	36.1
2	227	6004	3996	44.0
3	264	5898	4102	37.9
4	264	6047	3953	37.9
5	280	5986	4014	35.7
6	276	6000	4000	36.2
7	263	5895	4105	38.0
8	242	5909	4091	41.3
9	256	5916	4091	39.1
10	281	5916	4084	35.6
Average	*263*	*5956*	*4043*	*38.2*
% of tasks	*2.6*	*59.6*	*40.4*	

8 Probability of Usable Constrained 6R Robots

The mechanical constraint of a 6R planar robot to achieve a 12-bar one degree-of-freedom system requires the design of five RR constraints. Each of these design problems is independent and equivalent to the synthesis of a constrained 2R robot. Thus, the probability that a randomly selected task yields a usable constrained 6R robot is P^5, where P is the probability that the task yields a usable constrained 2R robot.

We compute the probability that a random task for a 6R chain yields a usable 12-bar linkage to be one in 38^5, or approximately one in 60 million.

9 Conclusions

This paper considers the difficulty of designing a constrained 6R robot such that it moves smoothly through five task positions. We first derive the probability that a randomly selected five position task will yield a four-bar linkage that has all five positions on one assembly mode, and numerical experiments yield results close to this value, though not exactly the same. We then consider the synthesis of a constrained 2R chain and find that the probabilities are almost one-fifth of what we expect. The design of a constrained 6R is equivalent to the independent synthesis of five constrained 2R chains, which yields the probability that a task yields a usable constrained 6R robot to be approximately one in 60 million.

This is our initial work in the study of the relationship between a specified task and the usefulness of the a linkage that is defined by solving the kinematic synthesis equations. These results seem to show that finding usable task positions is difficult.

References

1. Soh, G. S. and McCarthy, J. M., Mechanically constrained nR planar serial chains. In *Advances in Robot Kinematics: Mechanisms and Motion*, J. Lenarčič and B. Roth (Eds.). Springer, Dordrecht (2006).
2. Hartenberg, R. S. and Denavit, J., *Kinematic Synthesis of Linkages.* McGraw-Hill, New York (1964).
3. Burmester, L., *Lehrbuch der Kinematik*. Verlag Von Arthur Felix, Leipzig, Germany (1886).
4. Suh, C. H. and Radcliffe, C. W., Synthesis of spherical mechanisms with the use of the displacement matrix, *ASME Journal of Engineering for Industry*, **89**(2), 215–222 (1967).
5. Suh, C. H., Design of space mechanisms for rigid body guidance, *ASME Journal of Engineering for Industry*, **90**(3), 499–506 (1967).
6. Chen, P. and Roth, B., Design equations for the finitely and infinitesimally separated position synthesis of binary links and combined link chains, *ASME Journal of Engineering for Industry*, **91**, 209–219 (1969).
7. Suh, C. H. and Radcliffe, C. W., *Kinematics and Mechanism Design*. John Wiley and Sons, New York (1978).
8. Sandor, G. N. and Erdman, A. G., *Advanced Mechanism Design: Analysis and Synthesis, Vol. 2*. Prentice-Hall, Englewood Cliffs, NJ (1984).
9. McCarthy, J. M., *Geometric Design of Linkages*. Springer-Verlag (2000).
10. McCarthy, J. M. and Su, Hai-Jun, The computation of reachable surfaces for a specified set of spatial displacements. In: E. Bayro (Ed.), *Handbook of Computational Geometry for Pattern Recognition, Computer Vision, Neurocomputing and Robotics*. Springer Verlag (2005).
11. Perez-Gracia, A. and McCarthy, J. M., Kinematic synthesis of spatial serial chains using Clifford algebra exponentials, *Proc. IMechE Vol. 220 Part C: J. Mechanical Engineering Science*, **220**(C7), 951–966 (2006).
12. Waldron, K. J., Elimination of the branch problem in graphical Burmester mechanisms synthesis for four finitely separated positions, *ASME Journal of Engineering for Industry*, **98B**, 176–182 (1976).
13. Mirth, J. A. and Chase, T. R., Circuit analysis of Watt chain six-bar mechanisms, *ASME Journal of Mechanical Design*, **115**(2), 214-222 (1993).
14. Krovi, V., Ananthasuresh, G. K., and Kumar, V., Kinematic and kinetostatic synthesis of planar coupled serial chain mechanisms, *ASME Journal of Mechanical Design*, **124**(2), 301–312 (2002).
15. Lin, C.-S. and Erdman, A. G., Dimensional synthesis of planar triads for six positions, *Mechanism and Machine Theory*, **22**, 411-419 (1987).

Stiffness Analysis of Parallel Manipulators with Preloaded Passive Joints

A. Pashkevich, A. Klimchik and D. Chablat

Institut de Recherche en Communications et en Cybernétique de Nantes, France; École des Mines de Nantes, France; e-mail: {anatol.pashkevich, alexandr.klimchik}@emn.fr, damien.chablat@irccyn.ec-nantes.fr

Abstract. The paper presents a methodology for the enhanced stiffness analysis of parallel manipulators with internal preloading in passive joints. It also takes into account influence of the external loading and allows computing both the non-linear "load-deflection" relation and the stiffness matrices for any given location of the end-platform or actuating drives. Using this methodology, it is proposed the kinetostatic control algorithm that allows to improve accuracy of the classical kinematic control and to compensate position errors caused by elastic deformations in links/joints due to the external/internal loading. The results are illustrated by an example that deals with a parallel manipulator of the Orthoglide family where the internal preloading allows to eliminate the undesired buckling phenomena and to improve the stiffness in the neighborhood of its kinematic singularities.

Key words: Modeling, parallel manipulators, external loading, internal preloading, passive joints.

1 Introduction

Parallel manipulators have become very popular in many industrial applications due to their inherent advantages of providing better accuracy, lower mass/inertia properties, and higher structural rigidity compared to their serial counterparts [1]. These features are induced by the specific kinematic structure, which eliminates the cantilever-type loading and allows to minimize deflections caused by external torques and forces. One recent development in this area, which is targeted at high-precision manipulation, is a replacing the standard passive joints by preloaded ones, which contain internal passive springs eliminating the backlash or ensure some degree of static balancing [2,3]. This modification obviously improves the manipulator performances but requires some revision of existing stiffness analysis techniques that are in the focus of this paper.

In most of previous works, the manipulator stiffness analysis was based on the linear modeling assumptions which ignore influence of the external or internal forces [4–8]. Consequently, relevant techniques are targeted at linearization of the "force-deflection" relation in the neighborhood of the non-loaded equilibrium,

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_50, © Springer Science+Business Media B.V. 2010

which is perfectly described by the stiffness matrix [9, 10]. However, in the case of non-negligible internal and/or external loading, the manipulator may demonstrate essentially non-linear behaviour, which is not exposed in the unloaded case [11]. In particular, the loading may potentially lead to multiple equilibriums, to bifurcations of the equilibriums or to static instability of certain manipulator configurations [12, 13].

This paper presents an extension of our previous results [14] devoted to the stiffness analysis of parallel manipulators by generalizing them for case of internal preloading [15] in the passive joints. It implements the virtual joint method (VJM) of Salisbary [16] and Gosselin [17] that describes the compliance of the manipulator elements by a set of localized multi-dimensional springs separated by rigid links and perfect joints. The proposed technique allows computing the loaded equilibrium, finding the full-scale "load-deflection" relation and evaluating the corresponding stiffness matrices for any given location of the end-platform or actuating drives [18]. It is also developed a kinetostatic control algorithm that allows to improve accuracy of the classical kinematic control and to compensate position errors caused by elastic deformations in links/joints due to the external/internal loading.

The remainder of this paper is organized as follows. Section 2 defines the research problem and basic assumptions. Section 3 deals with computing of the loaded static equilibrium and corresponding "load-deflection" relation. Section 4 focuses on its linearization and evaluation of the stiffness matrix. Section 5 presents the kinetostatic control algorithm. Section 6 contains an illustrative example. And finally, Section 7 summarizes the main results and contributions.

2 Manipulator Model

Let us consider a general parallel manipulator that is composed of n serial kinematic chains connecting a fixed base and a moving platform Fig. 1. It is assumed, that the chain architecture ensures kinematic control of the manipulator but may introduce some redundant constraints that improve the rigidity. Following the VJM-concept [17], let us presents the manipulator chains as sequences of pseudo-rigid links separated by rotational or translational joints of one of the following types: (i) perfect passive joints; (ii) preloaded passive joints that include auxiliary flexible elements; (iii) virtual flexible joints that describe compliance of the actuators and manipulator links; (iv) actuating joints. Using this notation the geometrical model of the chain may be written as

$$\mathbf{t} = \mathbf{g}(\boldsymbol{\rho}, \mathbf{q}, \boldsymbol{\vartheta}, \boldsymbol{\theta}), \tag{1}$$

where the vector $\mathbf{t} = (\mathbf{p},\ \boldsymbol{\varphi})^T$ includes the Cartesian position $\mathbf{p} = (x,\ y,\ z)^T$ and orientation $\boldsymbol{\varphi} = (\varphi_x,\ \varphi_y,\ \varphi_z)^T$ of the end-platform, $\boldsymbol{\rho}$ is the vector of actuated coordinates (they are constant for static analysis), the vector $\mathbf{q}$ contains coordinates of all perfect passive joints, the vector $\boldsymbol{\vartheta}$ includes coordinates of the preloaded passive

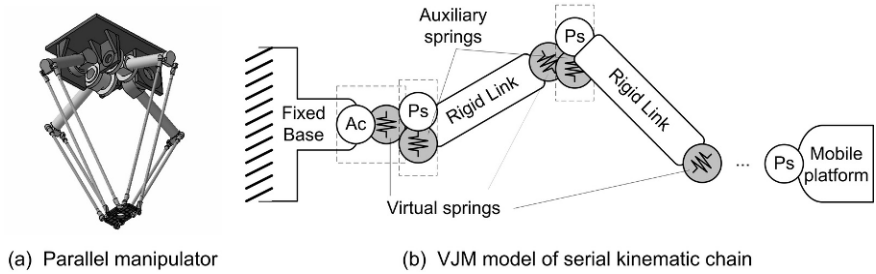


Fig. 1 Typical parallel manipulator and VJM-model of its kinematic chain. (Ac – actuator; Ps – Passive joint)

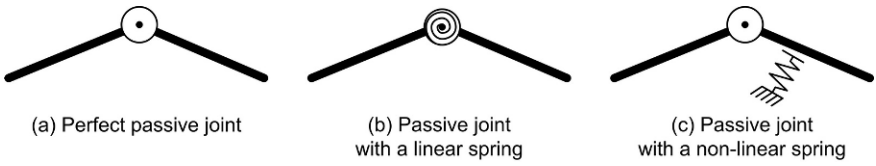


Fig. 2 Examples of auxiliary springs in preloaded passive joints.

joints, and the vector $\boldsymbol{\theta}$ collects coordinates of all virtual springs describing elasticity of the links and joints.

The above mentioned elements of the kinematic chain differ in their static characteristics. In particular, the joints (i) and (iii) are described by the standard expressions [14]

$$\boldsymbol{\tau}_q = \mathbf{0} \quad \text{and} \quad \boldsymbol{\tau}_\theta = \mathbf{K}_\theta \cdot \boldsymbol{\theta} \tag{2}$$

where $\boldsymbol{\tau}_q$ and $\boldsymbol{\tau}_\theta$ are the generalized force/torque reactions corresponding to the aggregated vectors of the passive joint coordinates $\mathbf{q}$ and virtual joint coordinates $\boldsymbol{\theta}$; $\mathbf{K}_\theta$ is the generalized stiffness matrix of all virtual springs. However, the preloaded passive joints (ii) may include both linear and non-linear auxiliary springs, some examples of which are shown in Fig. 2. In this paper, we will describe statics of the preloaded joints by a general expression

$$\tau_\vartheta = \mathbf{K}_\vartheta \cdot \mathbf{h}(\vartheta - \vartheta_0) \tag{3}$$

where $\boldsymbol{\tau}_\vartheta$ is the generalized force/torque reactions corresponding to the aggregated vectors of the preloaded joint coordinates $\boldsymbol{\vartheta}$; $\boldsymbol{\vartheta}_0$ defines the preloading value; $\mathbf{K}_\vartheta$ is the generalized stiffness matrix of preloaded joints, and the vector function $\mathbf{h}(\ldots)$ is assumed to be piecewise-linear, such that each of its scalar components $h_i(\ldots)$ can be expresses either as the difference $(\vartheta_i - \vartheta_{i0})$, or its positive or negative part $[\vartheta_i - \vartheta_{i0}]^+$, $[\vartheta_i - \vartheta_{i0}]^-$ (see Fig. 2 for details).

Using these assumptions, let us derive the stiffness model of the considered manipulator and sequentially consider the following sub-problems: (i) computing the loaded static equilibrium and obtaining the "load-deflection" relation; (ii) linearization of this relations in the neighborhood of this equilibrium and computing the

stiffness matrix; (iii) developing the kinetostatic control algorithm, which allows to compensate position errors caused by the elastic deformations and preloading.

3 Static Equilibrium

Let us obtain first the configuration of each kinematic chain ($\mathbf{q}, \boldsymbol{\theta}, \boldsymbol{\vartheta}$) and external force $\mathbf{F}$ that correspond to the static equilibrium with the end-point location $\mathbf{t}$. Obviously, it is a dual problem compared to the classical static analysis but it is more reasonable here because of strictly parallel structure of the considered manipulator (see Fig. 1). The latter allows applying the same technique to all kinematic chains (with the same end-point location) and to compute the total external loading as the sum of the partial loadings.

Taking into account the assumption on the piecewise-linear property of the function $\mathbf{h}(.)$, let us perform regrouping of the variables. In particular, for each current configuration of the chain, the coordinates of the preloaded passive joints described by the vector $\boldsymbol{\vartheta}$ may be separated into two parts $\boldsymbol{\vartheta}_\theta$ and $\boldsymbol{\vartheta}_q$, where the first one corresponds to the active state of the auxiliary springs and the second part describes non-active springs (see Figure 2 for geometrical interpretation). This allows replacing the original set of the configuration variables ($\mathbf{q}, \boldsymbol{\theta}, \boldsymbol{\vartheta}$) by a set of two vectors ($\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}}$), where $\tilde{\mathbf{q}}$ aggregates the joint coordinates ($\mathbf{q}, \boldsymbol{\vartheta}_q$) that currently are passive and the vector $\tilde{\boldsymbol{\theta}}$ collects all spring coordinates ($\boldsymbol{\theta}, \boldsymbol{\vartheta}_\theta$) (both virtual and passive).

Using these notations and applying the virtual work technique, the static equilibrium equation of the kinematic chain may be written as

$$\tilde{\mathbf{J}}_\theta^T \cdot \mathbf{F} = \tilde{\mathbf{K}}_\theta \cdot (\tilde{\boldsymbol{\theta}} - \tilde{\boldsymbol{\theta}}_0); \quad \tilde{\mathbf{J}}_q^T \cdot \mathbf{F} = \mathbf{0} \tag{4}$$

where $\mathbf{F}$ is the external force applied at the end-point of the chain, the vector $\tilde{\boldsymbol{\theta}}_0^T = [\mathbf{0}^T, \boldsymbol{\rho}_0^T]$ aggregates the spring preloadings (which is obviously zero for the virtual springs), $\tilde{\mathbf{K}}_\theta = \mathrm{diag}(\mathbf{K}_\theta, \mathbf{K}_\vartheta)$, and $\tilde{\mathbf{J}}_\theta$, $\tilde{\mathbf{J}}_q$ are the kinematic Jacobians derived from (1) by differentiating it with respect to $\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{q}}$. This system of Eq. (4) combined with the geometrical model (1), which must be rewritten in terms of the redefined variables

$$\mathbf{t} = \tilde{\mathbf{g}}(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}}) \tag{5}$$

This yields the desired joint coordinates of the static equilibrium for a separate kinematic chain with given end-point location.

Since the derived system is highly nonlinear, in general case a desired solution can be obtained only numerically. In this paper, it is proposed to use the following iterative scheme

$$\begin{bmatrix} \mathbf{F}_{i+1} \\ \tilde{\mathbf{q}}_{i+1} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{J}}_\theta(\tilde{\mathbf{q}}_i, \tilde{\boldsymbol{\theta}}_i) \cdot \tilde{\mathbf{K}}_\theta^{-1} \cdot \tilde{\mathbf{J}}_\theta^T(\tilde{\mathbf{q}}_i, \tilde{\boldsymbol{\theta}}_i) & \tilde{\mathbf{J}}_q(\tilde{\mathbf{q}}_i, \tilde{\boldsymbol{\theta}}_i) \\ \tilde{\mathbf{J}}_q^T(\tilde{\mathbf{q}}_i, \tilde{\boldsymbol{\theta}}_i) & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} \epsilon_i \\ 0 \end{bmatrix}$$
$$\tilde{\boldsymbol{\theta}}_{i+1} = \tilde{\mathbf{K}}_\theta^{-1} \cdot \tilde{\mathbf{J}}_\theta^T(\tilde{\mathbf{q}}_i, \tilde{\boldsymbol{\theta}}_i) \cdot \mathbf{F}_{i+1} + \tilde{\boldsymbol{\theta}}_0 \tag{6}$$
$$\epsilon_i = \mathbf{t} - \mathbf{g}(\tilde{\mathbf{q}}_i, \tilde{\boldsymbol{\theta}}_i) + \tilde{\mathbf{J}}_q(\tilde{\mathbf{q}}_i, \tilde{\boldsymbol{\theta}}_i) \cdot \tilde{\mathbf{q}}_i + \tilde{\mathbf{J}}_\theta(\tilde{\mathbf{q}}_i, \tilde{\boldsymbol{\theta}}_i) \cdot (\tilde{\boldsymbol{\theta}}_i + \tilde{\boldsymbol{\theta}}_0)$$

where the starting point $(\tilde{\boldsymbol{\theta}}_0, \tilde{\mathbf{q}}_0)$ is also computed iteratively, started from a nearest unloaded configuration where the joint coordinates are easily obtained from the inverse kinematic model. On the following iterations, to improve convergence, the system variables are slightly randomly disturbed. As follows from computational experiments, the proposed iterative algorithm possesses rather good convergence (3–5 iterations are usually enough).

4 Stiffness Matrix

To compute the desired stiffness matrix, let us consider the neighborhood of the equilibrium configuration and assume that the external force and the end-effector location are incremented by some small values $\delta\mathbf{F}$, $\delta\mathbf{t}$. Besides, let us assume that a new configuration also satisfies the equilibrium conditions. Hence, it is necessary to consider simultaneously two equilibriums corresponding to the manipulator state variables $(\mathbf{F}, \mathbf{q}, \boldsymbol{\theta}, \mathbf{t})$ and $(\mathbf{F} + \delta\mathbf{F}, \mathbf{q} + \delta\mathbf{q}, \boldsymbol{\theta} + \delta\boldsymbol{\theta}, \mathbf{t} + \delta\mathbf{t})$. Relevant equations of statics may be written as

$$\begin{aligned} &\tilde{\mathbf{J}}_\theta{}^T \mathbf{F} = \tilde{\mathbf{K}}_\theta(\tilde{\boldsymbol{\theta}} - \tilde{\boldsymbol{\theta}}_0); && \tilde{\mathbf{J}}_q{}^T \mathbf{F} = \mathbf{0}; \\ &(\tilde{\mathbf{J}}_{\boldsymbol{\theta}} + \delta\tilde{\mathbf{J}}_{\boldsymbol{\theta}})^T(\mathbf{F} + \delta\mathbf{F}) = \tilde{\mathbf{K}}_{\boldsymbol{\theta}}(\tilde{\boldsymbol{\theta}} - \tilde{\boldsymbol{\theta}}_0 + \delta\tilde{\boldsymbol{\theta}}); && (\tilde{\mathbf{J}}_{\mathbf{q}} + \delta\tilde{\mathbf{J}}_{\mathbf{q}})^T(\mathbf{F} + \delta\mathbf{F}) = \mathbf{0} \end{aligned} \tag{7}$$

where $\delta\tilde{\mathbf{J}}_q(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}})$ and $\delta\tilde{\mathbf{J}}_\theta(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}})$ are the differentials of the Jacobians due to changes in $(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}})$. Besides, in the neighborhood of $(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}})$, the kinematic equation (5) may be also presented in the linearized form:

$$\boldsymbol{\delta}\mathbf{t} = \tilde{\mathbf{J}}_\theta(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}}) \cdot \boldsymbol{\delta}\boldsymbol{\theta} + \tilde{\mathbf{J}}_q(\tilde{\mathbf{q}}, \boldsymbol{\theta}) \cdot \boldsymbol{\delta}\tilde{\mathbf{q}} \tag{8}$$

Hence, after neglecting the high-order small terms and expanding the differentials via the Hessians of the function $\Psi = \tilde{\mathbf{g}}(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}})^T\mathbf{F}$

$$\tilde{\mathbf{H}}_{qq}^F = \frac{\partial^2\Psi}{\partial\tilde{\mathbf{q}}^2}; \quad \tilde{\mathbf{H}}_{\theta\theta}^F = \frac{\partial^2\Psi}{\partial\tilde{\boldsymbol{\theta}}^2}; \quad \tilde{\mathbf{H}}_{q\theta}^F = (\tilde{\mathbf{H}}_{\theta q}^F)^T = \frac{\partial^2\Psi}{\partial\tilde{\mathbf{q}}\,\partial\tilde{\boldsymbol{\theta}}} \tag{9}$$

Eqs. (7) may be rewritten as

$$\begin{aligned} &\tilde{\mathbf{J}}_\theta^T(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}}) \cdot \delta\mathbf{F} + \tilde{\mathbf{H}}_{\theta q}^F(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}}) \cdot \delta\tilde{\mathbf{q}} + \tilde{\mathbf{H}}_{\theta\theta}^F(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}}) \cdot \delta\tilde{\boldsymbol{\theta}} = \tilde{\mathbf{K}}_\theta \cdot \delta\tilde{\boldsymbol{\theta}} \\ &\tilde{\mathbf{J}}_q^T(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}}) \cdot \delta\mathbf{F} + \tilde{\mathbf{H}}_{qq}^F(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}}) \cdot \delta\tilde{\mathbf{q}} + \tilde{\mathbf{H}}_{q\theta}^F(\tilde{\mathbf{q}}, \tilde{\boldsymbol{\theta}}) \cdot \delta\tilde{\boldsymbol{\theta}} = \mathbf{0} \end{aligned} \tag{10}$$

Besides, here the variable $\delta\tilde{\boldsymbol{\theta}}$ can be eliminated analytically: $\delta\tilde{\boldsymbol{\theta}} = \tilde{\mathbf{k}}_{\theta}^{F} \cdot \tilde{\mathbf{J}}_{\theta}^{T} \cdot \delta\mathbf{F} + \tilde{\mathbf{k}}_{\theta}^{F} \cdot \tilde{\mathbf{H}}_{\theta q}^{F} \cdot \delta\tilde{\mathbf{q}}$, where $\tilde{\mathbf{k}}_{\theta}^{F} = \left(\tilde{\mathbf{K}}_{\theta} - \tilde{\mathbf{H}}_{\theta\theta}^{\mathbf{F}}\right)^{-1}$. This leads to a system of matrix equations with unknowns $\delta\mathbf{F}$ and $\delta\tilde{\mathbf{q}}$

$$\begin{bmatrix} \tilde{\mathbf{J}}_{\theta} \cdot \tilde{\mathbf{k}}_{\theta}^{F} \cdot \tilde{\mathbf{J}}_{\theta}^{T} & \tilde{\mathbf{J}}_{\mathbf{q}} + \tilde{\mathbf{J}}_{\theta} \cdot \tilde{\mathbf{k}}_{\theta}^{F} \cdot \tilde{\mathbf{H}}_{\theta q}^{F} \\ \tilde{\mathbf{J}}_{q}^{T} + \tilde{\mathbf{H}}_{q\theta}^{F} \cdot \tilde{\mathbf{k}}_{\theta}^{F} \cdot \tilde{\mathbf{J}}_{\theta}^{T} & \tilde{\mathbf{H}}_{qq}^{F} + \tilde{\mathbf{H}}_{q\theta}^{F} \cdot \tilde{\mathbf{k}}_{\boldsymbol{\theta}}^{F} \cdot \tilde{\mathbf{H}}_{\theta q}^{F} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\delta}\mathbf{F} \\ \boldsymbol{\delta}\tilde{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\delta}\mathbf{t} \\ \mathbf{0} \end{bmatrix} \tag{11}$$

from which the desired Cartesian stiffness matrix of the chain $\mathbf{K}_c$ may be obtained by direct inversion of the the left-hand side and extracting from it the upper-left sub-matrix of size 6×6:

$$\begin{bmatrix} \mathbf{K}_c & * \\ * & * \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{J}}_{\theta} \cdot \tilde{\mathbf{k}}_{\theta}^{F} \cdot \tilde{\mathbf{J}}_{\theta}^{T} & \tilde{\mathbf{J}}_{q} + \tilde{\mathbf{J}}_{\theta} \cdot \tilde{\mathbf{k}}_{\theta}^{F} \cdot \tilde{\mathbf{H}}_{\theta q}^{F} \\ \tilde{\mathbf{J}}_{q}^{T} + \tilde{\mathbf{H}}_{q\theta}^{F} \cdot \tilde{\mathbf{k}}_{\theta}^{F} \cdot \tilde{\mathbf{J}}_{\theta}^{T} & \tilde{\mathbf{H}}_{qq}^{F} + \tilde{\mathbf{H}}_{q\theta}^{F} \cdot \tilde{\mathbf{k}}_{\theta}^{F} \cdot \tilde{\mathbf{H}}_{\theta q}^{F} \end{bmatrix}^{-1} \tag{12}$$

Finally, when the stiffness matrices for all kinematic chains are computed, the stiffness of the entire multi-chain manipulator can be found by simple summation $\mathbf{K}_{\Sigma} = \sum_{i=1}^{n} \mathbf{K}_{ci}$. It should be noted that, because of presence of the passive joints, the stiffness matrix of a separate serial kinematic chain is always singular, but aggregation of all the manipulator chains of a parallel manipulator produce a non-singular stiffness matrix.

5 Kinetostatic Control

In robotics, the manipulator motions are usually generated using the inverse kinematic model that allows computing the input (reference) signals for actuators $\boldsymbol{\rho}$ corresponding to the desired end-effector location $\mathbf{t}$. However, for manipulators with preloaded passive joints, the kinematic control becomes non-applicable because of changes in the end-platform location due to the internal loading. Hence, in this case, the control must be based on the inverse kinetostatic model that takes into account both the manipulator geometry and elastic properties of its links and joints [12].

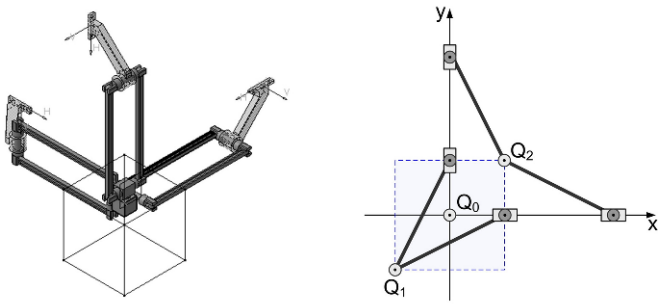
Using results from the previous sections, the desired inverse kinetostatic transformation can be performed iteratively, in the following way:

Step#1. For given target location of the end-platform $\mathbf{t}$, compute initial values of the actuated coordinates $\boldsymbol{\rho}_0$ by applying the inverse kinematic transformation.

Step#2. For current values of the actuated coordinates $\boldsymbol{\rho}_i$ and target location of the end-platform $\mathbf{t}$, find the equilibrium configuration for each kinematic chain and compute the corresponding total external loading $\mathbf{F}_{\Sigma}^{i}$ required to achieve the target location.

Step#3. If the computed external loading is less than the prescribed error, i.e. $\left|\mathbf{F}_{\Sigma}^{i}\right| < \varepsilon_F$, stop the algorithm, otherwise continue the next step

Step#4. Repeat Step#2 several times in the neighborhood of the current solution $\boldsymbol{\rho}_i$ and evaluate numerically the matrix $\mathbf{S}_{F\rho}^{i} = \partial\mathbf{F}_{\Sigma}^{i}/\partial\rho_i$ describing the sensitivity of $\mathbf{F}$ with respect to $\boldsymbol{\rho}$.



(a) 3D Orthoglide (a) 2D Orthoglide

Fig. 3 Architecture of the Orthoglide manipulator and its planar version.

Step#5. Compute new value of the actuated coordinates $\boldsymbol{\rho}_{i+1} = \boldsymbol{\rho}_i - \mathbf{S}_{F\rho}^{i\ -1} \cdot \mathbf{F}_{\Sigma}^{i}$ and repeat the algorithm starting from Step#2.

As follows from simulation results, this algorithm demonstrates good convergence and can be used both for on-line and off-line trajectory planning. It was successfully applied to the case-study presented in the following section.

6 Application Example

Let us apply the proposed techniques to the stiffness analysis of the planar manipulator of the Orthoglide family (Fig. 3). For illustration purposes, let us assume that the only source of the manipulator elasticity is concentrated in actuated drives, while the passive joints may be preloaded by (i) standard linear springs, or (ii) non-linear springs with mechanical stop-limit (see Fig. 2 for details).

For this manipulator, the kinematic model includes a single parameter L (the leg length) and the dexterous workspace was defined as the maximum square area that provides the velocity (and force) transmission factors in the range $[0.5,\ 2.0]$. Using the critical point technique developed for this type of manipulators [19], it was proved that the desired square vertices are located in the points $Q_1(-p,\ -p)$ and $Q_2(p,\ p)$, where $p = 0.45\ L$. Besides, the square centre $Q_0(0,\ 0)$ is isotropic with respect to the velocity and force transmission. The parameters of the actuating drives are also assumed identical and their linear stiffness is denoted as K_θ. The auxiliary springs incorporated in the passive joints adjacent to the actuators are described by two parameters: the angular stiffness coefficient K_ϑ and the activation angle ϑ_0 that defines the preloading activation point. During simulation, the manipulator end-point was displaced by value Δ in the direction Q_0Q_1 or Q_0Q_2, and it was computed corresponding magnitude for external force F.

The stiffness analysis results are summarized in Figs. 4, 5 and in Table 1. As follows from them, the original manipulator (without preloading in passive joints) demonstrates rather low stiffness in the neighborhood of the point Q_2, which is

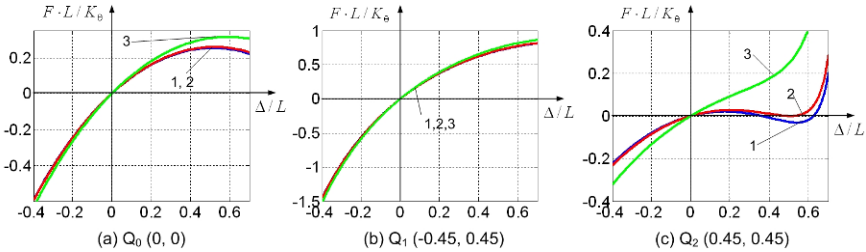


Fig. 4 Force-deflection relations $F = f(\Delta/L)$ in critical points: (1) $K_\vartheta = 0$; (2) $K_\vartheta = 0.01\ K_\theta\ L^2$; (3) $K_\vartheta = 0.1\ K_\theta\ L^2$ (case of preloading with linear springs).

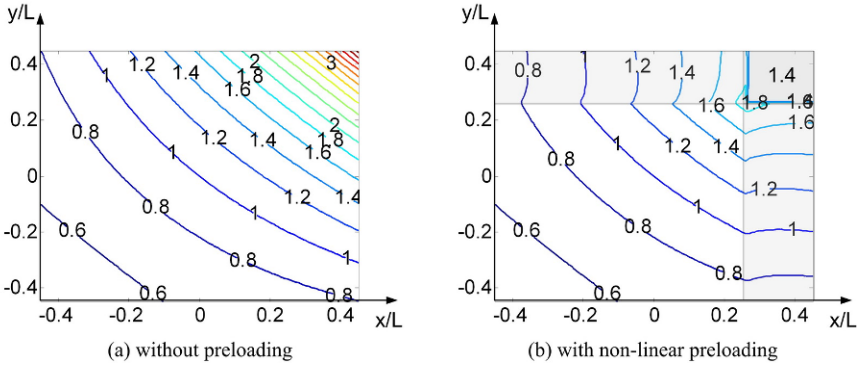


Fig. 5 Compliance maps for cases of: (a) manipulator without preloading; (b) manipulator with preloading non-linear springs with $K_\vartheta = 0.5\ K_\theta\ L^2$ and $\vartheta_0 = \pi/12$ (b).

roughly 4 times lower than in the isotropic point Q_0. In contrast, the linear stiffness in the point Q_1 is twice higher than in the point Q_0. Besides, in the point Q_2, the external loading may provoke the buckling phenomenon that is caused by a local minimum of the force-deflection relation. In this case, the distance-to-singularity is essentially lower that it is estimated from the kinematical model and the manipulator may easily loose its structural stability.

To improve the manipulator stiffness and to avoid the buckling in the neighborhood of Q_2, the passive joints were first preloaded by linear springs with activation angle $\vartheta_0 = 0$. As follows from Fig. 4, the preloading with parameter $K_\vartheta = 0.1\ K_\theta\ L^2$ allows completely eliminate buckling and improves the stiffness by the factor of 2.3. On the other hand, the stiffness in the points Q_0 and Q_1 changes non-essentially, by 10% and 5% respectively. Hence, with respect to the stiffness, such preloading has positive impact.

The only negative consequence of such preloading is related to changes of the actuator control strategy. In fact, instead of standard kinematic control, it is necessary to apply the kinetostatic control algorithm presented in Section 5. It allows compensating the position errors caused by elastic deformations due to the internal

Table 1 Manipulator stiffness for different linear preloading.

Stiffness in preloaded joints	$K_\vartheta = 0$	$0.01\ K_\theta\ L^2$	$0.05\ K_\theta\ L^2$	$0.1\ K_\theta\ L^2$
Point Q_0 (isotropic point)				
Actuating joint coordinates ρ	L	L	L	L
Manipulator stiffness $\mathbf{K}_c$	K_θ	$1.01\ K_\theta$	$1.05\ K_\theta$	$1.10\ K_\theta$
Point Q_1 (neighborhood of "bar" singularity)				
Actuating joint coordinates ρ	$0.437\ L$	$0.433\ L$	$0.419\ L$	$0.402\ L$
Manipulator stiffness $\mathbf{K}_c$	$2.276\ K_\theta$	$2.286\ K_\theta$	$2.329\ K_\theta$	$2.382\ K_\theta$
Point Q_2 (neighborhood of "flat" singularity)				
Actuating joint coordinates ρ	$1.345\ L$	$1.356\ L$	$1.399\ L$	$1.453\ L$
Manipulator stiffness $\mathbf{K}_c$	$0.24\ K_\theta$	$0.27\ K_\theta$	$0.39\ K_\theta$	$0.55\ K_\theta$
Critical force $\mathbf{F}_{cr}$	$0.020\ K_\theta\ L$	$0.027\ K_\theta\ L$	—	—

preloading and to achieve the target end-point location with modified values of the actuated joint coordinates. As follows from Table 1, corresponding adjustments of the joint coordinates may reach $0.1\ L$ and are not negligible for most of applications.

The most efficient solution that eliminates this problem is using of non-linear springs with mechanical stop-limits that are activated while approaching to Q_2. For instance, as follows from dedicated study, the preloading with the parameters $K_\vartheta = 0.5\ K_\theta\ L^2$, $\vartheta_0 = \pi/12$ provides almost the same improvements in Q_2 as the linear spring while preserving usual control strategies if the preloading is not activated. The efficiency of this approach is illustrated by the compliance maps presented in Fig. 5.

7 Conclusions

Recent advances in mechanical design of robotic manipulators lead to new parallel architectures that incorporates internal preloading in passive joints allowing to improve accuracy but leading to revision of existing stiffness analysis techniques. This paper presents new results in this area that allow simultaneously evaluate influence of internal and external loading and compute both the non-linear "load-deflection" relation and the stiffness matrices for any given location of the end-platform or actuating drives. Using this methodology, it is proposed the kinetostatic control algorithm that allows to improve accuracy of the classical kinematic control and to compensate position errors caused by elastic deformations in links/joints due to the external/internal loading. The efficiency of this technique is confirmed by an application example that deals with a parallel manipulator of the Orthoglide family where the internal preloading allows to eliminate the undesired buckling phenomena and to improve the stiffness in the neighborhood of its kinematic singularities.

In future, these results will be generalized to other types of preloading that may be generated by external gravity-compensation mechanisms and also applied to micromanipulators with flexure joints.

Acknowledgements The work presented in this paper was partially funded by the Region "Pays de la Loire" (project RoboComposite).

References

1. Merlet, J.-P.: *Parallel Robots*, 2nd ed. Springer, Dordrecht (2006).
2. Arsenault M., Gosselin C.: Kinematic and static analysis of a 3-PUPS spatial tensegrity mechanism. *Mechanism and Machine Theory* **44**, 162–179 (2009).
3. Griffis M.: Preloading compliant couplings and the applicability of superposition. *Mechanism and Machine Theory* **41**, 845–862 (2006).
4. Ceccarelli M., Carbone G.: A stiffness analysis for CaPaMan (Cassino Parallel Manipulator). *Mechanism and Machine Theory* **37(5)**, 427–439 (2002).
5. Company O., Krut S., Pierrot F.: Modelling and preliminary design issues of a 4-axis parallel machine for heavy parts handling. *Journal of Multibody Dynamics* **216**, 1–11 (2002).
6. Chen S., Kao I.: Conservative congruence transformation for joint and Cartesian stiffness matrices of robotic hands and fingers. *The International Journal of Robotics Research* **19**(9), 835–847 (2000).
7. Alici, G., Shirinzadeh, B.: The stiffness matrix in elastically articulated rigid-body systems. *Multibody System Dynamics* **18**(2), 169–184 (2007).
8. Ciblak N., Lipkin H.: Synthesis of Cartesian stiffness for robotic applications. In: *IEEE International Conference on Robotics and Automation (ICRA)*, Detroit, MI, pp. 2147–2152 (1999).
9. Kovecses J., Angeles J.: The stiffness matrix in elastically articulated rigid-body systems. *Multibody System Dynamics* **18**(2), 169–184 (2007).
10. Quennouelle C., Gosselin C.: Stiffness matrix of compliant parallel mechanisms. In: *Advances in Robot Kinematics: Analysis and Design*, J. Lenarčič and P. Wenger, Springer, Dordrecht, pp. 331–341 (2008).
11. Timoshenko S., Goodier J.N.: *Theory of Elasticity*, 3d ed. McGraw-Hill, New York (1970).
12. Su, H.J., McCarthy, J.M.: A polynomial homotopy formulation of the inverse static analysis of planar compliant mechanisms. *Journal of Mechanical Design* **128**, 776–786 (2006).
13. Carricato M., Duffy J., Parenti-Castelli V.: Catastrophe analysis of a planar system with flexural pivots. *Mechanism and Machine Theory* **37**, 693–716 (2002).
14. Pashkevich A., Chablat D., Wenger Ph.: Stiffness analysis of overconstrained parallel manipulators. *Mechanism and Machine Theory* **44**, 966–982 (2009).
15. Crane C.D., Bayat J., Vikas V., Roberts R.: Kinematic analysis of a planar tensegrity mechanism with pre-stressed springs. In: *Advances in Robot Kinematics: Analysis and Design*, J. Lenarčič and P. Wenger, Springer, Dordrecht, pp. 419–427 (2008).
16. Salisbury, J.: Active stiffness control of a manipulator in Cartesian coordinates. In: *19th IEEE Conference on Decision and Control*, pp. 87–97 (1980).
17. Gosselin, C.: Stiffness mapping for parallel manipulators. *IEEE Transactions on Robotics and Automation* **6**(3), 377–382 (1990).
18. Pashkevich, A., Klimchik, A., Chablat, D., Wenger, P.: Stiffness analysis of multichain parallel robotic systems with loading. *Journal of Automation, Mobile Robotics & Intelligent Systems* **3**(3), 75–82 (2009).
19. Chablat, D., Wenger, P.: Architecture optimization of a 3-DOF parallel mechanism for machining applications, the Orthoglide. *IEEE Transactions on Robotics and Automation* **19**(3), 403–410 (2003).

Characterization of Parallel Manipulator Available Wrench Set Facets

M. Gouttefarde and S. Krut

LIRMM, CNRS, University Montpellier 2, France;
e-mail: marc.gouttefarde@lirmm.fr, sebastien.krut@lirmm.fr

Abstract. The available wrench set is the set of wrenches that can be generated at the platform of a parallel manipulator. In a given configuration, this set is known to be a convex polytope and, recently, it has been pointed out that it is in fact a particular type of convex polytope called a zonotope. This paper deals with the case of parallel manipulators having as many or more actuators than degrees of freedom and discusses a characterization of the facets of the available wrench set.

Key words: Parallel manipulators, actuation redundancy, available wrench set.

1 Introduction

The determination of the available wrench set is a useful tool for parallel manipulator analysis and design. Notably, in the case of parallel cable-driven manipulators, the following relevant use of the available wrench set is proposed in [1]. Being given a wrench set T that the cable-driven manipulator is required to generate in order to achieve the tasks assigned to it, determine the available wrench set A and test if $T \subseteq A$, in which case T is feasible. This methodology proposed for cable-driven manipulator can obviously be applied to other types of parallel manipulator.

The available wrench set appears in numerous previous works and is known to be a convex polytope [2, 3]. In the case of serial manipulators, a set with a similar geometry is, for example, studied in [4] where it is called manipulability polytope. Recently, Bouchard et al. [5] pointed out that the available wrench set is a particular type of convex polytope called a *zonotope*. Based on specific properties of zonotopes, a simple method referred to as the hyperplane shifting method is introduced in [5]. This method provides a representation of the available wrench set as the set of solutions to a finite system of linear inequalities. By means of such a representation, it is usually straightforward to test whether or not a given required wrench set T is fully included in the available wrench set A.

The contribution of this paper is to provide a proof (Section 4) which justifies the hyperplane shifting method. This proof is mainly based on a characterization of zonotope facets which appears in [6] but without details. This lack of details explains

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_51, © Springer Science+Business Media B.V. 2010

probably why this characterization of the facets of a zonotope seems to have been overlooked in [5]. Moreover, the proof provided in the present paper leads directly to an improved version of the hyperplane shifting method (Section 5). The contribution of the paper is limited but it is nevertheless hoped to be useful in pointing out a clear characterization of the available wrench set facets, thereby complementing the work presented in [5].

2 Available Wrench Set Definition

Let us consider an n-degree-of-freedom parallel manipulator having m actuators ($m \geq n$). In a given configuration, the vector of actuator forces/torques $\boldsymbol{\tau}$ is usually mapped to the mobile platform wrenches $\mathbf{f}$ (combination of a force and a moment) according to the following linear relationship [3]

$$\mathbf{W}\boldsymbol{\tau} = \mathbf{f} \tag{1}$$

where $\mathbf{W}$ is an $n \times m$ matrix called the wrench matrix in this paper. Its ith column is denoted $\mathbf{w}_i$. Note that, in the remainder of the paper, the space of mobile platform wrenches is considered to be an affine space. Hence, it may be judged necessary to modify Eq. (1) so as to avoid physical inconsistencies, i.e., in order to avoid adding variables with different physical units in the case of parallel manipulators with mixed translational and rotational degrees of freedom.

The limited force/torque capabilities of the actuators imply that each component τ_i of vector $\boldsymbol{\tau}$ is to lie within an interval $[\tau_{i_{\min}}, \tau_{i_{\max}}]$ where $\tau_{i_{\min}}$ and $\tau_{i_{\max}}$ are the minimum and maximum values of actuator i force/torque, respectively. Note that usually $\tau_{i_{\min}} = -\tau_{i_{\max}}$ but a more general case is considered here in order to include parallel cable-driven manipulators for which $0 \leq \tau_{i_{\min}} < \tau_{i_{\max}}$ (a nonnegativity constraint due to the fact that cables can only pull and not push [1]). Let us also define the box $[\boldsymbol{\tau}]$ (hypercube) of admissible actuator forces/torques as

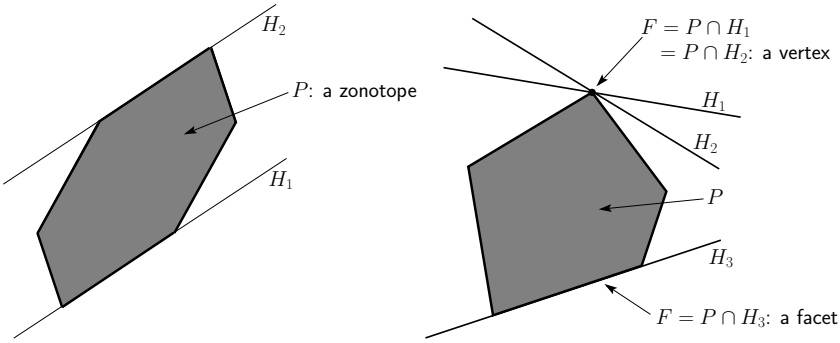
$$[\boldsymbol{\tau}] = \left\{ \boldsymbol{\tau} \mid \tau_i \in [\tau_{i_{\min}}, \tau_{i_{\max}}], \quad \forall\, i,\ 1 \leq i \leq m \right\}. \tag{2}$$

The present work deals with the set of wrenches A defined as

$$A = \{\mathbf{f} \mid \mathbf{f} = \mathbf{W}\boldsymbol{\tau},\ \boldsymbol{\tau} \in [\boldsymbol{\tau}]\} \tag{3}$$

which is, for a given configuration of a parallel manipulator, the set of platform wrenches that can be generated by the actuators with each τ_i in its admissible range $[\tau_{i_{\min}}, \tau_{i_{\max}}]$. Following [1,5], A is called the *available wrench set*.

This set is known to be a *convex polytope*. In fact, as pointed out in [5], since A is the image of the box $[\boldsymbol{\tau}]$ under the linear map given by matrix $\mathbf{W}$, A is affinely isomorphic to a particular type of polytope called a *zonotope* [7]. In Fig. 2, a two-dimensional zonotope is shown. It is the image of a three-dimensional box.



(a) A two-dimensional zonotope P and two parallel facet-defining hyperplanes H_1 and H_1 of P.

(b) A convex polygon P and three supporting hyperplanes H_i (lines). Both H_1 and H_2 define a vertex of P whereas H_3 is a facet-defining hyperplane (a facet of a convex polygon being an edge).

Fig. 1 A two-dimensional zonotope and a two-dimensional convex polytope (polygon).

3 Faces and Representation of a Convex Polytope

Let P be an n-dimensional convex polytope. An inequality $\mathbf{c}^T\mathbf{x} \leq d$, where $\mathbf{c}$ is an n-dimensional column vector and d a scalar, is said to be *valid* for P if it is satisfied for all $\mathbf{x} \in P$. Equivalently, $\mathbf{c}^T\mathbf{x} \leq d$ is valid for P if P is fully included in the halfspace $H^- = \{\mathbf{x} \mid \mathbf{c}^T\mathbf{x} \leq d\}$. An hyperplane $H = \{\mathbf{x} \mid \mathbf{c}^T\mathbf{x} = d\}$ is said to be a *supporting hyperplane* of P if $\mathbf{c}^T\mathbf{x} \leq d$ is a valid inequality for P and $P \cap H$ is not empty.

A *face* F of a convex polytope P is a subset of P which can be written as $F = P \cap H$ for some supporting hyperplane $H = \{\mathbf{x} \mid \mathbf{c}^T\mathbf{x} = d\}$ of P. The dimension $\dim(F)$ of a face F is defined as the dimension of its affine hull $\mathrm{aff}(F)$, the affine hull of F being the smallest affine set containing F or, equivalently, the intersection of all the affine sets that contain F. Faces of dimension 0, 1 and $n-1$ are called *vertices*, *edges* and *facets*, respectively. A *facet-defining hyperplane* is a supporting hyperplane H of P such that $F = P \cap H$ is a facet of P. Figure 2 illustrates these definitions by means of a two-dimensional example (a convex polygon). Let us note that if F is a face of a polytope P, $F = P \cap \{\mathbf{x} \mid \mathbf{c}^T\mathbf{x} = d\}$, then

$$d = \max_{\mathbf{x} \in P} \mathbf{c}^T\mathbf{x}. \tag{4}$$

It is a well-known fact [7, 8] that a convex polytope can be represented either as the convex hull of a finite set of points or as the intersection of a finite set of closed halfspaces. In fact, the facet-defining hyperplanes provide the latter representation since *a full dimensional polytope P is the intersection of the halfspaces bounded by its facet-defining hyperplanes* [7, 8]. Precisely, if $\{F_i,\ 1 \leq i \leq f\}$ is the set of

facets of P, $H_i = \{\mathbf{x} \mid \mathbf{c}_i^T \mathbf{x} = d_i\}$ the facet-defining hyperplane supporting P along F_i ($F_i = P \cap H_i$) and $H_i^- = \{\mathbf{x} \mid \mathbf{c}_i^T \mathbf{x} \leq d_i\}$ the closed halfspace bounded by H_i that contains P, then

$$P = \bigcap_{i=1}^{f} H_i^-. \tag{5}$$

Defining the $f \times n$ matrix $\mathbf{C}$ by $\mathbf{C} = (\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_f)^T$ and the f-dimensional column vector $\mathbf{d}$ by $\mathbf{d} = (d_1, d_2, \ldots, d_f)^T$, Eq. (5) gives P as *the set of solutions to a finite system of linear inequalities* since it implies that $P = \{\mathbf{x} \mid \mathbf{C}\mathbf{x} \leq \mathbf{d}\}$. The representation given in Eq. (5) is minimal in the sense that none of the facet-defining hyperplanes H_i^-, $1 \leq i \leq f$, can be removed.

4 The Available Wrench Set as a System of Linear Inequalities

This section sketches a proof of the characterization stated below which enables the representation of the available wrench set A as a system of finitely many linear inequalities.

The wrench set A has been defined in Section 2 as the image of the box of admissible actuator forces/torques $[\boldsymbol{\tau}]$ under $\mathbf{W}$. In the remainder of this paper, the wrench matrix $\mathbf{W}$ is assumed to have full rank n so that A is full dimensional, i.e., A is of dimension n.

Characterization of the available wrench set facet-defining hyperplanes

First assertion – A facet-defining hyperplane $H = \{\mathbf{x} \mid \mathbf{c}^T \mathbf{x} = d\}$ of the available wrench set A is such that $\mathbf{c}$ is orthogonal to $n - 1$ linearly independent column vectors $\mathbf{w}_i$ of $\mathbf{W}$ and

$$d = \sum_{I^+} \tau_{i_{\max}} \mathbf{c}^T \mathbf{w}_i + \sum_{I^-} \tau_{i_{\min}} \mathbf{c}^T \mathbf{w}_i \tag{6}$$

where I^- and I^+ are index sets defined as

$$I^+ = \{i,\ 1 \leq i \leq m \mid \mathbf{c}^T \mathbf{w}_i > 0\}, \quad I^- = \{i,\ 1 \leq i \leq m \mid \mathbf{c}^T \mathbf{w}_i < 0\}. \tag{7}$$

Second assertion – Conversely, to any set of $n - 1$ linearly independent column vectors $\mathbf{w}_i$ of $\mathbf{W}$, there correspond two facet-defining hyperplanes $H_1 = \{\mathbf{x} \mid \mathbf{c}_1^T \mathbf{x} = d_1\}$ and $H_2 = \{\mathbf{x} \mid \mathbf{c}_2^T \mathbf{x} = d_2\}$ of A. These two hyperplanes are *parallel* (e.g. H_1 and H_2 in Fig. 2) and such that $\mathbf{c}_1 \neq \mathbf{0}$ is orthogonal to the $n - 1$ linearly independent $\mathbf{w}_i$, $\mathbf{c}_2 = -\mathbf{c}_1$ and

$$d_1 = \sum_{I_1^+} \tau_{i_{\max}} \mathbf{c}_1^T \mathbf{w}_i + \sum_{I_1^-} \tau_{i_{\min}} \mathbf{c}_1^T \mathbf{w}_i, \quad d_2 = \sum_{I_2^+} \tau_{i_{\max}} \mathbf{c}_2^T \mathbf{w}_i + \sum_{I_2^-} \tau_{i_{\min}} \mathbf{c}_2^T \mathbf{w}_i \tag{8}$$

where $I_1^+ = \{i \mid \mathbf{c}_1^T \mathbf{w}_i > 0\}$, $I_1^- = \{i \mid \mathbf{c}_1^T \mathbf{w}_i < 0\}$, $I_2^+ = \{i \mid \mathbf{c}_2^T \mathbf{w}_i > 0\}$ and $I_2^- = \{i \mid \mathbf{c}_2^T \mathbf{w}_i < 0\}$. Note that $I_2^+ = I_1^-$, $I_2^- = I_1^+$ and $H_2 = \{\mathbf{x} \mid \mathbf{c}_1^T \mathbf{x} = -d_2 = \sum_{I_1^-} \tau_{i_{\max}} \mathbf{c}_1^T \mathbf{w}_i + \sum_{I_1^+} \tau_{i_{\min}} \mathbf{c}_1^T \mathbf{w}_i\}$.

4.1 Proofs of the First and Second Assertions

Proof of the first assertion. Let $H = \{\mathbf{x} \mid \mathbf{c}^T \mathbf{x} = d\}$ be a facet-defining hyperplane of A and $F = A \cap H$ the corresponding facet. Consider an arbitrary point (a wrench) $\mathbf{x}_F$ of F. Since $\mathbf{x}_F \in A$, we have

$$\mathbf{x}_F = \sum_{i=1}^{m} \tau_i \mathbf{w}_i, \quad \tau_i \in [\tau_{i_{\min}}, \tau_{i_{\max}}] \tag{9}$$

and since $\mathbf{x}_F \in H$, according to Eq. (4), we have $\mathbf{c}^T \mathbf{x}_F = d = \max_{\mathbf{x} \in A} \mathbf{c}^T \mathbf{x}$. Hence, the τ_i which define $\mathbf{x}_F$ in Eq. (9) are such that

$$\sum_{i=1}^{m} \tau_i \mathbf{c}^T \mathbf{w}_i = \max_{\mathbf{x} \in A} \mathbf{c}^T \mathbf{x}. \tag{10}$$

Let us decompose the sum in Eq. (10) as follows

$$\sum_{i=1}^{m} \tau_i \mathbf{c}^T \mathbf{w}_i = \sum_{I^+} \tau_i \mathbf{c}^T \mathbf{w}_i + \sum_{I^-} \tau_i \mathbf{c}^T \mathbf{w}_i + \sum_{I^0} \tau_i \mathbf{c}^T \mathbf{w}_i = \sum_{I^+} \tau_i \mathbf{c}^T \mathbf{w}_i + \sum_{I^-} \tau_i \mathbf{c}^T \mathbf{w}_i \tag{11}$$

where $I^0 = \{i,\ 1 \le i \le m \mid \mathbf{c}^T \mathbf{w}_i = 0\}$ and I^+ and I^- are defined in Eq. (7). According to Eq. (10), since the τ_i maximize the sum in Eq. (11), necessarily, $\tau_i = \tau_{i_{\max}}$ for all $i \in I^+$ and $\tau_i = \tau_{i_{\min}}$ for all $i \in I^-$. In other words, $\mathbf{x}_F$ is given by

$$\mathbf{x}_F = \sum_{I^+} \tau_{i_{\max}} \mathbf{w}_i + \sum_{I^-} \tau_{i_{\min}} \mathbf{w}_i + \sum_{I^0} \tau_i \mathbf{w}_i, \quad \tau_i \in [\tau_{i_{\min}}, \tau_{i_{\max}}] \text{ for all } i \in I^0. \tag{12}$$

Consequently, by definition of the index set I^0, $d = \mathbf{c}^T \mathbf{x}_F$ can be written as

$$d = \sum_{I^+} \tau_{i_{\max}} \mathbf{c}^T \mathbf{w}_i + \sum_{I^-} \tau_{i_{\min}} \mathbf{c}^T \mathbf{w}_i \tag{13}$$

so that Eq. (6) is proved.

Note that any point $\mathbf{x}$ of A which can be written as $\mathbf{x}_F$ in Eq. (12) belongs to the facet-defining hyperplane H since $\mathbf{c}^T \mathbf{x} = d$. Thus, such a point $\mathbf{x}$ belongs to the facet F (since $F = A \cap H$) and we have

$$F = \{\mathbf{x} \mid \mathbf{x} = \sum_{I^+} \tau_{i_{\max}} \mathbf{w}_i + \sum_{I^-} \tau_{i_{\min}} \mathbf{w}_i + \sum_{I^0} \tau_i \mathbf{w}_i,\ \tau_i \in [\tau_{i_{\min}}, \tau_{i_{\max}}] \text{ for all } i \in I^0\}. \tag{14}$$

Moreover, basic properties of affine sets imply that the affine hull $\text{aff}(F)$ of F is

$$\text{aff}(F) = \{\mathbf{x} \mid \mathbf{x} = \sum_{I^+} \tau_{i_{\max}} \mathbf{w}_i + \sum_{I^-} \tau_{i_{\min}} \mathbf{w}_i + \sum_{I^0} \tau_i \mathbf{w}_i,\ \tau_i \in \mathbb{R} \text{ for all } i \in I^0\} \tag{15}$$

and that its dimension is

$$\dim(\text{aff}(F)) = \text{rank}(\{\mathbf{w}_i \mid i \in I^0\}) = \text{rank}(\{\mathbf{w}_i \mid \mathbf{c}^T \mathbf{w}_i = 0\}). \tag{16}$$

Finally, F being a facet of A, $\dim(\text{aff}(F)) = n-1$ and hence Eq. (16) implies that there exists $n-1$ linearly independent $\mathbf{w}_i$ among those of the set $\{\mathbf{w}_i \mid \mathbf{c}^T \mathbf{w}_i = 0\}$. In other words, $\mathbf{c}$ is orthogonal to $n-1$ linearly independent $\mathbf{w}_i$ completing the proof of the first assertion.

Proof of the second assertion. Let us consider a set of $n-1$ linearly independent column vectors $\mathbf{w}_i$ of $\mathbf{W}$ which, without loss of generality, can be assumed to be $\mathbf{w}_1$, $\mathbf{w}_2, \ldots, \mathbf{w}_{n-1}$. Let $\mathbf{c}_1$ be any nonzero vector orthogonal to $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{n-1}$, d_1 be given by Eq. (8) and H_1 be the hyperplane $H_1 = \{\mathbf{x} \mid \mathbf{c}_1^T \mathbf{x} = d_1\}$.

With arguments similar to those used above in the proof of the first assertion, it can be shown that H_1 is a supporting hyperplane of A and that the corresponding face $F_1 = A \cap H_1$ is given by Eq. (14) with $I_1^0 = \{i \mid \mathbf{c}_1^T \mathbf{w}_i = 0\}$ instead of I^0 and I_1^+ and I_1^- instead of I^+ and I^-, respectively. Then, according to Eq. (16) (with F_1, I_1^0 and $\mathbf{c}_1$ in place of F, I^0 and $\mathbf{c}$, respectively) and since the $n-1$ linearly independent $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{n-1}$ belong to $\{\mathbf{w}_i \mid i \in I_1^0\}$, we have $\dim(\text{aff}(F_1)) = n-1$ so that F_1 is a facet of A. H_1 is thus a facet-defining hyperplane of A.

Defining the vector $\mathbf{c}_2$ as $\mathbf{c}_2 = -\mathbf{c}_1$, the hyperplane $H_2 = \{\mathbf{x} \mid \mathbf{c}_2^T \mathbf{x} = d_2\}$ with d_2 defined in Eq. (8) is parallel to H_1 and the same type of arguments as those used above show that H_2 is a facet-defining hyperplane of A which completes the proof.

4.2 A Finite System of Linear Inequalities

Referring to Section 3, the available wrench set A can be written as the intersection of the halfspaces bounded by its facet-defining hyperplanes which provides a representation of A as the set of solutions to a system of linear inequalities.

Let C be a finite set of nonzero vectors $\mathbf{c}_j$, $C = \{\mathbf{c}_j,\ 1 \leq j \leq p\}$, such that, on the one hand, each $\mathbf{c}_j \in C$ is orthogonal to $n-1$ linearly independent columns $\mathbf{w}_i$ of $\mathbf{W}$ and, on the other hand, for any set of $n-1$ linearly independent columns $\mathbf{w}_i$, there exists one and only one $\mathbf{c}_j \in C$ orthogonal to these $n-1$ columns $\mathbf{w}_i$. Then, according to the characterization of the facet-defining hyperplanes of A stated at the beginning of Section 4, we have

$$A = \bigcap_{j=1}^{p} (H_{j1}^{-} \cap H_{j2}^{-}) \tag{17}$$

where $H_{j1}^{-} = \{\mathbf{x} \mid \mathbf{c}_j^T \mathbf{x} \leq d_{j1}\}$ and $H_{j2}^{-} = \{\mathbf{x} \mid (-\mathbf{c}_j)^T \mathbf{x} \leq d_{j2}\}$ are two closed halfspaces such that $\mathbf{c}_j \in C$ and d_{j1} and d_{j2} are defined similarly to d_1 and d_2 in Eq. (8).

Moreover, in order to obtain a *minimal representation*, i.e., a representation in which, to each facet of A, there corresponds one and only one halfspace H_{jk}^{-}, $k = 1$ or 2, C must be such that no two of its vectors are collinear (i.e., $\forall(\mathbf{c}_j, \mathbf{c}_l) \in C \times C$, $\mathbf{c}_j \neq \mathbf{c}_l$, there does not exist α such that $\mathbf{c}_l = \alpha \mathbf{c}_j$). The number of facets f of A is equal to $2p$ if and only if C satisfies this property. When C does not satisfy this property, $f < 2p$. Finally, note that if no set of n columns $\mathbf{w}_i$ of $\mathbf{W}$ is a linearly dependent set then C necessarily satisfies the aforementioned property, *i.e*, $f = 2p$. Note however that a minimal representation is not mandatory in order to test if a required wrench set T is included in A.

5 Hyperplane Shifting Method

The characterization of the available wrench set A presented in the previous section leads naturally to the following method that provides a representation of A as the solution set of a system of linear inequalities. In [5], this method is referred to as the hyperplane shifting method. Compared to the version introduced in [5], step 2 of the one presented below avoids many useless computations. Moreover, it is pointed out how to obtain a minimal representation of A.

This method consists in considering all the possible combinations of $n - 1$ columns of the wrench matrix $\mathbf{W}$ in turn. At the beginning of the method, $j = 0$. For the current combination $\{\mathbf{w}_i \mid i \in I^0\}$, where I^0 is the current subset of $n - 1$ elements of $\{1, 2, \ldots, m\}$, do

Step 1: Test if the $n - 1$ column vectors $\mathbf{w}_i$, $i \in I^0$, are linearly independent. If it is the case, $j = j + 1$, determine a nonzero vector $\mathbf{c}_j$ orthogonal to these $n - 1$ columns $\mathbf{w}_i$ and go to step 2.

Step 2: Let I^+ and I^- be the subsets of $\{1, 2, \ldots, m\}$ defined as $I^+ = \{i \mid \mathbf{c}_j^T \mathbf{w}_i > 0\}$ and $I^- = \{i \mid \mathbf{c}_j^T \mathbf{w}_i < 0\}$. Compute d_{j1} and d_{j2} as follows

$$d_{j1} = \sum_{I^+} \tau_{i_{\max}} \mathbf{c}_j^T \mathbf{w}_i + \sum_{I^-} \tau_{i_{\min}} \mathbf{c}_j^T \mathbf{w}_i, \quad d_{j2} = -\sum_{I^-} \tau_{i_{\max}} \mathbf{c}_j^T \mathbf{w}_i - \sum_{I^+} \tau_{i_{\min}} \mathbf{c}_j^T \mathbf{w}_i. \tag{18}$$

At the end of the method, $j = p$ and the $2p$ halfspaces $H_{j1}^{-} = \{\mathbf{x} \mid \mathbf{c}_j^T \mathbf{x} \leq d_{j1}\}$ and $H_{j2}^{-} = \{\mathbf{x} \mid (-\mathbf{c}_j)^T \mathbf{x} \leq d_{j2}\}$, $j = 1, \ldots, p$, provide a representation of A as stated in Eq. (17). Gathering the vectors $\mathbf{c}_j$ and $-\mathbf{c}_j$ in a matrix $\mathbf{C}$ and the d_{j1} and d_{j2} in a vector $\mathbf{d}$ in an appropriate way, Eq. (17) gives A as the solution set of the system of linear inequalities $\mathbf{C}\mathbf{x} \leq \mathbf{d}$. For this representation of A to be minimal, in Step 1, it must be ensured that $\mathbf{c}_j$ is not collinear to any of the previously computed ones,

i.e., not collinear to any of the $\mathbf{c}_k$, $k = 1, \ldots, j-1$. Indeed, if such a collinearity exists, $\mathbf{c}_j$ can be left out from consideration since it yields a redundant inequality.

Step 1 can be implemented as follows. Let $\mathbf{W}_{I^0}$ be the $n \times n-1$ matrix whose columns are the current $n-1$ column vectors $\mathbf{w}_i$, $i \in I^0$. Let d be the dimension of the nullspace of $\mathbf{W}_{I^0}^T$ where $d \neq 0$ since $\mathbf{W}_{I^0}^T$ has more columns than rows. When $d > 2$, the current $n-1$ column vectors $\mathbf{w}_i$, $i \in I^0$, are linearly dependent. Otherwise, $d = 1$ and $\mathbf{c}_j$ can be any nonzero vector in the nullspace of $\mathbf{W}_{I^0}^T$ since then $\mathbf{w}_i^T \mathbf{c}_j = \mathbf{0}$ for all $i \in I^0$. Hence, with the help of a routine that determines the nullspace of a matrix, the hyperplane shifting method is quite straightforward to implement.

6 Conclusion

This paper has dealt with the characterization of the facet-defining hyperplanes of the available wrench set of a parallel manipulator which enables to represent this set as the set of solutions to a system of linear inequalities. With such a representation, being given a wrench set T that the parallel manipulator is required to generate, it is generally straightforward to test whether or not T is fully included in the available wrench set, i.e., whether or not T is feasible.

Acknowledgements The financial support of the ANR (grant 2009 SEGI 018 01) is greatly acknowledged.

References

1. Bosscher, P., Riechel, A. T., Ebert-Uphoff, I.: Wrench-feasible workspace generation for cable-driven robots. *IEEE Trans. on Robotics*, **22**(5), 890–902 (2006).
2. Firmani, F., Zibil, A., Nokleby, S. B., Podhorodeski, R. P.: Wrench capabilities of planar parallel manipulators. Part I: Wrench polytopes and performance indices. *Robotica*, **26**, 791–802 (2008).
3. Merlet, J.-P.: *Parallel Robots*, 2nd ed., Springer (2006).
4. Lee, J.: A study on the manipulability measures for robot manipulators. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Grenoble, France, pp. 1458–1465 (1997).
5. Bouchard, S., Gosselin, C. M., Moore, B.: On the ability of a cable-driven robot to generate a prescribed set of wrenches. In *Proc. ASME Int. Design Engineering Technical Conf. and Computers and Information in Engineering Conf.*, Brooklyn, NY, Paper No. DETC2008-49518 [CD-ROM] (2008).
6. McMullen, P.: On zonotopes. *Trans. Amer. Math. Soc.*, **159**, 91–109 (1971).
7. Ziegler, G. M.: *Lectures on Polytopes*. Graduate Texts in Mathematics, Vol. 152, Springer-Verlag (1994).
8. Boissonnat, J.-D., Yvinec, M.: *Algorithmic Geometry*, Cambridge University Press (1998).

PART 8

Constraint-Screw System Based Synthesis of Limb Arrangement of the 3-*PUP* Parallel Mechanism

D.M. Gan[1], J.S. Dai[1,2] and D.G. Caldwell[1]

[1]*Italian Institute of Technology, Genoa, Italy;*
e-mail: {dongming.gan, darwin.caldwell}@iit.it
[2]*King's College London, University of London, London, UK;*
e-mail: jian.dai@kcl.ac.uk

Abstract. This paper presents various limb arrangements of a 3-*PUP* parallel mechanism constructed by special *PUP* limbs. The constraints of the *PUP* limb are studied using reciprocal screws, which are used to describe the platform constraint system of the 3-*PUP* parallel mechanism that is divided into a torque and a force constraint set. By analyzing the system orders of the two sets to arrange the constraint screws in the three dimensional space, different types of constraint systems are defined, leading to various limb arrangements of the 3-*PUP* parallel mechanism corresponding to these constraint systems and their instantaneous mobility is given accordingly.

Key words: Parallel mechanism, limb arrangement, synthesis, constraint screw.

1 Introduction

Mathematics gives help to describe the nature and study its properties. In the kinematics world of mechanism research, many mathematics methods [1] have been used to describe geometrical constraints and spatial motions of mechanisms based on concepts introduced by mathematicians and physicists. One of the most-used methods is screw theory [2,3] which attracted more and more attention for the study of parallel mechanisms as it provides naturally relationship of limbs and joints in the mechanisms [4]. Application of screw theory covers kinematcis [5], configuration [6], stability [7] and Jacobian [8], with recent works on constraint [9], singularity [10], mobility [11, 12], conceptual design [13] and so on. Much work has been carried out in parallel mechanism synthesis using screws for rotational [14], translational [15], symmetrical [16], 3T1R [17] and the parallel mechanisms with multiple operation modes [18] and with metamorphic property [19].

However, most synthesis focuses on obtaining new parallel mechanisms with special mobility. In this paper, we present a study on screw systems by studying the system order [20] of the constraint screws from the special *PUP* limb to obtain different limb arrangement of a 3-*PUP* parallel mechanism. This gives a new way to demonstrate all configurations of parallel mechanisms based on special limb arrangements.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_52,

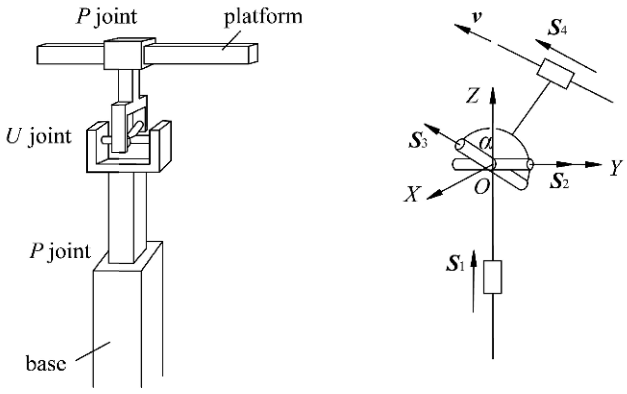


Fig. 1 The *PUP* limb.

2 The *PUP* Limb and Its Geometry Constraints

Figure 1 shows the *PUP* (*P* and *U* stand for prismatic and universal joints respectively) limb studied in this paper. The limb consists of two prismatic joints used to connect to a base and a platform respectively. For convenience, we call these two *P* joints as platform *P* joint and base *P* joint. In the limb, the two *P* joints are connected by a Hooke joint, where two rotational axes of the joint are perpendicular to two prismatic joints in both end of the joint.

In order to get the constraints that are applied to the platform connected to the *PUP* limb, a *OXYZ* limb coordinate system is located at center of the *U* joint with *Y* axis in line with the rotational axis of the *U* joint connected to the base joint and *Z* axis along the base *P* joint as in Fig. 1. For a general configuration, assuming the angle between the rotational axis connected to the platform *P* joint and *Z* axis as α and the unit vector along the platform *P* joint is $v=(v_1, v_2, v_3)^T$, the limb-motion screw system can be given as:

$$\{\boldsymbol{S}_{limb}\} = \left\{ \begin{array}{l} \boldsymbol{S}_1 = \left[0\ 0\ 0\ 0\ 0\ 1\right]^T \\ \boldsymbol{S}_2 = \left[0\ 1\ 0\ 0\ 0\ 0\right]^T \\ \boldsymbol{S}_3 = \left[\sin\alpha\ 0\ \cos\alpha\ 0\ 0\ 0\right]^T \\ \boldsymbol{S}_4 = \left[0\ 0\ 0\ v_1\ v_2\ v_3\right]^T \end{array} \right\}. \tag{1}$$

By calculating reciprocal screws to the limb-motion screw system, the constraint screw system of the limb can be obtained:

$$\{\boldsymbol{S}^r_{limb}\} = \left\{ \begin{array}{l} \boldsymbol{S}^r_1 = \left[0\ 0\ 0\ -\cos\alpha\ 0\ \sin\alpha\right]^T \\ \boldsymbol{S}^r_1 = \left[-v_2\ v_1\ 0\ 0\ 0\ 0\right]^T \end{array} \right\}. \tag{2}$$

Thus, the first constraint is a torque constraint perpendicular to two rotational axes of the U joint and the second constraint is a force constraint perpendicular to two P joints in the limb.

Using this *PUP* limb to assemble parallel mechanisms, a platform will be subjected with a force and a moment by each limb. In this paper, we are discussing the various limb arrangements of a 3-*PUP* parallel mechanism constructed by three of this kind of *PUP* limbs, and there will be six constraints on the platform. As different constraint structures in various mechanism configurations and the constraints in screw forms can be expressed in the three dimensional space, this leads to exploring different screw systems to obtain the corresponding mechanism limb arrangement.

In order to simplify analysis of various topologies of the 3-*PUP* parallel mechanism, an initial configuration of the limb has to be defined and is given as one when the limb does not rotate about any axis of the U joint, leading to $\alpha = 0$ and $v = (0,-1,0)^T$. Thus for this initial configuration, the limb constraint torque will be along the base P joint and the force constraint is perpendicular to the torque constraint, which will be used in the following analysis.

3 Constraint Screw Systems for the 3-*PUP* Parallel Mechanism

As aforementioned, there are six constraints supplied by three *PUP* limbs to the platform of the 3-*PUP* parallel mechanism, each limb provides one force and one torque constraint. The constraint system of the platform can be given as:

$$\left\{\boldsymbol{S}^r_{platform}\right\} = \left\{\boldsymbol{S}_{m1}, \boldsymbol{S}_{f1}, \boldsymbol{S}_{m2}, \boldsymbol{S}_{f2}, \boldsymbol{S}_{m3}, \boldsymbol{S}_{f3}\right\}, \tag{3}$$

where $\boldsymbol{S}_{mi}$ and $\boldsymbol{S}_{fi}$ $(i = 1,2,3)$ denote the torque and force constraint in limb i respectively.

In the three-dimensional space, screws that are linearly dependent on n given linearly independent screws form an order n $(0 \leq n \leq 6)$ screw system. In Eq. (3), there are three force and three torque constraints, they form a system of order k, where $2 \leq kleq6$. Systematically, discussion of different order systems from Eq. (3) can be used to show various limb arrangements of the 3-*PUP* parallel mechanism. Based on this method, the screws can be divided into two sets, one for torques and the other for forces. Consider the initial configuration of the *PUP* limb defined in the previous section, the force is perpendicular to the torque in a limb and the torque is along the base P joint. By listing all structures of the torque screw set first, adding the forces perpendicular to each torque will enumerate all screw order-systems for the platform constraint system, this presents all limb arrangement of the 3-*PUP* parallel mechanism.

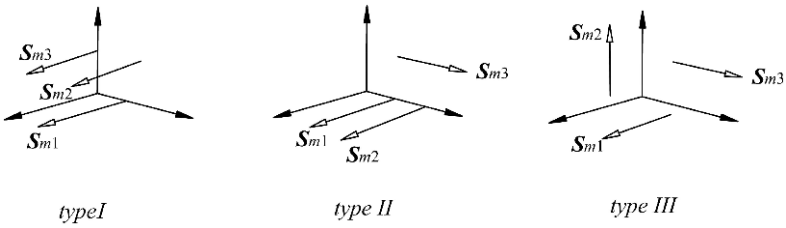


Fig. 2 Three types of torque screw systems.

3.1 Screw Systems Based on Torque Constraints

There are three torque constraint screws in the torque set. They can be arranged to be either dependent or independent from each other, leading to three different orders of screw systems corresponding to three types as in Fig. 2. When the three torque screws are parallel to each other, they are dependent and the torque set has order one which is denoted as *type I*. Two torques are parallel to each other and the third one is independent to them in *type II* with order two while in *type III* all three torques are independent and the order is three.

3.2 Screw Systems Including Force Constraints

Three types of torque-screw systems are presented in Fig. 2, based on which the final various screw systems for the platform constraint screw system can be obtained by further considering the force constraint set. In each limb, the force is perpendicular to the torque, thus in *type I* torque-screw system, the three force constraints are on a plane or different parallel planes as all the torque screws are parallel to each other. Here we only consider the forces on a plane which presents the case that they are on different parallel planes just by actuating the base P joints. Thus there will be four basic structures of the screw system for *type I* torque screw system considering force constraints as in Fig. 3 denoted as *type* I_i ($\imath = 1,2,3,4$). In *type* I_1 the three forces are parallel to each other and the force constraint set has order one while the order is two in *type* I_i ($i = 2,3$), three in *type* I_4 and the three forces intersect at one point or each other, or two parallel to each other and perpendicular to the third. In this study, we only consider some basic configurations of the constraint screws as they can also present the arbitrary cases, like two forces have an arbitrary angle.

Similarly, for *type II* torque constraint system, we can get three sub-types when three force constraint screws in the force set form an order 1, 2 or 3 system while they are parallel to each other, two are parallel to each other and perpendicular to the third or the three point to three different orientations in the space as in Fig. 3, where the sub-types are named as *type* II_i ($i = 1,2,3$).

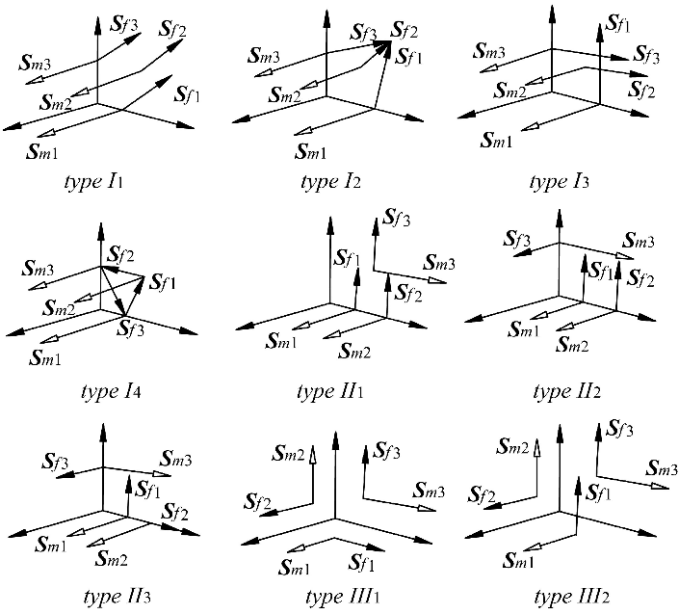


Fig. 3 Screw systems with force constraints.

Based on the torque screw configurations in *type III* torque constraint system, there are two sub-systems when the force constraint set has order 3 or 2 in *type III*$_1$ and *type III*$_2$ as in Fig. 3.

4 Various Limb Arrangements of the 3-*PUP* Parallel Mechanism and Their Instantaneous Mobility Corresponding to the Screw Systems

Using the different types of constraint screw systems in Fig. 3 to arrange the *PUP* limbs gives various configurations of the 3-*PUP* parallel mechanism.

4.1 Limb Arrangement Corresponding to Type I

In *type I* constraint screw systems, the torques are parallel to each other leading to the three limbs being parallel. The force constraint is perpendicular to both *P* joints in the limb. This can be used to arrange the platform *P* joints in the three limbs as all three base *P* joints are parallel to each other along the limbs. Thus four kinds of limb arrangement of the 3-*PUP* parallel mechanism can be obtained as in

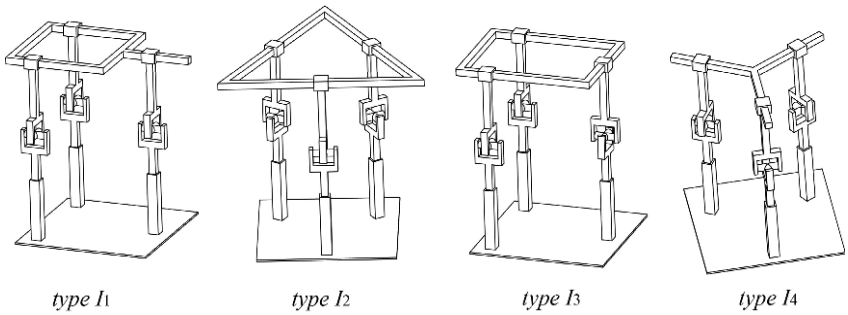


Fig. 4 Limb arrangement corresponding to *type I*

Fig. 4 according to four types of the screw systems in *type I*. In *type* I_1 both torque and force sets have order one, the platform constraint system has order 2 and the platform is subject to one force and one torque constraint. Thus the 3-*PUP* parallel mechanism in the initial configuration has mobility 4 with two rotations and two translations. In *type* I_i (i = 2, 3, 4), the order of the torque set is still 1, the force constraint set has order 2 in *type* I_2 and *type* I_3 and order 3 in *type* I_4 with one screw being redundant with the torque set, giving total order 3 to the platform constraint system and mobility 3 to the 3-*PUP* parallel mechanism with two rotations and one translation.

4.2 Limb Arrangement Corresponding to Type II

Using the same procedure above, there are three kinds of limb arrangement of the 3-*PUP* parallel mechanism corresponding to three types of screw systems in *type II* as in Fig. 5. In *type II* system, the torque-constraint set has order two and the force set has order 1, 2 and 3, leading to the platform constraint system to order 3, 4 and 5. Then the 3-*PUP* parallel mechanism will have mobility 3 in *type* II_1 with one rotation and two translations, mobility 2 in *type* II_2 with one rotation and one translation and mobility 1 in *type* II_3 with one rotation only.

4.3 Limb Arrangement Corresponding to Type III

Similarly, two kinds of limb arrangement of the 3-*PUP* parallel mechanism can be shown in Figure 6 by arranging the limbs based on two subtypes of screw systems in *type III* system in Figure 3. In *type* III_1 both torque and force constraint sets have order three, leading to the platform constraint system of order 6 and the 3-*PUP* parallel mechanism having mobility zero to become a structure. The force constraint

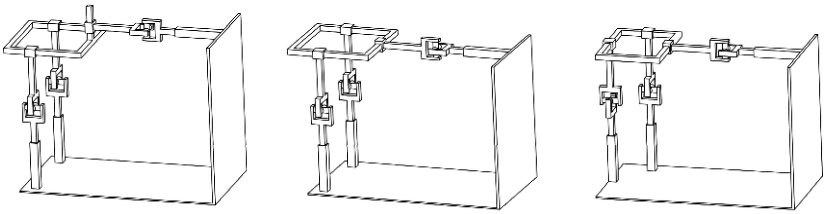


Fig. 5 Limb arrangement corresponding to *type II*

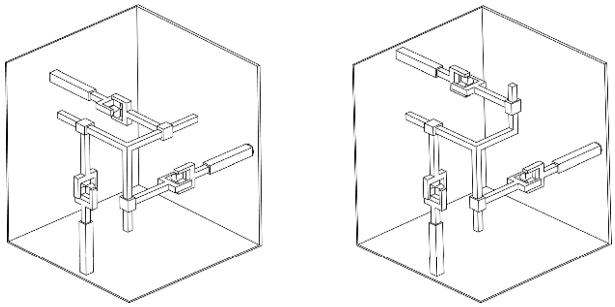


Fig. 6 Limb arrangement corresponding to *type III*.

set has order of 2 in *type III*$_2$ and the mechanism gains one translational degree of freedom to have mobility 1.

5 Conclusions

This paper demonstrates various limb arrangements of a 3-*PUP* parallel mechanism based on the special *PUP* limb. Screw analysis of the constraints of the *PUP* limb shows that the limb can give a moment and a force constraint to the platforms connected to it. Then the platform-constraint system of the 3-*PUP* parallel mechanism will have six constraint screws with three torques and three forces that are divided into two sets. By analyzing the system orders of two sets to arrange the constraint screws in the three-dimensional-space, there are eight basic types of constraint systems, leading to eight kinds of limb arrangement of the 3-*PUP* parallel mechanism with changeable instantaneous mobility from 0 through to 4.

Acknowledgements This work is partially supported by National Natural Science Foundation of China (project No.50775012, 50875027 and 50705010), Beijing Natural Science Foundation (project No. 3053017 and 3082014), National High Technology Research and Development Program

of China (863 program, project No. 2007AA04Z211), and the Engineering and Physical Science Research Council (EPSRC) of the United Kingdom under grant number of EP/E012574/1.

References

1. Dai, J.S.: An historical review of the theoretical development of rigid body displacements from Rodrigues parameters to the finite twist. *Mech. Mach. Theory* **41**, 41–52 (2006).
2. Ball, R.S.: *A Treatise on the Theory of Screws*. Cambridge University Press, Cambridge, UK (1900).
3. Hunt, K.H.: *Kinematic Geometry of Mechanisms*. Oxford University Press, London, UK (1978).
4. Waldron, K.J.: The constraint analysis of mechanisms. *J. Mech.* **1**, 101–114 (1966).
5. Mohamed, M.G., Duffy, J.: A direct determination of the instantaneous kinematics of fully parallel robot manipulators. *ASME J. Mech. Transm. Autom. Des.* **107**, 226–229 (1985).
6. Notash, L.: Uncertainty configurations of parallel manipulators. *Mech. Mach. Theory* **33**, 123–138 (1998).
7. Lee, J., Duffy, J., Keler, M.: The optimum quality index for the stability of in-parallel planar platform devices. *ASME J. Mech. Des.* **121**, 15–22 (1999).
8. Joshi, S.A., Tsai, L.W.: Jacobian analysis of limited-DOF parallel manipulators. *ASME J. Mech. Des.* **124**, 254–258 (2002).
9. Zhao, T.S., Dai, J.S., Huang, Z.: Geometric synthesis of spatial parallel manipulators with fewer than six degrees of freedom. *J. Mech. Eng. Sci.* **216**(12), 1175–1185 (2002).
10. Bonev, I.A., Zlatanov, D., Gosselin, C.M.: Singularity analysis of 3-DOF planar parallel mechanisms via screw theory. *ASME J. Mech. Des.* **125**, 573–581 (2003).
11. Dai, J.S., Huang, Z., Lipkin, H.: Mobility of overconstrained parallel mechanisms. *ASME J. Mech. Des.* **128**(1), 220–229 (2006).
12. Dai, J.S., Rees Jones, J.: Mobility in metamorphic mechanisms of foldable/erectable kinds. *ASME J. Mech. Des.* **121**, 375–382 (1999).
13. Su, H.J., Dorozhkin, D.V., Vance, J.M.: A screw theory approach for the conceptual design of flexible joints for compliant mechanisms. *ASME J. Mech. Rob.* **1**, 041009-1-8 (2009).
14. Fang, Y.F., Tsai, L.W.: Structure synthesis of a class of 3-DOF rotational parallel manipulators. *IEEE Trans. Robot. Autom.* **20**, 117–121 (2004).
15. Kong X.W., Gosselin, C.M.: Type synthesis of 3-DOF translational parallel manipulators based on screw theory. *ASME J. Mech. Des.* **126**(1), 83–92 (2004).
16. Huang, Z., Li, Q.C.: General methodology for type synthesis of lower-mobility symmetrical parallel manipulators and several novel manipulators. *Int. J. Robot. Res.* **21**, 131–145 (2002).
17. Kong X.W., Gosselin, C.M.: Type synthesis of 3T1R 4-DOF parallel manipulators based on screw theory. *IEEE Trans. Robot. Autom.* **20**, 181–190 (2004).
18. Kong X.W., Gosselin, C.M., Richard, P.L.: Type synthesis of parallel mechanisms with multiple operation modes. *ASME J. Mech. Des.* **129**(7), 595–601 (2007).
19. Gan, D.M., Dai, J.S., Liao, Q.Z.: Mobility change in two types of metamorphic parallel mechanisms. *ASME J. Mech. Rob.* **1**, 041007-1-9 (2009).
20. Dai, J.S., Rees Jones, J.: Interrelationship between screw systems and corresponding reciprocal systems and applications. *Mech. Mach. Theory* **36**(5), 633–651 (2001).

On Structural Properties of Sets of Finite Displacement Screws

I.A. Parkin

School of Information Technologies, The University of Sydney, N.S.W. 2006, Australia; e-mail: iaparkin@web55.com

Abstract. Sets of finite displacement screws are considered, each set comprising all motions of a chosen link in a mechanism from its location in a particular starting pose, as measured relative to some fixed link. It is shown, if all screws of such a set have a perpendicularly intersecting *nodal line* in common, then this situation applies in the set for every available pose adopted as starting pose and implies the existence of a *nodal line*-symmetric image of the chosen link which is embedded in the fixed link. It is also shown that, if every such set has the conformation of a *2-system*, then the sum of *principal pitches* is an invariant of all those 2-systems.

Key words: Kinematics, screw theory, finite displacement, finite displacement screw.

1 Introduction

Considerable advances in kinematic analysis, of the Bennett Mechanism in particular, have been made by considering the sets of displacement screws available to individual links of the mechanism in its motion. Early progress was rapid because the geometric structures in which these screws were found to occur conform with the familiar geometry of the 3-system [1] or the 2-sytem [2–4], these screw-theoretic systems having previously been elucidated by Hunt [5] and Phillips [6].

As Huang [1] has pointed out, the rule for composing two finite displacement (screw)s, cited later in Eq. (4), takes the form of a real linear combination of three basis screws – components of a 3-system – when the contributing finite displacements are purely rotational; but when analysis is extended – for example, when generalised displacements are considered in that equation [7] – recognisable structures are left behind and progress toward useful results is significantly slowed.

This paper attempts to assist that progress by establishing some provable structural properties of sets of finite displacement screws. Certain of these have been published before [3], but they were at that time mixed together and telescoped into the properties of the 2-system in the context of analysis of the Bennett Mechanism; here they have been teased apart to reveal their logical independence in order to aid their use in wider domains.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_53, © Springer Science+Business Media B.V. 2010

2 Notation and Basic Geometry

We write a *screw* $\hat{\mathbf{S}}$ as a 3-vector of dual numbers

$$\left.\begin{array}{l} \hat{\mathbf{S}} = |\hat{\mathbf{S}}|\,(1+\varepsilon\, p)\,\hat{\mathbf{s}},\ \ \hat{\mathbf{s}} = \mathbf{l}+\varepsilon\,\mathbf{M} \\ \hat{\mathbf{s}}^{\mathbf{2}} = \mathbf{l}^{\mathbf{2}}+\varepsilon\, 2\,\mathbf{l}\cdot\mathbf{M} = 1+\varepsilon\, 0,\ \ \mathbf{l}\times\mathbf{M} = \mathbf{R} \end{array}\right\} \tag{1}$$

in which ε is a quasi-scalar such that $(a+\varepsilon\, b = c+\varepsilon\, d) \Leftrightarrow\ \ (a = c) \wedge (b = d)$ for all real a, b, c, and d, and satisfying $\varepsilon^2 = 0$. $|\hat{\mathbf{S}}|$ is the real *magnitude* and p is the real *pitch* of the screw $\hat{\mathbf{S}}$, and $\hat{\mathbf{s}}$ (written in lower case) is its *normalised line* which, regarded as a screw in its own right, has unit magnitude and zero pitch. The line $\hat{\mathbf{s}}$ of the screw is spatially located by the *direction 3-vector* of direction cosines $\mathbf{l} = (l,\ m,\ n)$, and by the *moment 3-vector* $\mathbf{M} = (P,\ Q,\ R)$ which determines its *origin-radius* 3-vector $\mathbf{R}$.

We say that two screws $\hat{\mathbf{S}}_\mathbf{1}$, $\hat{\mathbf{S}}_\mathbf{2}$ are *perpendicular* if $\mathbf{l}_\mathbf{1}\cdot\mathbf{l}_\mathbf{2} = 0$, and *orthogonal* if $\hat{\mathbf{S}}_\mathbf{1}\cdot\hat{\mathbf{S}}_\mathbf{2} = 0$, which implies that each intersects the other at right angles. The *cross product* screw $\hat{\mathbf{S}}_\mathbf{1}\times\hat{\mathbf{S}}_\mathbf{2}$ is sited in the *common perpendicular* of $\hat{\mathbf{S}}_\mathbf{1}$, $\hat{\mathbf{S}}_\mathbf{2}$ and has pitch

$$p_1 + p_2 + a\cot(\phi) \tag{2}$$

where, respectively, p_1, p_2 are the pitches of $\hat{\mathbf{S}}_\mathbf{1}$, $\hat{\mathbf{S}}_\mathbf{2}$ and a, ϕ are the distance and angle between the lines of those screws.

3 Specification of a Finite Displacement Screw

We represent the general finite displacement of a body – comprising *translation* through distance d and *rotation* through angle θ, $-\pi < \theta \leq \pi$, about the unit *screw axis* $\hat{\mathbf{s}}$ ($\hat{\mathbf{s}}^{\mathbf{2}} = 1$) – by constructing the dual angle

$$\tfrac{1}{2}\hat{\theta} = \tfrac{1}{2}\theta+\varepsilon\,\tfrac{1}{2}d, \quad \text{so that } \sin\tfrac{1}{2}\hat{\theta} = \sin\tfrac{1}{2}\theta+\varepsilon\,\tfrac{1}{2}d\cos\tfrac{1}{2}\theta,$$

and by then writing the *sin-screw*

$$\hat{\mathbf{S}} = \sin\tfrac{1}{2}\hat{\theta}\,\hat{\mathbf{s}} = \sin\tfrac{1}{2}\theta\,(1+\varepsilon\, P_S)\,\hat{\mathbf{s}} \quad \text{where} \quad P_S = \tfrac{1}{2}d/\tan\tfrac{1}{2}\theta. \tag{3}$$

The sin-screw resultant, $\hat{\mathbf{S}}$, of successively applying two such displacement screws, first $\hat{\mathbf{S}}_\mathbf{1} = \sin\frac{1}{2}\hat{\theta}_1\,\hat{\mathbf{s}}_\mathbf{1}$ and then $\hat{\mathbf{S}}_\mathbf{2} = \sin\frac{1}{2}\hat{\theta}_2\,\hat{\mathbf{s}}_\mathbf{2}$, is conveniently written in the form

$$\begin{bmatrix} \cos\frac{1}{2}\hat{\theta} \\ \hat{\mathbf{S}} \end{bmatrix} = \begin{bmatrix} \cos\frac{1}{2}\hat{\theta}_1\cos\frac{1}{2}\hat{\theta}_2\ -\ \hat{\mathbf{S}}_\mathbf{1}\cdot\hat{\mathbf{S}}_\mathbf{2} \\ \cos\frac{1}{2}\hat{\theta}_2\,\hat{\mathbf{S}}_\mathbf{1}\ +\ \cos\frac{1}{2}\hat{\theta}_1\,\hat{\mathbf{S}}_\mathbf{2}\ -\ \hat{\mathbf{S}}_\mathbf{1}\times\hat{\mathbf{S}}_\mathbf{2} \end{bmatrix}, \tag{4}$$

which is a dualisation of *Rodrigues's equations* and comprises the *biquaternion product* rule [8].

4 Sets of Relocation Screws

Consider a mechanism which, among its links, has two which are especially identified as link *Fix* and, a coupler, link *Cpl*. Our interest lies in the set of finite displacement screws – specified in a frame in which link *Fix* is fixed – which describe displacements of the coupler *Cpl* from its location in some particular pose (or configuration) of the mechanism, A, to its location in every general pose G available to the mechanism.

Since, for the present purpose, we shall not be concerned with alternation of links but only with alternation of poses, it will suffice to omit the names of the links from our symbolism and to write the typical such displacement of the coupler *Cpl*, relative to the fixed link *Fix*, as the sin-screw

$$\hat{\mathbf{S}}_{A\to G} = \sin\tfrac{1}{2}\hat{\theta}_{A\to G}\hat{\mathbf{s}}_{A\to G} \quad \text{where} \quad \hat{\theta}_{A\to G} = \theta_{A\to G} + \varepsilon d_{A\to G}. \tag{5}$$

We intend that the set of screws $\hat{\mathbf{S}}_{A\to G}$ should comprise the complete set of motions available to the coupler *Cpl* as the mechanism moves from pose A to any one of the poses G generally available to it. We illustrate the comprehensive nature of this description by showing that, from this set for pose A, we may derive the corresponding set of screws $\hat{\mathbf{S}}_{B\to G}$ appropriate to any other available pose, B, of the mechanism.

We observe, firstly, that the screw $\hat{\mathbf{S}}_{B\to A} = -\hat{\mathbf{S}}_{A\to B}$, which carries the coupler from its location in pose B to that in pose A, is a member of the latter set just as its negative, $\hat{\mathbf{S}}_{A\to B}$, is a member of the former set; the common line of these equal and opposite screws is present in both sets.

Now the displacement $\hat{\mathbf{S}}_{B\to G}$ of the coupler from its location in pose B to that in some general pose G may be achieved by, firstly, moving it from its location in pose B to that in pose A by use of the screw $-\hat{\mathbf{S}}_{A\to B}$ and, subsequently, from there to its location in pose G by use of $\hat{\mathbf{S}}_{A\to G}$, We compose the displacements of this sequence, both drawn from the set for pose A, by use of eqn. (4) to obtain

$$\begin{bmatrix} \cos\frac{1}{2}\hat{\theta}_{B\to G} \\ \hat{\mathbf{S}}_{B\to G} \end{bmatrix} = \begin{bmatrix} \cos\frac{1}{2}\hat{\theta}_{A\to B}\cos\frac{1}{2}\hat{\theta}_{A\to G} + \hat{\mathbf{S}}_{A\to B}\cdot\hat{\mathbf{S}}_{A\to G} \\ -\cos\frac{1}{2}\hat{\theta}_{A\to G}\hat{\mathbf{S}}_{A\to B} + \cos\frac{1}{2}\hat{\theta}_{A\to B}\hat{\mathbf{S}}_{A\to G} + \hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{S}}_{A\to G} \end{bmatrix} \tag{6}$$

where, respectively, $\frac{1}{2}\hat{\theta}_{B\to G}$, $\frac{1}{2}\hat{\theta}_{A\to B}$, $\frac{1}{2}\hat{\theta}_{A\to G}$ are the dual half-angles inherent in the sinusoids of the screws $\hat{\mathbf{S}}_{B\to G}$, $\hat{\mathbf{S}}_{A\to B}$, $\hat{\mathbf{S}}_{A\to G}$.

Using this expression for $\hat{\mathbf{S}}_{B\to G}$, we find a useful constant relationship:

$$\begin{aligned} \left(\hat{\mathbf{S}}_{B\to A}\times\hat{\mathbf{S}}_{B\to G}\right)^2 &= \cos^2\tfrac{1}{2}\hat{\theta}_{A\to B}\left(\hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{S}}_{A\to G}\right)^2 + \left[\hat{\mathbf{S}}_{A\to B}\times\left(\hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{S}}_{A\to G}\right)\right]^2 \\ &= \left(\cos^2\tfrac{1}{2}\hat{\theta}_{A\to B} + \hat{\mathbf{S}}_{A\to B}^2\right)\left(\hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{S}}_{A\to G}\right)^2 \\ &= \left(\hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{S}}_{A\to G}\right)^2 \end{aligned} \tag{7}$$

since $\hat{\mathbf{S}}^2_{A\to B} = \sin^2 \frac{1}{2}\hat{\theta}_{A\to B}$; this will later be found to be useful in indicating the equality of pitches of the cross-product screws.

5 Presence of a Nodal Line

Consider that the screws $\hat{\mathbf{S}}_{A\to G}$ associated with the arbitrarily chosen pose A enjoy the special property that each is orthogonal to – i.e. intersects at right angles – a particular line which we shall refer to as the *nodal line* of the set. We shall demonstrate that the set of screws associated with any other available pose, B, enjoys the same property.

Let $\hat{\mathbf{n}}_A$, such that $\hat{\mathbf{n}}_A^2 = 1$, be the unit nodal line of the set of screws for pose A so that, by its definition,

$$\hat{\mathbf{S}}_{A\to G} \cdot \hat{\mathbf{n}}_A = 0$$

for every available pose G. For any available pose B, we will show that the unit line

$$\hat{\mathbf{n}}_B = \cos\tfrac{1}{2}\hat{\theta}_{A\to B}\,\hat{\mathbf{n}}_A \;+\; \hat{\mathbf{S}}_{A\to B} \times \hat{\mathbf{n}}_A$$

is orthogonal to every screw $\hat{\mathbf{S}}_{B\to G}$ of the set for pose B and is, therefore, a nodal line for that set. Using the expression for $\hat{\mathbf{S}}_{B\to G}$ provided by eqn. (6), we evaluate, for every available pose G,

$$\begin{aligned}\hat{\mathbf{S}}_{B\to G} \cdot \hat{\mathbf{n}}_B = &-\cos\tfrac{1}{2}\hat{\theta}_{A\to B}\cos\tfrac{1}{2}\hat{\theta}_{A\to G}\hat{\mathbf{S}}_{A\to B}\cdot\hat{\mathbf{n}}_A \;+\; \cos^2\tfrac{1}{2}\hat{\theta}_{A\to B}\hat{\mathbf{S}}_{A\to G}\cdot\hat{\mathbf{n}}_A\\ &+\cos\tfrac{1}{2}\hat{\theta}_{A\to B}\hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{S}}_{A\to G}\cdot\hat{\mathbf{n}}_A - \cos\tfrac{1}{2}\hat{\theta}_{A\to G}\hat{\mathbf{S}}_{A\to B}\cdot\hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{n}}_A\\ &+\cos\tfrac{1}{2}\hat{\theta}_{A\to B}\hat{\mathbf{S}}_{A\to G}\cdot\hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{n}}_A \;+\; \hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{S}}_{A\to G}\cdot\hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{n}}_A\end{aligned}$$

so that, since the first, second and sixth terms vanish under the nodal line property of $\hat{\mathbf{n}}_A$, and the fourth vanishes under a standard screw identity,

$$\begin{aligned}\hat{\mathbf{S}}_{B\to G} \cdot \hat{\mathbf{n}}_B &= \cos\tfrac{1}{2}\hat{\theta}_{A\to B}\left(\hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{S}}_{A\to G} + \hat{\mathbf{S}}_{A\to G}\times\hat{\mathbf{S}}_{A\to B}\right)\cdot\hat{\mathbf{n}}_A\\ &= 0\end{aligned}$$

since the remaining terms cancel, being negatives, the one of the other. It follows that $\hat{\mathbf{n}}_B$ is a nodal line for the set of screws $\hat{\mathbf{S}}_{B\to G}$ associated with pose B.

Since $\hat{\mathbf{n}}_A^2 = 1$ and $\hat{\mathbf{S}}_{A\to B}\cdot\hat{\mathbf{n}}_A = 0$, we observe that

$$\begin{aligned}\hat{\mathbf{n}}_A \times \hat{\mathbf{n}}_B &= \hat{\mathbf{S}}_{A\to B} \;-\; \hat{\mathbf{S}}_{A\to B}\cdot\hat{\mathbf{n}}_A\,\hat{\mathbf{n}}_A\\ &= \hat{\mathbf{S}}_{A\to B}\end{aligned}$$

which indicates that the screw $\hat{\mathbf{S}}_{A\to B}$ lies on the common perpendicular of the nodal lines $\hat{\mathbf{n}}_A$, $\hat{\mathbf{n}}_B$. We deduce, also, that the nodal line $\hat{\mathbf{n}}_B$ is translated and rotated about the screw $\hat{\mathbf{S}}_{A\to B}$ from the location of the nodal line $\hat{\mathbf{n}}_A$ by exactly half of the distance

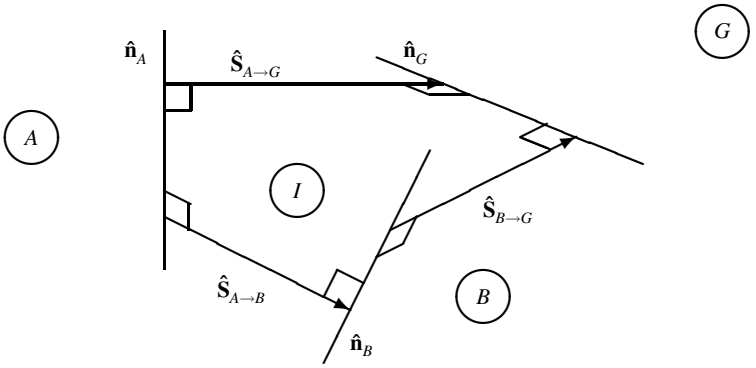


Fig. 1 The screw triangle.

and angle through which the coupler *Cpl* displaces about $\hat{\mathbf{S}}_{A\to B}$ in passing from its location in pose A to that in pose B.

We may summarise the results of this section in the following theorem: *if the set of relocation screws for a given pose has a* nodal line, *orthogonal to all members, then the set for every pose does so; and the cross product of the unit nodal lines for the sets of any two poses is exactly the relocation sin-screw which carries the coupler from its location in one of those poses to the other.*[1]

6 Interpretation

For a geometric interpretation we refer to the *screw triangle* [8] depicted in Fig. 1 with attached symbols appropriate to the previous discussion. The figure shows an imaginary image I of a displacing body – such as the coupler *Cpl* introduced earlier – which has been reflected into three actual instances of the body, namely A, B, G, by three line-mirrors, respectively $\hat{\mathbf{n}}_A$, $\hat{\mathbf{n}}_B$, $\hat{\mathbf{n}}_G$; it is well established that for three arbitrarily disposed locations of a body, such as A, B, G, suitable locations can always be found for the body-image I and the three line-mirrors.

The equalities of distance and angle established by the actions of the line mirrors ensure that the Mozzi–Chasles axis for displacement of the body from A to B, for example, lies on the common perpendicular of the corresponding line-mirrors $\hat{\mathbf{n}}_A$ and $\hat{\mathbf{n}}_B$; in fact, if those mirror lines are of unit magnitude, the sin-screw for that displacement is given by

$$\hat{\mathbf{S}}_{A\to B} = \hat{\mathbf{n}}_A \times \hat{\mathbf{n}}_B, \tag{8}$$

[1] If there is no unique nodal line, the theorem applies equally in every subset of screws for which a common perpendicularly intersecting line exists; limitations of space do not permit this aspect to be developed here.

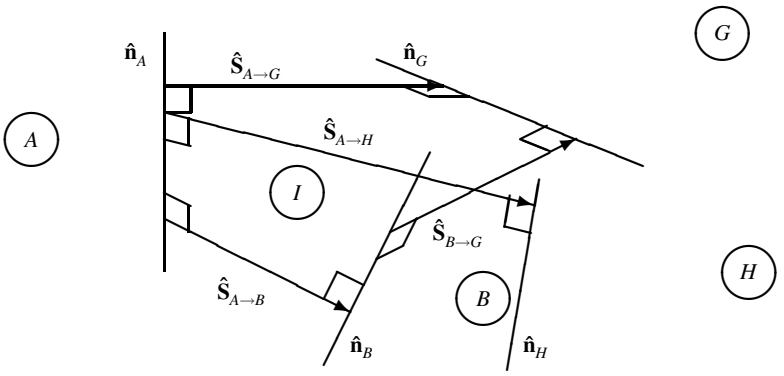


Fig. 2 Adding a further location H, generated by reflection of I in $\hat{\mathbf{n}}_H$.

and correspondingly for the other displacements.

We see that the results of the preceding section apply in Figure 1: the mirror lines act as nodal lines, there being one such present in the set of screws for displacement away from each of the locations A, B, and G.

Consider, that we now add to the situation a further location, H, of the body: for H generally disposed, the construction of a screw triangle to encompass locations A, B, and H will generally require creation of a new body-image, differently located from I and, as well as a new mirror $\hat{\mathbf{n}}_H$, will require the addition of new mirrors, different from $\hat{\mathbf{n}}_A$ and $\hat{\mathbf{n}}_B$ which can reflect this new body-image into A and B.

However, as depicted in Fig. 2, more specialised locations for H are clearly available which can be obtained from the original body-image I by suitable placement of a new line-mirror $\hat{\mathbf{n}}_H$; such arrangement preserves the uniqueness of the body-image I and the 1-to-1 relationship between each body location and the line-mirror which creates it by reflection of I.

7 Mechanism with Explicit Line Symmetry

Consider that the body-image of the preceding section is made explicit in the mechanism of interest so that, in every pose, there exists a line-mirror axis which reflects the fixed link *Fix* into the coupler *Cpl* (and *vice versa*); that is to say that the link *Fix* can be carried into identity with *Cpl* by a pure half-turn rotation, without translation, about that axis.

Since, by the definition of Eq. (3), the sin-screw for such a displacement is a unit line, we may conveniently represent both the location of the line-mirror, and its action, by a unit line; e.g. in arbitrarily chosen poses A and B, by unit lines $\hat{\mathbf{n}}_A$ and $\hat{\mathbf{n}}_B$. Now, as measured relative to link *Fix*, the displacement $\hat{\mathbf{S}}_{A\to B}$ suffered by link *Cpl* as the mechanism moves from pose A to pose B may be composed from,

firstly, a negative half-turn about $\hat{\mathbf{n}}_A$ (into the fixed location of *Fix*) followed by a positive half-turn about $\hat{\mathbf{n}}_B$. Composing the resultant of these displacements by use of Eq. (4), we find, since the cosine terms vanish,

$$\hat{\mathbf{S}}_{A\to B} = \hat{\mathbf{n}}_A \times \hat{\mathbf{n}}_B. \tag{9}$$

It follows, since B is an arbitrarily chosen pose, that the line-mirror axis of the *Cpl*, *Fix* pair is an orthogonal nodal line for *all* screws $\hat{\mathbf{S}}_{A\to B}$ for displacement of the link *Cpl* from pose A to another pose; and since A is an arbitrarily chosen pose, that this is so in every pose.

8 Conformation with a 2-System

Consider that each displacement screw $\hat{\mathbf{S}}_{A\to G}$ in the set associated with pose A can be expressed as a linear combination

$$\hat{\mathbf{S}}_{A\to G} = \beta\,\hat{\mathbf{S}}_{A\to B} + \gamma\hat{\mathbf{S}}_{A\to C} \tag{10}$$

where $\hat{\mathbf{S}}_{A\to B}$, $\hat{\mathbf{S}}_{A\to C}$ are particular members of that set and β, γ are real coefficients. We examine the possibility that this property holds in the set of displacement screws of other poses, in particular, of pose B. By the use of Eq. (6) we obtain

$$\hat{\mathbf{S}}_{B\to G} = \left[\cos\tfrac{1}{2}\hat{\theta}_{A\to G} - \beta\cos\tfrac{1}{2}\hat{\theta}_{A\to B} - \gamma\cos\tfrac{1}{2}\hat{\theta}_{A\to C}\right]\hat{\mathbf{S}}_{B\to A} \;+\; \gamma\hat{\mathbf{S}}_{B\to C} \tag{11}$$

in which, by use of the same Eq. (6),

$$\hat{\mathbf{S}}_{B\to C} = -\cos\tfrac{1}{2}\hat{\theta}_{A\to C}\hat{\mathbf{S}}_{A\to B} + \cos\tfrac{1}{2}\hat{\theta}_{A\to B}\hat{\mathbf{S}}_{A\to C} + \hat{\mathbf{S}}_{A\to B}\times\hat{\mathbf{S}}_{A\to C}.$$

We observe in eqn. (11) that, while the screws associated with pose B can be expressed as a combination of two of their number, the coefficients of combination – specifically, the quantity in square brackets – are not generally real numbers, but dual.

Nevertheless, it is known [2–4] that there are mechanisms in which displacement screws occur as linearly combinations of a pair of basis screws, combined with real coefficients, in each of a succession of poses; for such we may make the following useful observation.

It is well-established [5] that any such set of screws contains a pair of *principal screws* which intersect one another at right angles and which can serve as basis screws for the set. If, then, in the set of screws associated with pose A we were to name these principal screws $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$, we can write

$$\hat{\mathbf{S}}_{A\to B} = \alpha\,\hat{\mathbf{X}} + \beta\,\hat{\mathbf{Y}} \qquad \hat{\mathbf{S}}_{\mathbf{A}\to\mathbf{G}} = \gamma\hat{\mathbf{X}} + \delta\,\hat{\mathbf{Y}} \tag{12}$$

where α, β, γ, δ are real coefficients. We derive

$$\hat{\mathbf{S}}_{A\to B} \times \hat{\mathbf{S}}_{A\to G} = (\alpha\,\delta - \beta\,\gamma)\,\hat{\mathbf{X}} \times \hat{\mathbf{Y}} \tag{13}$$

which indicates, through Eq. (2), that the pitch of the screw on the left is equal to the sum of pitches of the principal screws $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$. But Eq. (7) shows the pitch of the screw-form on the left to be invariant between poses A and B: so we see the sum of pitches of the principal screws to be an invariant throughout the linearly combined screw sets associated with all poses.

9 Conclusion

A number of structural properties of sets of finite displacement screws have been elucidated, each set comprising all motions of some chosen link in a mechanism from its location in a particular pose.

Acknowledgements The author acknowledges the assistance of the School of Information Technologies, the University of Sydney, in preparation of this paper.

References

1. C. Huang. On the finite screw system of the third order associated with a revolute-revolute chain. In *Advances in Design Automation, Proc. 1993 ASME Design Technical Conf.*, Albuquerque, New Mexico, September 19–22, Vol. 2, pp. 81–89 (1993).
2. C. Huang. The cylindroid associated with finite motions of the Bennett mechanism. *Journal of Mechanical Design, Trans. ASME*, **119**, 521–524 (1997).
3. I.A. Parkin and J. Preston. Analysis of the Bennett mechanism by means of finite displacement screws. In *Proceedings of a Symposium Commemorating the Legacy, Works, and Life of Sir Robert Stawell Ball*, Trinity College, University of Cambridge, July (2000).
4. A. Perez and J.M. McCarthy. Bennett's linkage and the cylindroid. *Mechanism and Machine Theory* **37**, 1245–1260 (2002).
5. K.H. Hunt. *Kinematic Geometry of Mechanisms.* Clarendon Press, Oxford (1990).
6. J. Phillips. *Freedom in Machinery.*, Vols. 1 and 2. Cambridge University Press (1990).
7. I.A. Parkin. Dual systems of finite displacement screws in the screw triangle. *Mechanism and Machine Theory* **32**(8), 993–1003 (1997).
8. O. Bottema and B. Roth. *Theoretical Kinematics.* North-Holland Publishing Company, Amsterdam (1979). Reprinted Dover, New York (1990).

An Autonomous and Safe Homing Strategy for Parallel Kinematic Five-Bar Manipulators

F. Dietrich, K. Kaiser, R.J. Ellwood and A. Raatz

Institute of Machine Tools and Production Technology,
Technische Universität Braunschweig, 38106 Braunschweig, Germany;
e-mail: {f.dietrich, k.kaiser, j.ellwood, a.raatz}@tu-bs.de

Abstract. This article is concerned with the homing of parallel kinematic five-bar manipulators that have non-absolute position sensors. A strategy for joint-based homing is proposed which requires simple homing switches only. This strategy allows the drives of a five-bar mechanism to be referenced without user interaction. Though well established for single drives its application to coupled kinematics is quite challenging, due to the risk of workspace violation and uncontrollable approaching of singularities. Mathematical relations are presented which are necessary to place the home position in such a way that homing can be accomplished safely from any starting point. Under consideration of the results presented, an autonomous and safe homing procedure can be implemented on five-bar manipulators with the help of simple homing switches.

Key words: Five-bar mechanism, parallel kinematic, homing, incremental position encoders.

1 Introduction

The power-up sequence of a parallel kinematic manipulator [1] is challenging if the position encoders attached to the joints do not provide the absolute position. Then a homing procedure is required to find a predefined reference position. During homing of a parallel kinematic manipulator, singularities and violation of the workspace borders must be avoided, just as during normal operation. But unfortunately any monitoring of these properties (e.g. singularity avoidance [2]) requires absolute position information which is not available until the homing procedure has been finished successfully. For this reason the homing of many parallel manipulators cannot be executed autonomously but must be supervised by a trained operator.

Package size is one of the reasons why absolute encoders might not be favorable. For example, the miniaturized five-bar manipulator PARVUS has been put into operation successfully, having incremental encoders attached to the motor side of strain wave gears [3]. There, the decision to use incremental encoders in combination with strain wave gears results in high position resolution. The five-bar manipulator of the Collaborative Research Center 562 is a non-miniaturized example, which is also

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_54,

equipped with incremental encoders [4]. This manipulator features 4-zone homing switches which already allow autonomous execution of the homing routine.

This article focusses on the homing procedure of the five-bar mechanism with 2-zone homing switches. An autonomous homing strategy is investigated that only needs simple homing switches. It will be shown that the successful implementation of this strategy is tied to suitable placement of the reference point.

1.1 The Five-Bar Mechanism

In this article a planar five-bar mechanism is considered as depicted in Fig. 1. A miniaturized realization of this mechanism can be found in [5]. It possesses two drives A_1, A_2 which means that it is neither redundant nor under-actuated. The manipulator is equipped with incremental position encoders that indicate the angles of the actuated joints ϕ_1, ϕ_2 of A_1, A_2. Additionally A_1 and A_2 feature the homing switches H_1 and H_2, respectively. A homing switch H is defined similarly to [6]:

$$H(\phi) = \begin{cases} -1, & \phi + \alpha \in [0°, 180°] \\ 1, & else \end{cases} \tag{1}$$

The angles ϕ_1, ϕ_2 where H switches yield the absolute home position. Hence the offset α is available to choose the placement of the home position. Based on Eq. (1) the homing vector $\boldsymbol{H}$ is defined, having four possible values:

$$\boldsymbol{H} = \left[H(\phi_1), H(\phi_2)\right]^T = [H_1, H_2]^T, \tag{2}$$

$$\boldsymbol{H}_1 = \left[1,1\right]^T, \quad \boldsymbol{H}_2 = \left[1,-1\right]^T, \quad \boldsymbol{H}_3 = \left[-1,1\right]^T, \quad \boldsymbol{H}_4 = \left[-1,-1\right]^T. \tag{3}$$

2 A Strategy for Autonomous Homing

$\boldsymbol{H}$ indicates the direction in which the home position is located in joint space relatively to the actual position. Hence the following strategy is proposed:

Phase 1: "Move both drives simultaneously, each of them towards its individual home position. Move both drives with the same constant velocity."
Switch Condition: "Proceed to Phase 2 when one homing switch value changes."
Phase 2: " Stop the drive which reached its home position. Keep on moving the other drive with the same velocity."
End Condition: "Exit when the second home value switches."

This joint-based homing strategy is well established in linear stages and rotary positioning systems that are commercially available. But this strategy does not inherently work with coupled kinematics due to singularities and possible workspace

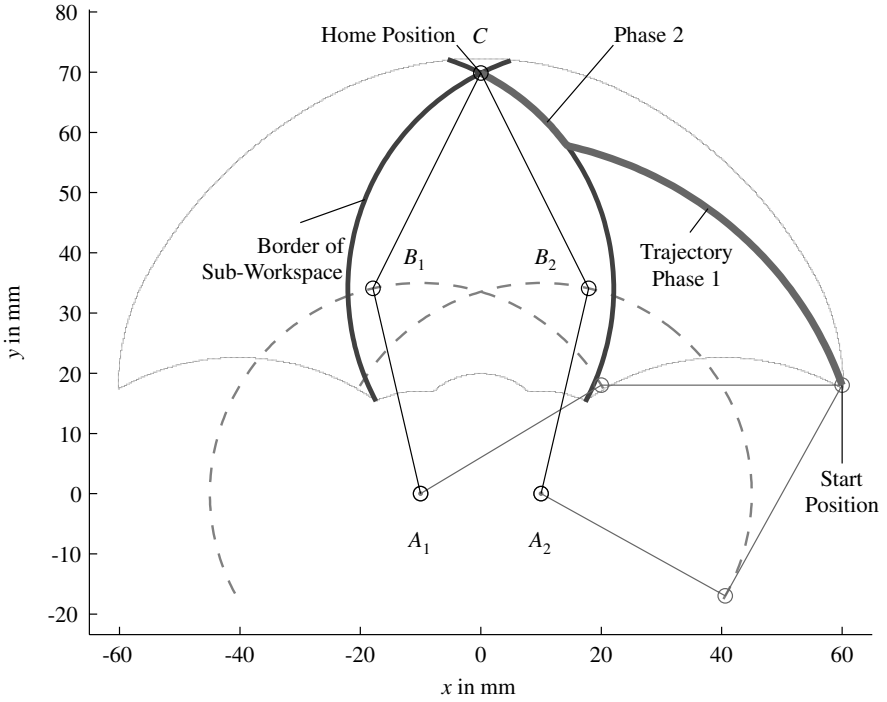


Fig. 1 Exemplary homing sequence of a five-bar mechanism.

violations during the homing procedure [7]. Figure 1 exemplarily shows such a homing movement. The mechanism is drawn in an arbitrary start position ($x = 60\,\text{mm}$, $y = 17\,\text{mm}$) and in a predefined home position ($x = 0\,\text{mm}$, $y = 68\,\text{mm}$). The trajectory on which the end-effector travels during homing can be seen. The trajectory changes apruptly when phase 2 begins. The end-effector enters a circular trajectory at this point.

3 Analysis of the Homing Strategy Proposed

Under practical considerations the following two questions arise:

1. Is there a home position within the workspace which can be reached safely from every location in the workspace?
2. If such a home position exists, where is it located? If multiple of them exist, how can the set of home positions be determined?

At the five-bar no singularities are present within the workspace.[1] Hence unintended situations only occur when the end-effector violates the border of the workspace. This means that a homing strategy for the five-bar must ensure that the end-effector remains within the workspace.

3.1 Segmentation into Two Phases

The following analysis considers the two phases in reverse order which can be stated in the following question: Is every point in the workspace reachable from the home position?

3.1.1 Analysis of the Second Phase

In the second phase the end-effector moves on a circular trajectory which is centered around B_1 or B_2 (dependendent on which homing switch is reached first). The circular trajectories represent borders which divide the workspace into four sub-workspaces. The following text will refer to them as the north, the east, the west and the south sub-workspace (above, right, left and below the home position in Fig. 1, respectively). Based on the way the sub-workspaces are defined each sub-workspace has its specific homing vector.

Once phase 2 is reached, the end-effector will safely arrive at the home position. This can be seen in Fig. 1: It is not possible to place the home position at a location where one of the circular trajectories violates the workspace between the border of the workspace and the home position. Hence, phase 2 can be considered to be safe.

3.1.2 Analysis of the First Phase

The analysis of the first phase can be reduced to the analysis of the outer workspace border. This results due to the definition of the workspace which excludes all singularities (type 1 and type 2) from the workspace. Hence it is sufficient to ensure the safety of the positions which are located on the outer workspace border (all inner points are safe by definition). These positions are safe if the corresponding velocity vector which results from the homing movement does not point outside the workspace. If this property can be garantueed the end-effector will eventually move along the border of the workspace but will sooner or later turn back towards the interior of the workspace.

A velocity vector $\dot{\boldsymbol{x}}(\boldsymbol{x})$ that corresponds to an arbitrary end-effector position $\boldsymbol{x}$ is defined *valid* if it does not drive the end-effector into an unintended situation after infinitesimal time dt. This means that all velocity vectors which correspond

[1] It is not assumed that the five-bar is operated on both sides of the type-2-singularity.

to positions inside the workspace are valid, except for the velocity vectors which correspond to positions on the border of the workspace. The validity of the velocity vectors in the inner of the workspace results from the definition of the workspace which excludes all singularities (type 1 and type 2) from the workspace. Based on this definition a sub-workspace is defined *valid* if all velocity vectors that belong to it are valid (including the section of the border of the workspace affected).

For practical analysis the border of the workspace is discretized. Then the Jacobian $\boldsymbol{J}$ [8] is calculated for each point of the discretized border of the workspace:

$$\boldsymbol{J} = \frac{\partial \dot{\boldsymbol{x}}}{\partial \dot{\boldsymbol{q}}}. \tag{4}$$

Afterwards the normalized velocity vectors which may occur for all homing vectors are calculated for each point:

$$\dot{\boldsymbol{x}}_i(\boldsymbol{x}, \boldsymbol{H}_i) = \boldsymbol{J}(\boldsymbol{x}) \cdot \boldsymbol{H}_i \qquad i = 1 \ldots 4. \tag{5}$$

Figure 2 shows the discretized border of the workspace and some of the velocity vectors which correspond to these points.

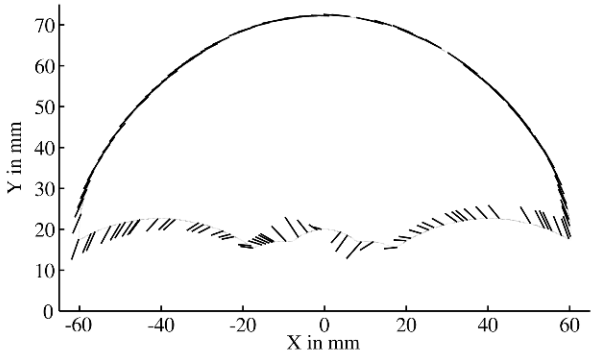
3.2 Choice of Workspace Partitioning and its Restrictions

It can be seen from Fig. 2 that there are velocity vectors that point outwards the workspace. There is only one suitable set of homing vectors which can be used for the five-bar: In the northern sub-workspace $\boldsymbol{H}_3$ is the only suitable homing vector, in the eastern sub-workspace this is $\boldsymbol{H}_1$, in the western $\boldsymbol{H}_4$ and in the southern sub-workspace $\boldsymbol{H}_2$.

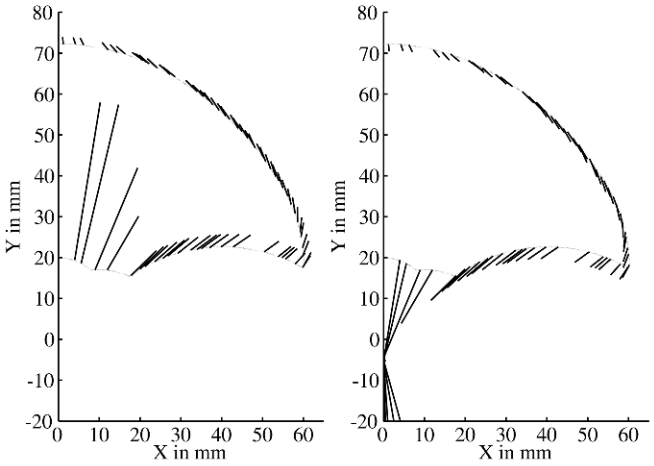
Consequently the restrictions for the placement of the home position are elaborated: If $\boldsymbol{H}_3$ is used for the northern sub-workspace, its validity is always given.[2] The southern sub-workspace is also always valid, because when using $\boldsymbol{H}_2$ the whole lower edge of the workspace is valid. Regarding the eastern and western sub-workspace the situation is different: Consider the lower edge of the eastern sub-workspace ($\boldsymbol{H}_1$, Fig. 2a). Towards $x = 0$ there are points on the workspace border that are invalid. Hence the eastern sub-workspace may cover the outermost bow segment of the lower edge of the workspace but not the inner ones. The same applies to the western workspace due to symmetry.

On this basis the set of valid home positions is determined. Figure 1 contains the special case where the area of the eastern and the western sub-workspaces are maximized. The home position must be located on the circular trajectories drawn, or above both of them. Hence the set of valid home positions is equivalent to the northern sub-workspace in Fig. 1.

[2] In this case the whole upper edge of the workspace is valid.

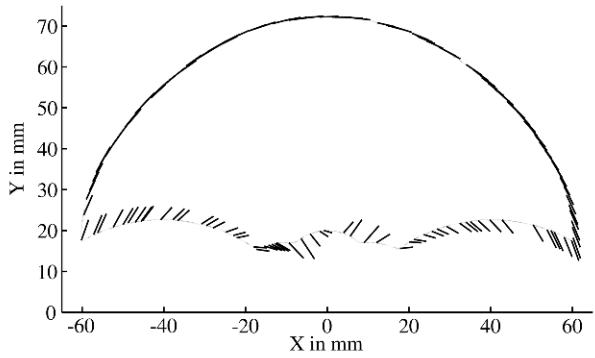


(a) Homing Vector $\boldsymbol{H}_1 = (1,1)^T$



(b) Homing Vector $\boldsymbol{H}_2 = (-1,1)^T$ (symmetric)

(c) Homing Vector $\boldsymbol{H}_3 = (1,-1)^T$ (symmetric)



(d) Homing Vector $\boldsymbol{H}_4 = (-1,-1)^T$

Fig. 2 Velocity vectors corresponding to homing vectors.

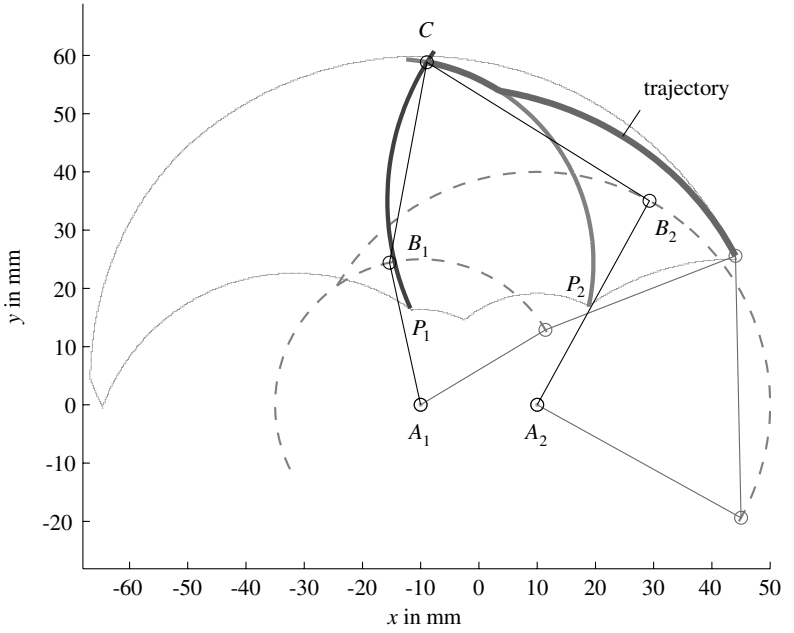


Fig. 3 A non-symmetric five-bar, the borders of the four sub-workspaces and an exemplary homing trajectory.

4 Generalization for the Non-Symmetric Five-Bar

Now the results concerning the symmetrical five-bar shall be generalized to arbitrary bar lengths. Just as with the symmetric case the northern sub-workspace must be found for the non-symmetric case. This area can be obtained geometrically, as shown in Fig. 3. The following steps yield the solution and have to be repeated for both, the left and right arm set (index $j = 1;2$): Bring C to point P_j, representing the inner end of the outermost bow segment. This defines the position of joint B_j. Draw a cirular sub-workspace border around the joint B_j, which is fixed. The set of valid home positions is equivalent to the northern sub-workspace.

5 Conclusions

This article is concerned with finding a valid set of home positions for a five-bar manipulator which is equipped with incremental encoders and homing sensors. It is proposed to employ a joint-based search strategy which is already well established in linear and rotary drive technology in non-coupled kinematics. The homing

strategy proposed relies on homing switches on each drive which determine the direction in which the home position must be sought.

The conditions that ensure that a home position can be found safely from every point within the workspace were presented. It was found that the velocity vectors must not point outside the workspace to ensure successful homing. Consequently the velocity vectors of an exemplary structure were analyzed. Therefrom a geometrical method for the determination of the set of valid home positions was derived. This method was generalized for non-symmetric geometries of the five-bar manipulator.

In order to avoid the need for assistance by an operator during the power-up phase of the machine it is common to equip a five-bar either with absolute encoders or with multiple homing switches. If not, a manual homing procedure must be carried out by a trained operator. With the knowledge obtained in this article such a manipulator may be designed in such a way that autonomous homing becomes possible even though the mechanical and electrical design is kept simple. Only very simple sensors are required which may be realized and integrated cheaply. On top of that there are many off-the-shelf servo drive amplifiers which feature the joint-based homing functionality proposed. With these products its implementation becomes straightforward. In summary this article contributes to the simplification of the design and the operation of planar five-bar manipulators.

References

1. Merlet, J.-P.: *Parallel Robots*, Solid Mechanics and Its Applications, Vol. 74. Kluwer Academic Publishers, Dordrecht (2001).
2. Hesselbach, J., Maaß, J. and Bier, C.: Singularity prediction for parallel robots for improvement of sensor-integrated assembly. *CIRP Annals-Manufacturing Technology*, **54**(1), 349–352 (2005).
3. Burisch, A., Degen, R., Hesselbach, J. and Raatz, A.: Assembly of micro systems with the high pecision robot PARVUS. In: *SmartSystems Integration 2007*. VDE Verlag GmbH, Berlin Offenbach, pp. 443–445 (2007).
4. Kolbus, M., Wobbe, F., Algermissen, S., Stachera, K., Schumacher, W. and Sinapius, M.: Sliding mode control of a parallel robot with robust vibration control. In: *Proc. of the 9th International Conference on Motion and Vibration Control (MoViC 2008)* (2008).
5. Burisch, A., Wrege, J., Raatz, A. and Hesselbach, J.: PARVUS – Miniaturised robot for improved flexibility in micro production. *Journal of Assembly Automation*, **27**(1), 65–73 (2007).
6. Belda, K and Pisa, P: Homing, calibration and model-based predictive control for planar parallel robots. In: *Proceedings of the UKACC International Conference on Control 2008* (2008).
7. Wang, J., Wu, J., Wang, L. and Li, T.: Homing strategy for a redundantly actuated parallel kinematic machine. *Journal of Mechanical Design*, **130**(4), 044501/1–5 (2008).
8. Tsai, L.-W.: *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*, Wiley-Interscience, New York (1999).

Singularities of Regional Manipulators Revisited

P. Donelan[1] and A. Müller[2]

[1] *Victoria University of Wellington, New Zealand; e-mail: peter.donelan@vuw.ac.nz*
[2] *Universität Duisburg-Essen, Germany; e-mail: andreas-mueller@uni-due.de*

Abstract. The workspace singularities of 3R regional manipulators have been much analyzed. The presence of cusps in the singularity locus is known to admit singularity-avoiding posture change. Cusps arise in singularity theory as second-order phenomena – specifically they are $\Sigma^{1,1}$ Thom–Boardman singularities. The occurrence of such singularities requires that the kinematic mapping be generic (in the sense of Pai and Leu [1]). Genericity and the occurrence of higher-order singularities in families of regional manipulators are investigated using Lie-theoretic properties of the Euclidean group.

Key words: Regional manipulator, cusp, Thom–Boardman singularity, genericity.

1 Introduction

A spatial serial 3-link manipulator is frequently termed a *regional manipulator* in recognition of its use as the positioning component for the wrist-centre of a wrist-partitioned 6-dof industrial manipulator. An example is illustrated in Fig. 1. The importance of such a design is that the inverse kinematics reduces to solving a degree 4 polynomial [2]. Hsu and Kohli [3] used this to show that there are, for a typical regional manipulator, surfaces in the joint space that divide it into regions corresponding to different numbers of poses. These regions were also studied, in a more general setting, by Burdick [4]. Taking a different perspective, Stanišić and Engelberth [5] demonstrated, using screw systems, that the wrist-positioning subassembly gives rise to a singularity of the whole manipulator when the wrist centre lies on a certain surface, dependent on the subassembly configuration. This was referred to as an *instantaneous singular set* in [6] and was subsequently used as the basis for a singularity metric [7].

A number of researchers have used ideas from the mathematical theory of singularities in the study of manipulators [1, 8, 9]. Pai and Leu examined the stratification of the singularity locus by the *corank* of the singularity, that is, the instantaneous loss of degrees of freedom (dofs). In particular, they introduced the concept of *generic* manipulator, to describe one whose kinematic mapping has a nice singularity

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_55,

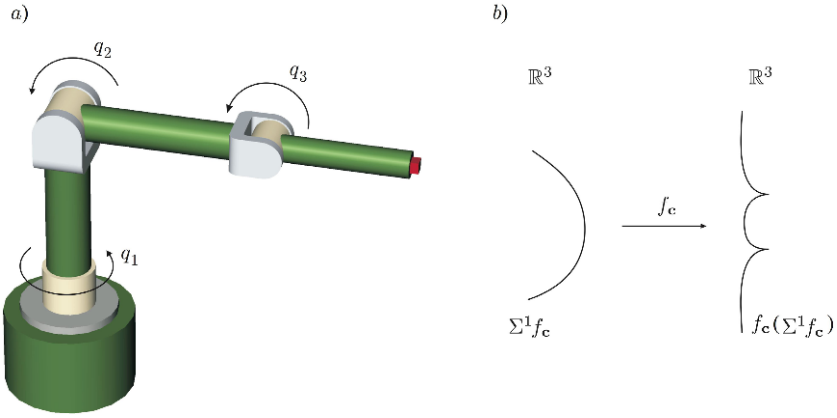


Fig. 1 (a) Ortho-parallel regional manipulator [10], (b) Visualisation of cusp singularities.

locus. In singularity theory, the term 'generic' is used to describe a property that pertains for a topologically large set (for example, open and dense, residual, or having complement of measure zero) in a given parametrised family of mappings. In this setting, the family could be the set of 3R manipulators, or it could be the set of wrist-centres for a given manipulator, or the entire family with both the serial structure and wrist-centre as parameters. Generic properties are typically realised via *transversality* to a given family of manifolds – an intersection condition satisfied when certain vectors span a given space. In this case, the given family of manifolds can be interpreted as the sets Σ^r of Jacobian matrices of fixed corank r.

In much of the literature on this subject, the term 'generic' is used to describe a manipulator for which the transversality condition holds rather than the property itself. A more accurate term is *transverse-regular* [11]. There is, as pointed out by Tchoń [8], no certainty that the transversality condition will indeed hold for most manipulators in a given class. Regardless of whether the condition holds for most manipulators, whenever it does hold it guarantees that the singular loci $\Sigma^r f$, corresponding to singularities of fixed rank, are themselves manifolds in the joint space of the kinematic mapping f. On the other hand, the singular image in the workspace can exhibit singularities, such as cusps. For further details of genericity and concepts of singularity theory, see [11–15].

In the context of regional manipulators, algebraic conditions for genericity (i.e. transversality) were obtained in [1, 9]. In particular, they showed that it is not possible to encounter Σ^2 transversely, so that rank 1 singularities are ruled out. Burdick [16] gave an alternative geometric criterion to the algebraic equations of [3]: when a 3R regional manipulator is in a singular configuration there exists a screw of pitch zero whose axis passes through the wrist centre and intersects the axes of each joint screw. He observed that for an open set of 3R regional manipulators, there exist trajectories in joint space that do not intersect the singular locus, yet effect a change of posture. Such manipulators have been termed *cuspidal* and

have been explored in detail by Wenger et al. [17–19]. Smith and Lipkin [20, 21] showed that the inverse kinematics of a given wrist-centre for a 3R regional manipulator can be encoded by a pencil of conics. Exceptional pencils in which the conics possess some tangency correspond to singular configurations, while high-order tangency (3rd or 4th order or paired) correspond to higher-order singularities, including cusps. Recent classifications focus on specific classes of manipulator, for example orthogonal [22], where a closed form expression for the Jacobian can be found, and on workspace topologies [10].

Selig [23] analysed the kinematics and singularities of 3R manipulators using product-of-exponentials formulation for the kinematics and results of Lie theory. It is this approach that we pursue. A cusp arises as a singular point of the restriction of the kinematic mapping to the singular locus: in the notation of Thom–Boardman singularities [24] it is $\Sigma^{1,1}$. Our aim is to develop the singularity analysis of regional manipulators in a reasonably broad context, provisionally allowing for 1-dof joints of any sort, and deriving local descriptions of singular loci using methods of Lie groups and Lie algebras. In this setting the two different aspects of the singularity problem for regional manipulators – choice of the underlying serial manipulator structure and of the wrist centre – can be developed together.

2 The Kinematic Mapping

The motion associated with each 1-dof joint of a manipulator can be represented by a non-zero twist X – an element of the Lie algebra $\mathfrak{se}(3)$ of the group $SE(3)$ of Euclidean isometries. The motion itself is given by the exponential $\exp(qX)$, a path in the group of transformations, where q is the joint variable. The twist is relative to a given choice of coordinates in the link and the ambient space; under a change of coordinates represented by an isometry $g \in SE(3)$, the twist transforms by conjugacy and this is the *adjoint* action of the Lie group on its Lie algebra:

$$X \mapsto \mathrm{Ad}\,(g)(X) = gXg^{-1}, \tag{1}$$

where the elements of the group and twists can be written in matrix form. In a given coordinate frame, the twist may be replaced by any non-zero multiple, a twist of the same pitch, the joint variable being scaled by the inverse of the multiple. In other words the joint is really represented by a *screw*. The kinematic mapping of a serial manipulator with k 1-dof joints can then be written as a product of exponentials

$$f(q_1, \ldots, q_k) = \exp(q_1 X_1) \cdots \exp(q_k X_k), \tag{2}$$

where X_i, $i = 1, \ldots, k$, is the twist representing the ith joint, in a chosen home configuration with respect to given space (or base) coordinates, and $q_i \in \mathbb{R}$ is the joint variable. Again, exponentials can be evaluated as matrices via the standard series formula. The image of each exponential map is the one-parameter subgroup of Euclidean transformations in $SE(3)$, parametrised by the joint variable. The twists

X_i can be equivalently represented by either: a 6-vector $(\boldsymbol{\omega}_i, \mathbf{v}_i)$ comprised of two 3-vectors corresponding to infinitesimal rotation and translation, or a 4×4 matrix partitioned as

$$\left(\begin{array}{c|c} \widetilde{\omega} & \mathbf{v} \\ \hline \mathbf{0}^t & 0 \end{array}\right), \tag{3}$$

where the identification of $\boldsymbol{\omega} \in \mathbb{R}^3$ and the 3 skew-symmetric matrix $\widetilde{\omega}$ proceeds with

$$\boldsymbol{\omega} = \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} \leftrightarrow \widetilde{\omega} = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix}. \tag{4}$$

If the joint X_i is revolute then $\boldsymbol{\omega}_i \cdot \mathbf{v}_i = 0$, while if it is prismatic then $\boldsymbol{\omega}_i = \mathbf{0}$. A priori, there is no need to assume that the joints are either of these types, that is, they may have pitch $\boldsymbol{\omega}_i \cdot \mathbf{v}_i / \boldsymbol{\omega}_i \cdot \boldsymbol{\omega}_i \neq 0, \infty$. While the exponential map is defined independent of the representation used for the Lie algebra, in the matrix form it can be computed by the usual exponential series.

For a regional manipulator $k = 3$, and there is a choice of wrist-centre $\mathbf{c} \in \mathbb{R}^3$ (in body coordinates for the third link). The kinematic mapping for the wrist centre is the function

$$f_{\mathbf{c}} : \mathbb{R}^3 \to \mathbb{R}^3; \quad f_{\mathbf{c}}(q_1, q_2, q_3) = \exp(q_1 X_1) \exp(q_2 X_2) \exp(q_3 X_3) \cdot \mathbf{c}. \tag{5}$$

The 'evaluation' map $\epsilon_{\mathbf{c}} : SE(3) \to \mathbb{R}^3$ is given by the action of the group on the wrist-centre $\mathbf{c}$, that is for $g \in SE(3)$, $\epsilon_{\mathbf{c}}(g) = g \cdot \mathbf{c}$. Then $f_{\mathbf{c}}$ is the composition of $\epsilon_{\mathbf{c}}$ with the manipulator kinematic mapping f in (2).

3 Jacobian Matrices

A kinematic mapping f has a singularity at $\mathbf{q}$ when the rank of its derivative $Df(\mathbf{q})$ drops below its maximum possible value, which is the minimum of the dimensions of the joint-space and the configuration space. The derivative is represented by the (analytic) Jacobian matrix of partial derivatives. This represents a linear mapping into the tangent space at the image $f(\mathbf{q}) \in SE(3)$ rather than into the Lie algebra, so that the columns are not themselves, in general, twists. However the group structure can be used to 'pull back' the tangent space to the identity by either left multiplication (corresponding to body coordinates) or right translation (space coordinates) to give a more familiar geometric Jacobian. For the manipulator mapping f in (2) with $k = 3$ the matrix is therefore 6×3. To find an explicit form requires the derivative of an exponential:

$$\frac{d}{dq} \exp(qX) = X \cdot \exp(qX) = \exp(qX) \cdot X, \tag{6}$$

where the operations between the transformation $\exp(qX)$ and the twist X can be realised by matrix multiplication. Following [23], if $g \in SE(3)$ can be written as $g = (R, \mathbf{t}) \in SO(3) \times_s \mathbb{R}^3$ (where $\times_s$ denotes semi-direct product), then g is represented by the 6×6 partitioned matrix

$$\left(\begin{array}{c|c} R & O \\ \hline \widetilde{t}R & R \end{array}\right) \tag{7}$$

with the skew-symmetric matrix $\widetilde{t}$ defined in (4). Note that the exponential mapping commutes with its defining twist. However, it does not commute with a general twist and we require:

$$Y \exp(qX) = \exp(qX) \exp(-qX) Y \exp(qX) = \exp(qX) \mathrm{Ad}\,(\exp(-qX))(Y). \tag{8}$$

Differentiating Ad as in (1) with respect to $g \in SE(3)$ gives the adjoint representation of the Lie algebra $\mathfrak{se}(3)$ on itself. This also determines the *Lie bracket* operation in the Lie algebra:

$$\mathrm{ad}\,(Y)(X) = [Y, X] \tag{9}$$

In matrix terms $[Y, X] = YX - XY$, while in screw coordinates

$$[(\boldsymbol{\omega}_1, \mathbf{v}_1), (\boldsymbol{\omega}_2, \mathbf{v}_2)] = (\boldsymbol{\omega}_1 \times \boldsymbol{\omega}_2, \boldsymbol{\omega}_1 \times \mathbf{v}_2 + \mathbf{v}_1 \times \boldsymbol{\omega}_2)\,. \tag{10}$$

It is a theorem of matrix Lie groups that

$$\mathrm{Ad}\,(\exp(qX)) = \mathrm{Exp}(q\ \mathrm{ad}\,(X)) = \sum_{n=0}^{\infty} \frac{q^n}{n!} (\mathrm{ad}\,X)^n, \tag{11}$$

where the exponential Exp is an operator on the Lie algebra. We obtain the analytic Jacobian of f as follows (where vertical dots separate column vectors):

$$Jf(\mathbf{q}) = \Big(\exp(q_1X_1)X_1 \exp(q_2X_2) \exp(q_3X_3) \,\vdots\, \exp(q_1X_1) \exp(q_2X_2)X_2 \exp(q_3X_3) \,\vdots\, \exp(q_1X_1) \exp(q_2X_2) \exp(q_3X_3)X_3\Big) \tag{12a}$$

$$= \Big(f(q_1, q_2, q_3) \cdot \mathrm{Exp}(-q_3\ \mathrm{ad}\,X_3)\mathrm{Exp}(-q_2\ \mathrm{ad}\,X_2)X_1 \,\vdots\, f(q_1, q_2, q_3) \cdot \mathrm{Exp}(-q_3\ \mathrm{ad}\,X_3)X_2 \,\vdots\, f(q_1, q_2, q_3) \cdot X_3\Big)\,. \tag{12b}$$

The second expression is obtained by applying (8) and the effect of $f(q_1, q_2, q_3)$ on each term is to translate the twists in the tangent space at the identity (the Lie algebra $se(3)$) to the tangent space at the given configuration. The corresponding geometric Jacobian, consisting of the instantaneous joint screws in end-effector coordinates, is therefore:

$$J_{\text{geom}} = \left(X_1' \,\vdots\, X_2' \,\vdots\, X_3' \right) \tag{13}$$

where $X_1' := \text{Exp}(-q_3 \,\text{ad}\, X_3)\text{Exp}(-q_2 \,\text{ad}\, X_2)X_1$, $X_2' := \text{Exp}(-q_3 \,\text{ad}\, X_3)X_2$, $X_3' := X_3$.

An important object for establishing transversality of the kinematic mapping at a given configuration is the Lie subalgebra generated by the joint screws X_i since it contains the the subspace spanned by X_i at any configuration. Assuming that we are interested in the configuration $\mathbf{q} = \mathbf{0}$ then (12b) reduces to (13). Expanding (13) as a series in q_1, q_2, q_3 by means of (11) gives:

$$\begin{aligned} X_1' &= X_1 + q_2[X_1, X_2] + q_3[X_1, X_3] + O(2) \\ &= (\boldsymbol{\omega}_1, \mathbf{v}_1) + q_2(\boldsymbol{\omega}_1 \times \boldsymbol{\omega}_2, \mathbf{r}_{12}) + q_3(\boldsymbol{\omega}_1 \times \boldsymbol{\omega}_3, \mathbf{r}_{13}) + O(2) \qquad (14a) \\ X_2' &= X_2 + q_3[X_2, X_3] + O(2) \\ &= (\boldsymbol{\omega}_2, \mathbf{v}_2) + q_3(\boldsymbol{\omega}_2 \times \boldsymbol{\omega}_3, \mathbf{r}_{23}) \qquad (14b) \\ X_3' &= X_3 = (\boldsymbol{\omega}_3, \mathbf{v}_3), \qquad (14c) \end{aligned}$$

where $\mathbf{r}_{ij} = \boldsymbol{\omega}_i \times \mathbf{v}_j + \mathbf{v}_i \times \boldsymbol{\omega}_j$.

4 The Singular Locus

For the wrist-centre kinematic mapping $f_{\mathbf{c}}$ in (5), the Jacobian is 3×3. Its columns are the result of applying the columns of (12b), considered as elements of $SE(3)$, to $\mathbf{c}$. However, recalling that $f_{\mathbf{c}} = \epsilon_{\mathbf{c}} \circ f = f \cdot \mathbf{c}$, and applying the chain rule to get $Df_{\mathbf{c}} = D\epsilon_{\mathbf{c}} \circ Df$, it is clear that the rank of the derivative of $f_{\mathbf{c}}$ is less than 3, i.e. $f_{\mathbf{c}}$ has a singularity, if and only if one of the following occurs [6]:

i. f itself has a singularity;
ii. the kernel of the derivative of the evaluation map $\epsilon_{\mathbf{c}}$ has non-trivial intersection with the image of the derivative of f.

Case (i) corresponds to Burdick's 'extra branch singularities' [16]. Here, the defining screws X_1, X_2, X_3 are linearly dependent in the Lie algebra. We assume none are zero and no two adjacent joint screws are permanently linearly dependent (since then the manipulator effectively only has 2 dof). In particular the screws must span a 2-dimensional subspace so correspond to a Σ^1 singularity. It follows from Theorem 3.1 in [13] that the singularity occurs transversely so long as the the subspace is not a subalgebra (i.e. closed under the Lie bracket), in which case that manipulator would have 2 dof only. The 2-dimensional subalgebras are the algebra of cylindrical motion and pure 2 dof translations.

In case (ii), the kernel of the derivative of $\epsilon_{\mathbf{c}}$ at the identity is precisely the set of pitch-zero twists whose axes pass through $\mathbf{c}$: in the terminology of the Klein quadric, this is an α-plane. This provides the principle for determining the instantaneous singular sets in terms of the screw system in [5, 6]: a singularity can be detected

when the determinant of the 6×6 matrix, whose columns are 3 twists spanning the α-plane and 3 from the Jacobian of f, vanishes.

This can be exploited by choosing coordinates in the end-effector so that $\mathbf{c}$ is the origin. Then the twists spanning the α-plane can be taken as infinitesimal rotations about the coordinate axes and the Jacobian has the partitioned form

$$\left(\begin{array}{c|c} I & J_{\boldsymbol{\omega}} f(\mathbf{q}) \\ \hline O & J_{\mathbf{v}} f(\mathbf{q}) \end{array}\right), \tag{15}$$

where $J_{\boldsymbol{\omega}} f(\mathbf{q})$, $J_{\mathbf{v}} f(\mathbf{q})$ denote the projections of the Jacobian $Jf(\mathbf{q})$ onto the subspaces of infinitesimal rotations about the origin and infinitesimal translations, respectively. The determinant is simply equal to that of the lower right 3×3 block coming from the 'translational' part of f. Expanding this as a triple scalar product of its columns and using (14) gives the following expression for $\det J_{\mathbf{v}} f(\mathbf{q})$:

$$h(\mathbf{q}) := \mathbf{v}_1 \cdot (\mathbf{v}_2 \times \mathbf{v}_3) + q_2 \mathbf{v}_2 \cdot (\mathbf{v}_3 \times \mathbf{r}_{12}) + q_3 \mathbf{v}_3 \cdot (\mathbf{v}_1 \times \mathbf{r}_{23} + \mathbf{v}_2 \times \mathbf{r}_{13}) + O(2) \tag{16}$$

The condition $\mathbf{v}_1 \cdot (\mathbf{v}_2 \times \mathbf{v}_3) = \mathbf{0}$ (equivalently, $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ linearly dependent) affirms that $\mathbf{q} = \mathbf{0}$ itself is a singular point of $f_{\mathbf{c}}$. Indeed, if $\mathbf{v}$ lie in the orthogonal complement to the subspace spanned by $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ then the twist $(\mathbf{v}, \mathbf{0})$ passes through the wrist centre and is clearly reciprocal to each joint. In particular, if the joints are revolute then the line intersects their axes [25], giving Burdick's geometric criterion mentioned in Section 1.

The form (16) gives an equation $h(\mathbf{q}) = 0$ for the singularity locus in a neighbourhood of $\mathbf{q} = \mathbf{0}$. By the Implicit Function Theorem, if either of the coefficients of the q_i, $i = 2, 3$ in (16) is non-zero, then the singular locus is a 2-dimensional submanifold of the joint space in a neighbourhood of $\mathbf{0}$. Since each coefficient is a polynomial in the screw coordinates of the three joints, their zero sets are closed subspaces (affine varieties) and so there is an open set of 3 dof (not necessarily 3R) manipulators for which a given wrist centre has a smooth singular locus. In other words, this family is generic with respect to transverse regularity.

5 Cusps

The defining equation for the singular locus enables us to deduce a criterion for the wrist centre to be a cusp point in the case that the locus is a manifold. In the notation of the Thom–Boardman singularities [24, 26] a cusp point is a point of $\Sigma^{1,1} f_{\mathbf{c}}$; that is, a point in the joint space at which the restriction of the kinematic mapping $f_{\mathbf{c}}$ to its singular locus itself has a corank 1 singularity. Computationally, this is similar to a Lagrange multiplier problem – find the singular points of a function constrained to a given submanifold. In Section 4 we have not produced a closed form expression for the singular locus – it is only known as a series expansion up to first-order, though in

principle more terms can be calculated and indeed there are closed form expressions for $\exp(q \text{ ad } X)$ [27]. However this is sufficient for the criterion we seek.

The required condition is that the derivative (or Jacobian matrix) for the augmented map $(f_{\mathbf{c}}, h) : \mathbb{R}^3 \to \mathbb{R}^4$ should have corank 1, in addition to the underlying requirement that $Df_{\mathbf{c}}(\mathbf{q})$ itself has corank 1. Since we have only expanded the singular locus about $\mathbf{q} = \mathbf{0}$, it is only possible to apply this criterion at that point, where we have enough information to determine the Jacobian of $(f_{\mathbf{c}}, h)$. The 4×3 matrix arising from differentiating (5) and (16) is:

$$\begin{pmatrix} v_{11} & v_{21} & v_{31} \\ v_{12} & v_{22} & v_{32} \\ v_{13} & v_{23} & v_{33} \\ 0 & \mathbf{v}_2 \cdot (\mathbf{v}_3 \times \mathbf{r}_{12}) & \mathbf{v}_3 \cdot (\mathbf{v}_1 \times \mathbf{r}_{23} + \mathbf{v}_2 \times \mathbf{r}_{31}) \end{pmatrix} \tag{17}$$

where $\mathbf{v}_i = (v_{i1}, v_{i2}, v_{i3})^T$ for $i = 1, 2, 3$, and a sufficient condition for this to have rank 2 is that all 3×3 submatrices have determinant zero. Taking the first three rows automatically ensures that $\mathbf{q} = \mathbf{0} \in \Sigma^1 f_{\mathbf{c}}$. While there are three further submatrices, only one condition is algebraically independent: if the kinematic mapping is well behaved (satisfies an appropriate transversality condition) then $\Sigma^{1,1} f_{\mathbf{c}}$ will be a 1-dimensional submanifold (curve) in joint space.

6 Example: Ortho-Parallel Manipulator

For an ortho-parallel manipulator, take as home configuration the one shown in Fig. 1, where the parallel joints and the wrist centre all lie in a plane orthogonal to the axis of the first joint. Then with the wrist-centre as origin of coordinates and suitable choice of axes, the kinematics (5) can be defined using the following screw coordinates $(\boldsymbol{\omega}_i, \mathbf{v}_i)$, $i = 1, 2, 3$ for the joints:

$$\begin{aligned} X_1 &= (0, 0, 1, -(a_1 + a_2 + a_3), d_2 + d_3, 0) \\ X_2 &= (1, 0, 0, 0, 0, a_2 + a_3) \\ X_3 &= (1, 0, 0, 0, 0, a_3), \end{aligned} \tag{18}$$

where the a_i, d_i are DH parameters (see, for example, [10]). In the home configuration the joints are linearly independent. However $\mathbf{v}_1 \cdot (\mathbf{v}_2 \times \mathbf{v}_3) = 0$ so this is a singular configuration. Locally, the singular locus is defined by

$$h(\mathbf{q}) = -a_2 a_3 (a_1 + a_2 + a_3) q_3 + O(2) \tag{19}$$

Thus, the singular locus is locally a surface for non-trivial DH parameters unless $a_1 + a_2 + a_3 = 0$. The condition for a cusp is that the matrix

$$\begin{pmatrix} -(a_1+a_2+a_3) & 0 & 0 \\ d_2+d_3 & 0 & 0 \\ 0 & a_2+a_3 & a_3 \\ 0 & 0 & -a_2a_3(a_1+a_2+a_3) \end{pmatrix} \tag{20}$$

has rank 2, which is clearly not the case provided the conditions above hold. Notice that for the manipulator in Fig. 1 we have $a_1 = d_2 = 0$. While it is of limited practical value, such a local singularity analysis remains straightforward for general screw joints of non-zero pitch.

It is also possible to analyse the effect of varying the wrist centre to a point $\mathbf{c} = (c_1, c_2, c_3)^T$. One way to do this is to transform coordinates by means of a translation so that the wrist centre remains the origin. The joint twist $X_i = (\boldsymbol{\omega}_i, \mathbf{v}_i)$ transforms to $X_i = (\boldsymbol{\omega}_i, -\boldsymbol{\omega}_i \times_i \mathbf{c} + \mathbf{v}_i)$. The linear part of the equation for the singularity locus of an ortho-parallel manipulator becomes:

$$\begin{aligned}(a_1+a_2+a_3-c_2)c_3a_2 + c_3^2a_2q_2 \\ - \Big[(a_2+a_3-c_2)c_3^2 - (a_3-c_2)((a_1+a_2+a_3-c_2)a_2+c_3^2)\Big]q_3.\end{aligned} \tag{21}$$

In particular, a shift of wrist centre parallel to the base axis (c_3-direction) moves the wrist centre off the singular locus.

7 Conclusions

Traditionally kinematic singularities of robotic manipulators have mostly been studied as a first order phenomenon in the sense of the Thom–Boardman singularity theory. It was realized that a 3R regional manipulator can change its posture without meeting a singularity if it exhibits cusp singularities (a second-order phenomena). Therefore, and because the 3 dof manipulator kinematics is accessible to symbolic manipulations, cusp singularities of 3R regional manipulators have been the subject of extensive studies. Today much is known in the most important cases employed in industrial manipulators, in particular for orthogonal and, to some extent, for ortho-parallel manipulators. The significance of higher order singularities of general manipulators remains an open question. Here we have revisited the general problem for regional manipulators in terms of the manipulator screw system using the adjoint action of the Euclidean group and the Thom–Boardman singularity theory. This provides the basis for exploring genericity conditions of regional manipulators without resorting to DH parameters. With the approach taken here it shall be possible to derive geometrically interpretable second order genericity conditions for regional manipulators with general screw joints (ensuring that $\Sigma^{1,1}$ is a manifold). Eventually we shall be able to determine second order genericity of general manipulators.

References

1. Pai, D. K. and Leu, M. C.: Genericity and singularities of robot manipulators, *IEEE Trans. Robotics and Automation*, **8**, 545–559 (1992).
2. Pieper, D. and Roth, B.: The kinematics of manipulators under computer control. In: *Proc. 2nd World Congress on the Theory of Machines and Mechanisms*, Zakopane, Poland, Vol. 2, pp. 159–169 (1969).
3. Hsu, M.-S. and Kohli, D.: Boundary surfaces and accessibility regions for regional structures of manipulators, *Mech. Machine Theory*, **22**, 277–289 (1987).
4. Burdick, J. W.: PhD thesis, Department of Mechanical Engineering, Stanford University (1988).
5. Stanišić, M. M. and Engelberth, J. W.: A Cartesian description of arm-subassembly singularities in terms of singular surfaces. In: *Proc. 20th ASME Biennial Mechanisms Conf.*, Orlando FL, ASME, pp. 325–332 (1988).
6. Cocke, M. W., Donelan, P. S. and Gibson, C. G.: Instantaneous singular sets associated to spatial motions. In: Tari, F. and Bruce, J. W. (Eds.), *Real and Complex Singularities*, São Carlos. Chapman and Hall/CRC Press, Boca Raton, pp. 147–163 (2000).
7. Stanišić, M. M. and Goehler, C. M.: Singular planes of serial wrist-partitioned manipulators and their singularity metrics, *Mech. Machine Theory*, **42** 889–902 (2007).
8. Tchoń, K.: Differential topology of the inverse kinematic problem for redundant robot manipulators, *Int. J. Robotics Research*, **10**, 492–504 (1991).
9. Tsai, K. Y., Arnold, J. and Kohli, D.: Generic maps of mechanical manipulators, *Mech. Machine Theory*, **28**, 53–64 (1993).
10. Ottaviano, E. and Ceccarelli, M. and Husty, M.: Workspace topologies of industrial 3R manipulators, *Int. J. Advanced Robotic Systems*, **4**, 355–364 (2008).
11. Müller, A.: Conditions for transverse-regularity of serial manipulators. In: *Proc. 2007 Mediterranean Conference on Control and Automation*, Athens, Greece. IEEE, Piscataway, NY, pp. 1–6 (2007).
12. Donelan, P. S.: Singularity-theoretic methods in robot kinematics, *Robotica*, **25**, 641–659 (2007).
13. Donelan, P.: Genericity conditions for serial manipulators. In: Lenarčič, J. and Wenger, P. (Eds.), *Advances in Robot Kinematics: Mechanisms and Motion*. Springer, Dordrecht, pp. 185–192 (2008).
14. Müller, A.: A genericity condition for general serial manipulators. In: *Proc. 2009 IEEE International Conference on Robotics and Automation*, Kobe, Japan. IEEE, Piscataway, NY, pp. 2951–2956 (2009).
15. Müller, A.: Generic mobility of rigid body mechanisms, *Mech. Machine Theory*, **44**, 1240–1255 (2009).
16. Burdick, J. W.: A classification of 3R regional manipulator singularities and geometries, *Mech. Machine Theory*, **30**, 71–89 (1995).
17. Wenger, P. and El Omri, J.: Changing posture for cuspidal robot manipulators. In: *Proc. 1996 IEEE International Conference on Robotics and Automation*, Minneapolis MN. IEEE, Piscataway, NY, pp. 3173–3178 (1993).
18. Wenger, P.: Classification of 3R positioning manipulators, *ASME J. Mechanical Design*, **120**, 327–332 (1998).
19. Wenger, P.: Cuspidal and noncuspidal robot manipulators, *Robotica*, **25**, 677–689 (2007).
20. Smith, D. R. and Lipkin, H.: Kinematic analysis of solvable manipulators using conic sections. In: McCarthy, J. M. et al. (Eds.), *Proc. 21st ASME Mechanisms Conference*, Chicago IL. IEEE, Piscataway, NY, pp. 16–19 (1990).
21. Smith, D. R. and Lipkin, H.: Higher order singularities of regional manipulators. In: *Proc. 1993 IEEE International Conference on Robotics and Automation*, Atlanta GA. IEEE, Piscataway, NY, pp. 194–199 (1993).
22. Zein, M., Wenger, P. and Chablat, D.: An exhaustive study of the workspace topologies of all 3R orthogonal manipulators with geometric simplifications, *Mech. Machine Theory*, **41**, 971–986 (2006).

23. Selig, J.: *Geometrical Fundamentals of Robotics*. Springer Verlag, New York (2005).
24. Boardman, J. M.: Singularities of differentiable maps, *Publ. Math. I.H.E.S.*, **33**, 21–57 (1967).
25. Gibson, C. G. and Hunt, K. H.: Geometry of screw systems I & II, *Mech. Machine Theory*, **25**, 1–27 (1990).
26. Golubitsky, M. and Guillemin, V.: *Stable Mappings and Their Singularities*, Springer Verlag, New York (1973).
27. Selig, J.: Lie groups and Lie algebras in robotics. In: Byrnes, J. and Ostheimer, G. (Eds.), *Computational Noncommutative Algebra and Applications*, Il Ciocco, Italy. Kluwer, Dordrecht, pp. 101–125 (2004).

Robot-Based HiL Test of Joint Endoprostheses

C. Woernle[1], M. Kähler[1], R. Rachholz[1], S. Herrmann[2], J. Zierath[1],
R. Souffrant[2] and R. Bader[2]

[*]*Department of Mechanical Engineering and Marine Technology,*
University of Rostock, Germany;
e-mail: {woernle, michael.kaehler, roman.rachholz, janos.zierath}@uni-rostock.de
[2]*Department of Orthopedics, University of Rostock, Germany;*
e-mail: {rainer.bader, sven.herrmann, robert.souffrant}@med.uni-rostock.de

Abstract. To simulate the dislocation behavior of total hip endoprostheses in their anatomical environment a novel Hardware-in-the-Loop (HiL) simulator is built up. It couples a real endoprosthesis with a numerical simulation of its environment by means of an industrial robot as actuator system. The simulation model describes the dynamics of the biomechanical motions including the tissue and muscle forces. The motion and joint constraint forces are calculated by the simulation model and applied to the endoprosthesis by the robot under hybrid position/force control. The actual position of the endoprosthesis in the constrained directions and torques in the unconstrained directions are measured and fed back into the simulation model closing the control loop. To demonstrate the functional principle of the HiL simulator the dynamic behavior of a test setup is numerically simulated.

Key words: Hardware-in-the-loop, position/force control, endoprosthesis, biomechanics.

1 Introduction

A relatively frequent and serious complication of total hip replacement is subluxation and dislocation of the artificial joint (Fig. 1) [1]. Typical mechanisms of dislocation are so-called impingements between prosthetic components or surrounding bone structures, but dislocation may even occur under dynamic forces in conjunction with damaged muscles without an impingement event. Despite increasing clinical experiences the dislocation rate has not been reduced significantly during the last years. There are still numerous open questions about influence parameters like implant design, implant position, actual loads as well as anatomic conditions like muscles, ligaments, and capsule structures.

Therefore the objective is to investigate and compare actual joint endoprostheses with respect to their dynamic behavior under realistic motion and load conditions. As in vivo measurements of failure scenarios are not possible, the anatomical environment of the endoprosthesis is transferred into a numerical simulation. For this purpose a novel robot-based Hardware-in-the-Loop (HiL) joint simulator is developed [2]. In a HiL simulation physical components of a system are tested with a numerical simulation of their environment running on a computer in real time [3]. The physical

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_56, © Springer Science+Business Media B.V. 2010

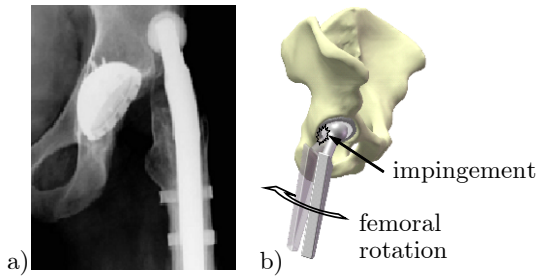


Fig. 1 (a) Dislocated hip endoprosthesis. (b) Prosthetic impingement possibly causing dislocation.

component and the numerical simulation process mutually interact in a closed control loop. HiL simulations are particularly advantageous if the real environment of the physical component is not accessible as in the case of joint endoprostheses. To include a mechanical component into a HiL control loop sensors and actuators are used for the interaction between the mechanical component and the simulation computer leading to a mechatronic HiL simulator. A six-axis industrial robot is used as actuator system in order to realize the large range of motion of hip and knee joints. By comparison a force-controlled hexapod platform with pneumatic actuators is under development for testing of cervical vertebrae pairs [4].

2 HiL Simulator for Testing of Endoprostheses

The HiL simulator for testing of joint endoprostheses is shown in Fig. 2. The hip endoprosthesis is moved and loaded by an industrial robot (STÄUBLI TX 200). The femoral component of the endoprosthesis is attached to the end-effector. The acetabular cup is implanted into an artificial pelvis structure. The pelvis support is elastically compliant in all three translational directions. The compliance is used for force control of the robot. The forces and torques in the endoprosthesis are measured by a 6D force-torque sensor (FTS).

The overall functional principle of the HiL simulator is shown in Fig. 3. The mechanical setup of Fig. 2 is linked with a simulation computer running a biomechanical multibody model of the anatomic environment of the hip joint. The simplified representations of the robot and the biomechanical multibody model in Fig. 3 show in-plane motions and forces only while spatial motions and forces are actually considered. Further the biomechanical model briefly addressed in Section 4 is more complex and has additional bodies.

The HiL concept is based on complementary sets of free and constrained spatial directions of the biomechanical multibody system. In the planar multibody model shown in Fig. 3 the free direction is the rotation of the femur with the angle q while the constrained directions are the displacements $\bar{q}_y^{\mathrm{r}}$, $\bar{q}_z^{\mathrm{r}}$ of the joint center $\bar{P}$. For an actual time instant t with given state variables q and $\dot{q}$ the equations of motion deliver the acceleration $\ddot{q}$ in the free direction and reaction forces f_y^{r}, f_z^{r} in the two constrained directions. The muscle and soft tissue forces f_{M} as well as gravity forces f_{G}

a) Overall setup b) Compliant support of the implant

Fig. 2 HiL simulator for testing of joint endoprostheses.

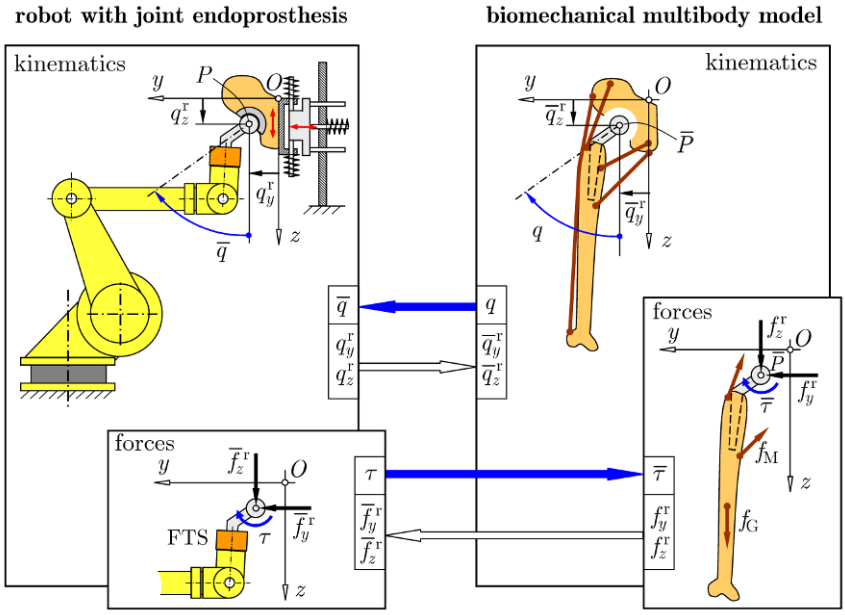


Fig. 3 Functional principle of the HiL joint simulator.

and inertial forces are taken into account. The acceleration $\ddot{q}$ is used for numerical integration of the equations of motion in order to update the state variables q, $\dot{q}$.

The values of the coordinate q and of the reaction forces f_y^{r}, f_z^{r} are transferred to the robot controller making the robot rotate the endoprosthesis into the position $\bar{q}$ and to apply the reaction forces $\bar{f}_y^{\mathrm{r}}$, $\bar{f}_z^{\mathrm{r}}$ onto the endoprosthesis. In the ideal case the robot values $\bar{q}$, $\bar{f}_y^{\mathrm{r}}$, $\bar{f}_z^{\mathrm{r}}$ are identical with the corresponding values q, f_y^{r}, f_z^{r} of the multibody model. In reality differences occur due to the limited dynamic bandwidth of the controlled robot, signal delay times, and other errors.

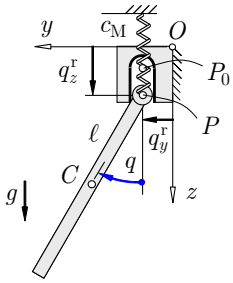


Fig. 4 Mechanical test system to illustrate the dynamics of the HiL simulator.

Force control is realized by means of the compliant support [5]. Basically the robot moves in direction of the desired force until the desired force value is achieved whereby the actual force value is measured by the FTS.

The robot is able to apply the reaction forces $\bar{f}_y^r$, $\bar{f}_z^r$ if the endoprosthesis withstands these loads in the corresponding directions. Then no dislocation will occur, i.e., the position of the joint center P relative to the acetabular cup, described by the distances q_y^r, q_z^r, remains constant. If the endoprosthesis does not withstand the applied forces in a certain direction, it dislocates in that direction, i.e., the distances q_y^r, q_z^r change.

To close the HiL control loop the actual values of q_y^r, q_z^r are measured and fed back into the multibody model. Another loop closure is achieved by the forces in the unconstrained directions, here the torque τ along the coordinate q, that are also measured and fed back into the multibody model. The torque τ may be caused by friction forces in the endoprosthesis, but also by prosthetic or bone impingements. Again disturbances of the measurements and the signal transmissions cause differences between the actual robot values τ, q_y^r, q_z^r and the corresponding values $\bar{\tau}$, $\bar{q}_y^r$, $\bar{q}_z^r$ fed back into the numerical model.

3 Simplified Mathematical Model of the HiL Simulator

The dynamic behavior of the HiL simulator is illustrated by means of a simple mechanical test system. It consists of a planar pendulum with its end point P moving in a vertical slot of a bearing block according to Fig. 4. As long as P is kept in its home position P_0 by a pre-loaded spring (stiffness c_M), the bearing is equivalent to a revolute joint in P_0. Due to the forces acting on the pendulum during its motion with the rotation angle $q(t)$, a displacement $\bar{q}_z^r$ of the end point P may occur that corresponds to the dislocation of the hip joint. Thus the spring force stands for the muscle and tissue forces holding the joint together against the other forces.

The HiL control loop with this test system is seen in Fig. 5. In the following a simplified mathematical model is established in order to illustrate the dynamic behavior of the HiL simulation and to demonstrate a dislocation scenario. For the actual time t with the position of point $\bar{P}$ the equations of motion of the pendulum are (mass center C, distance $\overline{C\bar{P}} = \ell$, mass m, inertia $I_{\bar{P}}$ with respect to $\bar{P}$)

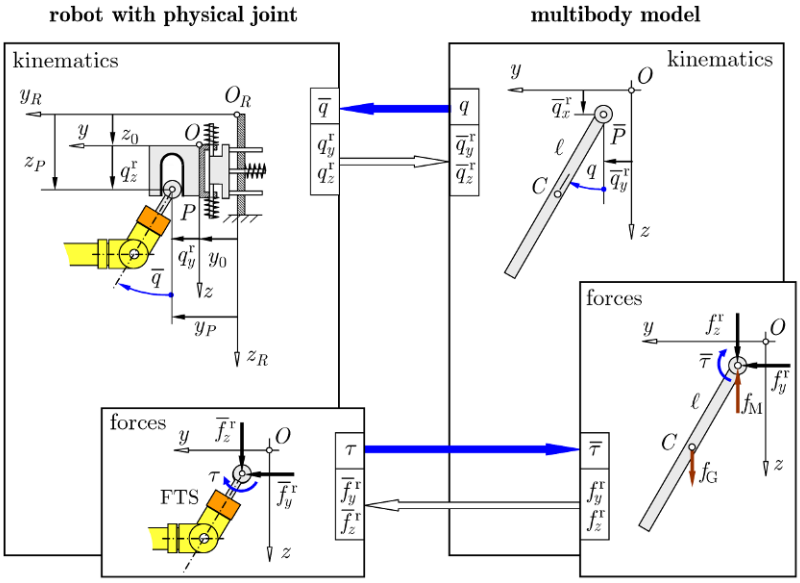


Fig. 5 HiL setup for the mechanical test system of Fig. 4.

$$I_{\bar{P}}\,\ddot{q} = -m\,g\,\ell\,\sin q + \bar{\tau} \tag{1}$$

or in state-space form with the state variables q and $\omega = \dot{q}$

$$\dot{q} = \omega\,, \tag{2}$$

$$\dot{\omega} = \frac{1}{I_{\bar{P}}}(\bar{\tau} - m\,g\,\ell\,\sin q)\,. \tag{3}$$

The reaction forces in point $\bar{P}$ are obtained from the law of momentum under consideration of the accelerations $\ddot{y}_C$, $\ddot{z}_C$ of C and the "muscle" spring force f_{M} expressed by the stiffness c_{M} and the prestress extension s_z for $\bar{q}_z^{\mathrm{r}} = 0$,

$$f_y^{\mathrm{r}} = m\ddot{y}_C \qquad \text{with} \qquad \ddot{y}_C = \ell\cos q\dot{\omega} - \ell\sin q\omega^2\,, \tag{4}$$

$$f_z^{\mathrm{r}} = m\ddot{z}_C + c_{\mathrm{M}}\,(\bar{q}_z^{\mathrm{r}} + s_z) - mg \qquad \text{with} \qquad \ddot{z}_C = -\ell\sin q\dot{\omega} - \ell\cos q\omega^2\,. \tag{5}$$

The reaction forces f_y^{r}, f_z^{r} and the position q (the velocity $\dot{q}$ can be used as an additional control input for the robot) are transferred to the corresponding robot variables $\bar{f}_y^{\mathrm{r}}$, $\bar{f}_z^{\mathrm{r}}$, and $\bar{q}$, whereby first-order transfer functions are used to approximate the dynamics of the controlled robot (time constants T_{fy}^{r}, T_{fz}^{r}, T_q),

$$f_i^{\mathrm{r}} \to \bar{f}_i^{\mathrm{r}}: \qquad T_{fi}^{\mathrm{r}}\,\dot{\bar{f}}_i^{\mathrm{r}} + \bar{f}_i^{\mathrm{r}} = f_i^{\mathrm{r}}\,, \quad i = y,z\,, \tag{6}$$

$$q \to \bar{q}: \qquad T_q\,\dot{\bar{q}} + \bar{q} \quad = q\,. \tag{7}$$

The robot applies the forces $\bar{f}_y^{\mathrm{r}}$, $\bar{f}_z^{\mathrm{r}}$ by means of a force controller. As addressed in Section 2 force control is realized by means of the stiffnesses c_y^{c}, c_z^{c} of the compliant support and the forces acting on the robot point P. The force controller generates velocity components $\dot{y}_P$, $\dot{z}_P$ of the robot point P that are proportional to the differences between the force components $\bar{f}_y^{\mathrm{r}}$, $\bar{f}_z^{\mathrm{r}}$ to be applied and the actual forces f_y^{FTS}, f_z^{FTS} measured by the FTS, respectively, according to (gain factor K_{f})

$$\dot{y}_P = -K_{\mathrm{f}}\,(\bar{f}_y^{\mathrm{r}} - f_y^{\mathrm{FTS}})\,, \tag{8}$$

$$\dot{z}_P = -K_{\mathrm{f}}\,(\bar{f}_z^{\mathrm{r}} - f_z^{\mathrm{FTS}})\,. \tag{9}$$

In the numerical simulation model addressed here the actual forces f_y^{FTS}, f_z^{FTS} are calculated by means of the stiffnesses c_y^{c}, c_z^{c} according to

$$f_y^{\mathrm{FTS}} = -c_y^{\mathrm{c}}(y_P - y_0) \qquad \text{(no dislocation admitted)}\,, \tag{10}$$

$$f_z^{\mathrm{FTS}} = \begin{cases} 0 & \text{for} \quad q_z^{\mathrm{r}} > 0 \quad \text{(dislocation)} \\ -c_z^{\mathrm{c}}(z_P - z_0) & \text{for} \quad q_z^{\mathrm{r}} = 0 \quad \text{(no dislocation).} \end{cases} \tag{11}$$

In order to detect the actual dislocations q_y^{r}, q_z^{r} the displacements y_0, z_0 of the compliant support are measured. The dislocations then are obtained from

$$q_y^{\mathrm{r}} = y_P - y_0\,, \qquad q_z^{\mathrm{r}} = z_P - z_0\,. \tag{12}$$

In the considered example the horizontal dislocation q_y^{r} is always zero as the vertical slot of the bearing block does not admit a dislocation in y-direction. However a dislocation q_z^{r} in vertical direction occurs if the reaction force $\bar{f}_z^{\mathrm{r}}$ becomes negative.

In the direction of the coordinate $\bar{q}$ of the free motion a torque τ may occur for example due to dry friction,

$$\tau = \qquad \text{with} \qquad \bar{f}^{\mathrm{r}} = \sqrt{(\bar{f}_y^{\mathrm{r}})^2 + (\bar{f}_z^{\mathrm{r}})^2}\,, \tag{13}$$

with the frictional coefficient μ and the absolute value of the reaction force $\bar{f}^{\mathrm{r}}$. The actual torque τ is measured by the FTS.

The dislocations q_y^{r}, q_z^{r} and the torque τ are fed back to the simulation model. The transitions into the corresponding values $\bar{q}_y^{\mathrm{r}}$, $\bar{q}_z^{\mathrm{r}}$, $\bar{\tau}$ are again modeled by first-order transfer functions (time constants T_{qy}^{r}, T_{qz}^{r}, T_τ),

$$q_i^{\mathrm{r}} \to \bar{q}_i^{\mathrm{r}}: \qquad T_{qi}^{\mathrm{r}}\,\dot{\bar{q}}_i^{\mathrm{r}} + \bar{q}_i^{\mathrm{r}} = q_i^{\mathrm{r}}\,, \quad i = y,z\,, \tag{14}$$

$$\tau \to \bar{\tau}: \qquad T_\tau\,\dot{\bar{\tau}} + \bar{\tau} \;\; = \tau\,. \tag{15}$$

Results for the numerical integration of the system of differential equations (2), (3), (6), (7), (8), (9), (14), (15) are shown in Fig. 6. The mechanical system is simulated by means of the multibody program SIMPACK as it will be done for the actual biomechanical models. The results show the oscillation $q(t)$ of the pendulum slightly damped by the friction torque from Eq. (13). The robot motion $\bar{q}(t)$

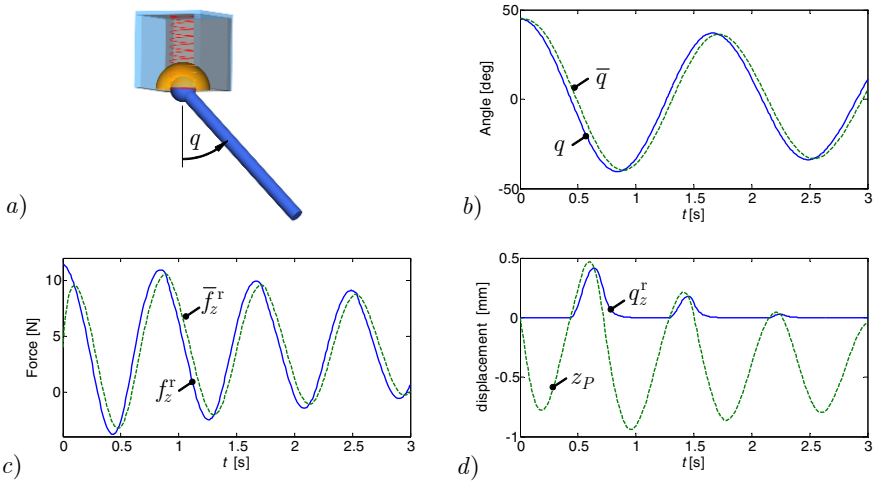


Fig. 6 Simulation for the test system of Fig. 5: (a) SIMPACK model. (b) Pendulum angle $q(t)$ of the multibody model and $\bar{q}(t)$ of the robot. (c) Reaction force $f_z^{\mathrm{r}}(t)$ of the multibody model and $\bar{f}_z^{\mathrm{r}}(t)$ of the robot. (d) Dislocation $q_z^{\mathrm{r}}(t)$ and displacement $z_P(t)$ of the robot point P.

is delayed because of the time constant T_q from Eq. (7). Accordingly the reaction forces $f_z^{\mathrm{r}}(t)$ of the multibody model and $\bar{f}_z^{\mathrm{r}}(t))$ of the robot slightly differ. In z-direction the robot point P dislocates, indicated by $q_z^{\mathrm{r}}(t) > 0$, if its displacement $z_P(t)$ is positive.

4 Biomechanical Multibody Model

At present the described procedure is transferred to the HiL simulation of endoprostheses by means of the setups in Figs. 2 and 3. The biomechanical simulation is based on a SIMPACK multibody system model of the lower extremity. The scalable model consists of five bodies linked by four joints (Fig. 7). In a first approach the ankle joint, the patella-femoral joint and the tibio-femoral joint are simplified by revolute joints. The soft tissue is represented by capsule, ligament and muscle structures mainly based on the data provided by [6]. The muscle forces are obtained by means of inverse dynamics. The redundancy of the muscular actuation is resolved by means of optimization approaches.

5 Conclusion

The functional principle of a HiL simulator for testing of joint endoprostheses is demonstrated by means of a simplified model. The specific property of the HiL concept are two types of control loops being applied in complementary spatial dir-

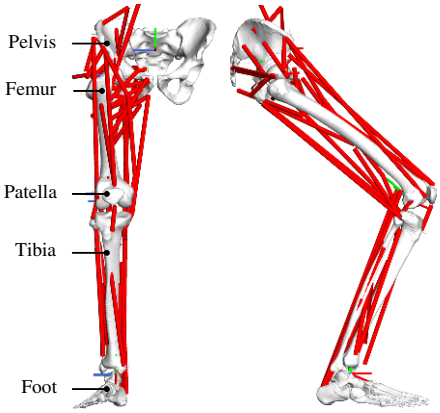


Fig. 7 Musculoskeletal SIMPACK-model of the right leg.

ections of the joint. In the unconstrained directions of the joint the biomechanical model calculates motions to be applied by the robot on the endoprosthesis, and forces/torques are fed back. Vice versa in the constrained directions of the joint the biomechanical model calculates reaction forces to be applied by the robot on the endoprosthesis, and displacements representing possible dislocations are fed back. The relative comparison of different implants under physiological conditions represents a fundamental leap beyond existing testing methods for joint endoprostheses.

Acknowledgements The research work reported here is partly supported by Deutsche Forschungsgemeinschaft under grants no. BA 3347/3-1 and WO 452/8-1.

References

1. Kähler, M., Souffrant, R., Dryba, S., Kluess, D., Bader, R., Woernle, C.: Hardware-in-the-loop-simulator for testing of total hip endoprostheses. In J. van der Sloten et al. (Eds.), *Proc. 4th Eur. Conf. Int. Fed. Med. Biol. Eng. (IFMBE)*, pp. 1785–1788 (2008).
2. Kähler, M., Woernle, C., Bader, R.: Hardware-in-the-loop-simulaton of constraint elements in mechanical systems. In A. Kecskeméthy, A. Müller (Eds.), *Computational Kinematics*, Springer, Berlin, pp. 159–166 (2009).
3. Isermann, R.: *Mechatronic Systems*. Springer, Berlin (2005).
4. Liem, K., Sing, M.D., Leontjew, V., Müller, A., Kecskemethy, A.: Improved force control of a compliant pneumatic-muscle driven parallel platform. In K. Arczewski et al. (Eds.), *Proceedings of the Eccomas Thematic Conference Multibody Dynamics*, Warsaw (2009).
5. Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G.: *Robotics: Modelling, Planning and Control*. Springer, Berlin (2008).
6. Klein Horsman, M.D., Koopman, H.F., van der Helm, F.C., Prose, L.P., Veeger, H.E.: Morphological muscle and joint parameters for musculoskeletal modelling of the lower extremity. *Clin. Biomech.* **22**, 239–247 (2007).

An Algorithm for Real-Time Forward Kinematics of Cable-Driven Parallel Robots

Andreas Pott

Fraunhofer Institute for Manufacturing Engineering and Automation IPA, Stuttgart, Germany; e-mail: andreas.pott@ipa.fraunhofer.de

Abstract. To operate and control a cable-driven parallel robot in practice one has to solve the kinematic transformation in real-time. Therefore, an algorithm is needed which can find a solution within a strictly bounded time period. Since no closed-form solution is known for parallel robots of general geometry, a combination of interval techniques and an iterative solver is proposed and implemented into the real-time control of a cable robot. Experimental results with the cable robot IPAnema are presented.

Key words: Cable-driven parallel robot, forward kinematics, controller, real-time.

1 Introduction

In the last decade, a lot of research has been carried out to study both, theory (see e.g. [1–3]) and implementation [4] of cable-driven parallel robots.

For a mobile platform with n degrees-of-freedom, in general, at least $m = n + 1$ cables are required to fully control the motion [5]. Therefore, many cable robots are under-determined with respect to distribution of forces in the cables and over-determined with respect to forward kinematics (Fig. 1a). As a consequence of the latter, it is challenging to calculate the forward kinematics of the cable robot in real-time. Thus, one has to estimate the pose of the mobile platform from given length of the cables. In the literature, different approaches for that problem were suggested. In general, the forward kinematics of parallel robots can have up to 40 solutions and the algorithm by Husty [6] gives deep insight into the number of solutions and their mathematical structure. Unfortunately it seems inadequate for real-time implementation. Additional cables in general do not necessarily reduces the number of solutions and special geometries maintain this maximum solution set [7]. An incremental forward kinematics to follow a trajectory was presented [8]. Merlet [9] used interval analysis to calculate the forward kinematics of Stewart–Gough platforms in a robust and guaranteed way. A more specialized method for cable robots with linear drives and elastic deformation in the cables was recently presented [10]. A closed-form kinematic code for the so-called 3-2-1 configuration is well suitable for

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_57, © Springer Science+Business Media B.V. 2010

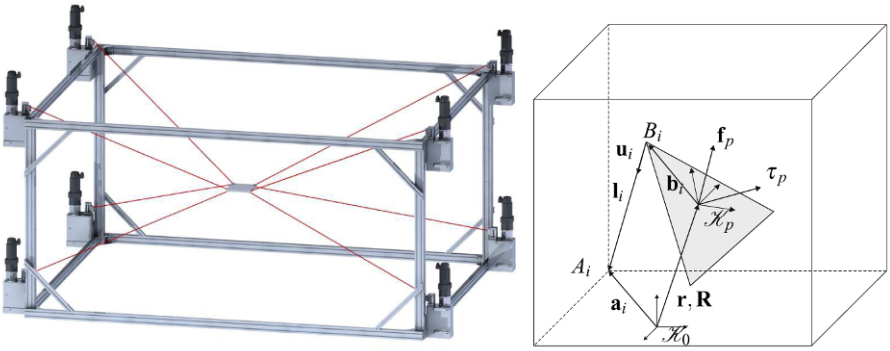


Fig. 1 (a) CAD draft of the spatial cable-driven parallel robot IPAnema with eight cables and six degrees-of-freedom. (b) Geometry and kinematics of a general cable robot.

real-time application [11, 12] but relies on a special non-generic geometry. Bruckmann [13] presented a method to cope with winches using pulley mechanisms to guide the cables.

2 Forward Kinematics of Cable-Driven Parallel Robots

For better reference, the kinematic foundation of cable robots are briefly reviewed. Figure 1b shows the kinematic structure of a spatial cable robot, where the vectors $\mathbf{a}_i$ denote the proximal anchor points on the robot base, the vectors $\mathbf{b}_i$ are the relative positions of the distal anchor points on the movable platform, and $\mathbf{l}_i$ denote the vector of the cables. The length of the cables is abbreviated by $l_i = ||\mathbf{l}_i||_2$. Applying a vector loop, the closure-constraint reads

$$\mathbf{a}_i - \mathbf{r} - \mathbf{R}\mathbf{b}_i - \mathbf{l}_i = \mathbf{0} \quad \text{for} \quad i = 1, \ldots, m, \tag{1}$$

where the vector $\mathbf{r}$ is the Cartesian position of the platform and the rotation matrix $\mathbf{R}$ represents the orientation of the platform. We denote the full pose composed from position and orientation of the mobile platform by $\mathbf{y}$.

This paper presents an algorithm for forward kinematics to be used rather for real-time control of a well designed robot than for analysis of possibly ill-conditions or architecturally singular robots. The following assumptions were made taking into account practical needs:

- The pose $\mathbf{y}$ of the mobile platform to be estimated belongs to the workspace (positive tension in the cables) and the control error measured by the length sensors is moderate. If the cable lengths are too short it may cause either overloading the motors or breaking the cables. If the cable lengths are too long, we loose

the control on the platform. In both cases, the control system must perform an emergency stop.
- The cables of the robot are elastic allowing for small changes in length around the given length l_i. Nevertheless, the presented algorithm does not take into account changes in the length due to the actual tension.
- The geometry of the mobile platform $\mathbf{b}_i$ was chosen such that the rotation matrix $\mathbf{R} = \mathbf{I}_3$ is in the workspace or close to the workspace. This is a minor restriction since cable robots allow only for relatively small orientation workspace.
- The cable robot has more cables m than degrees-of-freedom n, i.e. it is kinematically over-constrained.
- The size of the mobile platform is small compared to the machine frame, i.e. $||\mathbf{b}_i - \mathbf{b}_j||_2 \ll ||\mathbf{a}_i - \mathbf{a}_j||_2$ for $i, j = 1, \ldots, m \quad i \neq j$.

The algorithm should satisfy the following requirements:

- Real-time capability: the computation time of the algorithm must be strictly bounded and in the range of milliseconds on available real-time hardware.
- The geometry of the robot is generic i.e. no special constraints like linearity, planarity, etc. are assumed for the mobile platform $\mathbf{b}_i$ or the machine frame $\mathbf{a}_i$. Nevertheless, it is assumed that the robot geometry was designed to avoid architectural singularities and the like.
- Errors have to be reported reliably e.g. if no solution was found because it does not exist. Note that for the control system such conditions are exceptions requiring an emergency stop of the robot.

From Eq. (1) we receive m nonlinear equations for forward kinematics

$$\Psi_i(\mathbf{l}, \mathbf{r}, \mathbf{R}) = ||\mathbf{a}_i - \mathbf{r} - \mathbf{R}\mathbf{b}_i||_2 - l_i^2 = 0 \quad \text{for} \quad i = 1, \ldots, m \tag{2}$$

that form an over-constrained system. Here, we consider the cables to be linear springs. In general we cannot expect to solve the above equations exactly, but we can minimize the error which can be interpreted as minimizing the potential energy in pre-tensed cables which yields the function for forward kinematics

$$\phi(\mathbf{l}) = \min_{\mathbf{r},\mathbf{R}} \sum_{i}^{m} \Psi_i(\mathbf{l}, \mathbf{r}, \mathbf{R}), \tag{3}$$

where the given vector $\mathbf{l} = [l_1, \ldots, l_m]^{\mathrm{T}}$ is the vector containing the cable lengths. Then the function $\phi(\mathbf{l})$ yields the values $\mathbf{r}^*, \mathbf{R}^*$ that minimize the right hand side of Eq. (3).

3 Real-Time Algorithm

In the literature, iterative schemes [8, 13] as well as interval methods [9, 10] were proposed for forward kinematics where the former methods may suffer from not

converging, while the latter does not fulfill real-time constraints. In this contribution, it is proposed to combine both approaches in a two-step process. Firstly, an initial solution for the pose of the platform $\mathbf{y}_0$ is estimated together with guaranteed bounds through an interval analysis inspired technique. Secondly, a Levenberg–Marquardt algorithm is used to iterate the platform pose from this initial estimate through a least square approach of the over-constrained nonlinear equations.

In the first step of the proposed algorithm an estimate of the pose $\mathbf{y}_0$ is determined. To estimate the initial position of the platform an interval analysis inspired approach is adopted. Although inspired by interval analysis, the implementation performs the computation with standard arithmetics since interval libraries are not available on all real-time systems. The basic idea is to strictly bound the position of the TCP. This can be done by axis-aligned bounding boxes that are placed around the winches. For cable-driven parallel robots the anchor points $\mathbf{a}_i$ are distributed around the workspace. Therefore, the region of intersection of these boxes is relatively small and can be used as an initial estimate for the position.

The vector loop Eq. (1) can be rewritten to

$$\mathbf{a}_i - \mathbf{r} = \mathbf{l}_i + \mathbf{R}\mathbf{b}_i \tag{4}$$

and applying the triangle inequality removed the rotation matrix $\mathbf{R}$ to receive

$$||\mathbf{a}_i - \mathbf{r}||_2 \leq l_i + ||\mathbf{b}_i||_2. \tag{5}$$

Thus, the TCP lies inside a sphere with radius $l_i + ||\mathbf{b}_i||_2$ around anchor point $\mathbf{a}_i$. Using an interval estimation for this sphere by enclosing the sphere with a box, one receives the bounds

$$\left.\begin{aligned} \mathbf{r}_i^{low} &= \mathbf{a}_i - (l_i + ||\mathbf{b}_i||_2)[1,1,1]^{\mathrm{T}} \\ \mathbf{r}_i^{high} &= \mathbf{a}_i + (l_i + ||\mathbf{b}_i||_2)[1,1,1]^{\mathrm{T}} \end{aligned}\right\}. \tag{6}$$

Then we can calculate the intersection of all m bounding boxes from

$$\mathbf{r}^{low} = \max_i \mathbf{r}_i^{low} \quad \text{and} \quad \mathbf{r}^{high} = \min_i \mathbf{r}_i^{high}. \tag{7}$$

The center $\mathbf{r}_0 = \frac{1}{2}(\mathbf{r}^{low} + \mathbf{r}^{high})$ of this bounding box is used as initial estimate. If $\mathbf{r}^{low} > \mathbf{r}^{high}$ then no solutions exist. Note that the equations above are trivial to implement on a computer.

In Fig. 2 a simplified example of a planar cable robot with three winches is illustrated. The dashed circles are centered around the anchor points $\mathbf{a}_i$ and have a radius of $l_i + ||\mathbf{b}_i||_2$. The rectangles represent the interval estimate of these circles. The light gray rectangle is the intersection of the boxes and is guaranteed to contain the TCP. The dark gray region can be computed from the intersection of circles which typically gives a better estimate, but is also more complex to determine. For example using interval consistency methods one can shrink the box to contain only the dark gray region [14]. However, here we simply use the center of the light gray box

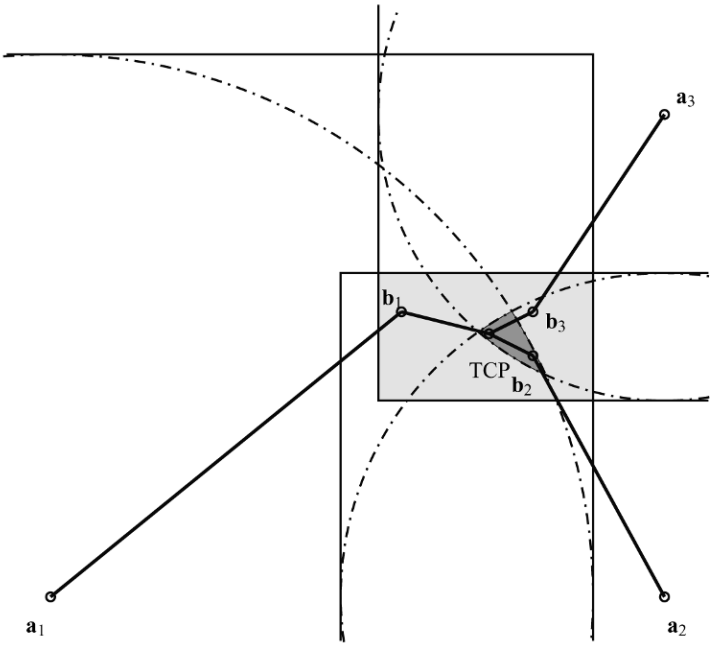


Fig. 2 Bounding of solution with axis aligned boxes.

to start the iteration with the Levenberg–Marquardt method. Note that this bounding technique exploits the over-constrained equations since more equations impose more restrictions on the box and thus produce more accurate estimates.

To determine the pose $\mathbf{y}$ of the cable robot, a Levenberg–Marquardt method is applied [15]. Given a function $\phi : \mathbb{R}^n \to \mathbb{R}^m$ with $m \geq n$, the Levenberg–Marquardt algorithm can be used for obtaining the argument $\mathbf{y}$ that minimizes $||\phi(\mathbf{y})||_2$. This is done by an iterative procedure $\mathbf{y}_{i+1} = \mathbf{y}_i + \mathbf{h}$ where a step $\mathbf{h}$ of the Levenberg–Marquardt algorithm is determined by solving the linear system

$$\left[\mathbf{J}(\mathbf{y}_i)\mathbf{J}^{\mathrm{T}}(\mathbf{y}_i) + \mu\mathbf{I}\right]\mathbf{h} = \mathbf{J}^{\mathrm{T}}(\mathbf{y}_i)\phi(\mathbf{y}), \tag{8}$$

where μ is the damping parameter and $\mathbf{J}(\mathbf{y}_i)$ is the Jacobian of $\phi(\mathbf{y}_i)$. The procedure is terminated if the improvement from the current step is small $||\mathbf{h}_i||_2 < \varepsilon_2(||\mathbf{y}||_2 + \varepsilon_2)$ or if $||J(\mathbf{y}_i)\phi(\mathbf{y}_i)||_2 < \varepsilon_1$.

The computational effort for the bounding procedure is constant and negligibly small. The effort for each iteration step is constant and a maximum number of iterative steps can be defined. Thus, the algorithm can be integrated into a real-time environment, given that a reasonable small number of steps is needed. In Section 4 the convergency of the algorithms is investigated.

Remark: The efficiency of this estimation procedure can be improved by performing a rigid body transformation to the platform anchor points $\mathbf{b}_i$. Firstly, the

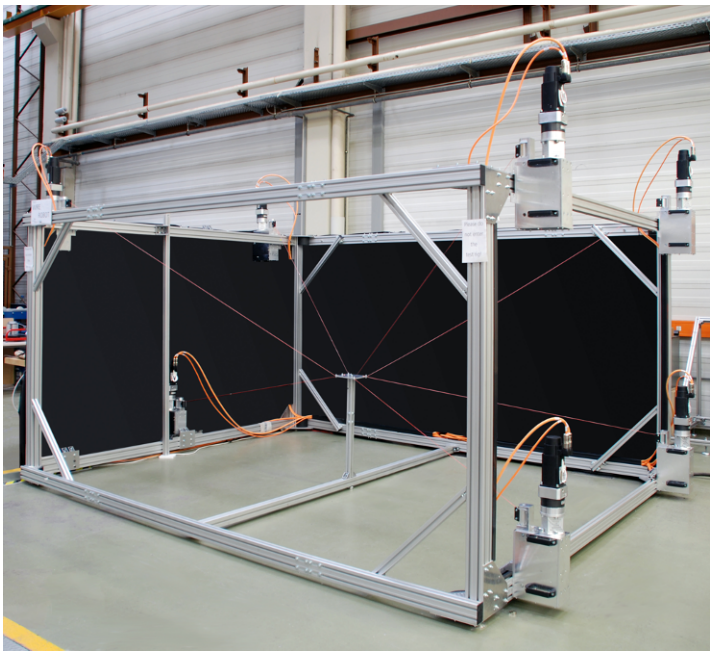


Fig. 3 Cable-driven parallel robot IPAnema in seven-cable setup.

TCP should be translated to the geometric center of platform $\mathbf{b}_c = \frac{1}{m}\sum_i^m \mathbf{b}_i$. Then the platform anchor points are transformed by $\mathbf{b}'_i = \mathbf{b}_i - \mathbf{b}_c$. Secondly, a rotation $\mathbf{R}_c$ is applied to the anchor points $\mathbf{b}''_i = \mathbf{R}_c \mathbf{b}'_i$ such that the orientation $\mathbf{R} = \mathbf{I}_3$ is inside the workspace. Clearly, the determined pose has to be transformed back to the original coordinate system by rotating with $\mathbf{R}_C^{\mathrm{T}}$ and translating along $\mathbf{b}_c$.

4 Implementation and Experimental Results

The cable-driven parallel robot IPAnema (Fig. 3) is currently under investigation at the laboratories of Fraunhofer IPA. This robot provides a six degrees-of-freedom end-effector with seven or eight cables and focuses on industrial applications in the field of material handling as well as fast pick-and-place applications. The winches are equipped with multi-turn absolute encoders allowing to obtain the absolute cable length at any time with a resolution of 50 µm. The control system is based on the PC-based real-time extension RTX and an adopted NC-controller by ISG (Stuttgart, Germany). The robot can be programmed by G-Code (DIN 66025) similar to machine tools. On the industrial PC the interpolation cycle time of the trajectory generator is 2 ms. The kinematic code for inverse and forward kinematics described in

Table 1 IPAnema's geometrical parameters: platform vectors **b** and base vectors **a**.

cable i	platform vector $\mathbf{b}_i$	base vector $\mathbf{a}_i$
1	$[-1.8, 1.5, 2.0]^{\mathrm{T}}$	$[-0.1, 0.1, 0.0]^{\mathrm{T}}$
2	$[1.8, 1.5, 2.0]^{\mathrm{T}}$	$[0.1, 0.1, 0.0]^{\mathrm{T}}$
3	$[1.8, -1.5, 2.0]^{\mathrm{T}}$	$[0.1, -0.1, 0.0]^{\mathrm{T}}$
4	$[-1.8, -1.5, 2.0]^{\mathrm{T}}$	$[-0.1, -0.1, 0.0]^{\mathrm{T}}$
5	$[-2.0, 1.3, 0.0]^{\mathrm{T}}$	$[-0.1, 0.1, 0.2]^{\mathrm{T}}$
6	$[2.0, 1.3, 0.0]^{\mathrm{T}}$	$[0.1, 0.1, 0.2]^{\mathrm{T}}$
7	$[2.0, -1.3, 0.0]^{\mathrm{T}}$	$[0.1, -0.1, 0.2]^{\mathrm{T}}$
8	$[-2.0, -1.3, 0.0]^{\mathrm{T}}$	$[-0.1, -0.1, 0.2]^{\mathrm{T}}$

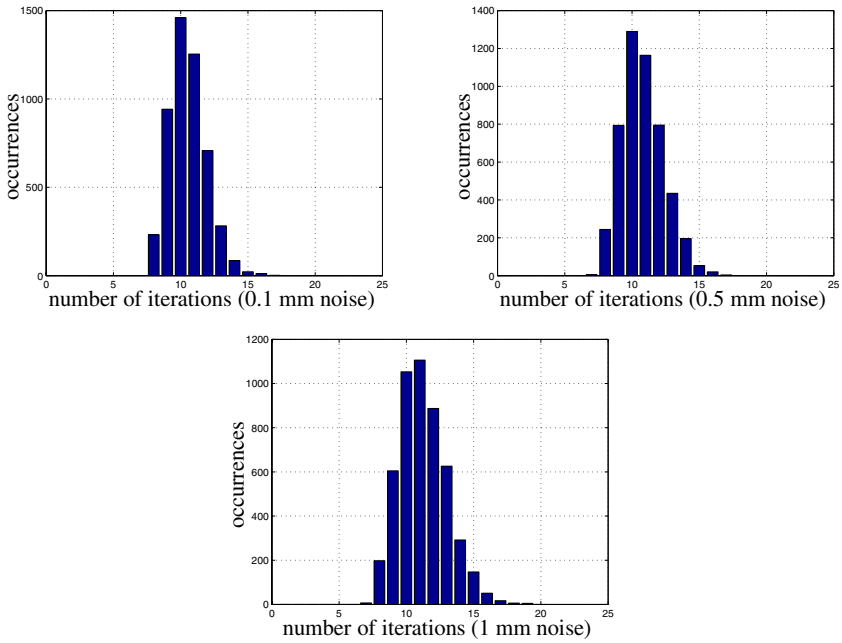


Fig. 4 Histogram of number of iterations for forward kinematics with Levenberg–Marquardt algorithm for noise 0.1 mm, 0.5 mm, 1 mm on the cable length.

Section 2 is implemented into that control system in C language, where the implementation of the Levenberg–Marquardt algorithm is based on [16].

The computation time was determined both on a standard PC (Intel Core 2 Duo, 2.26 GHz) and on the PC-based real-time controller system (Intel Core 2 Duo, 2.4 GHz). For testing the geometry of the IPAnema system was used (Table 1). Totally 5000 randomly chosen poses $\mathbf{y}_i$ within the workspace of the robot were tested and different magnitudes of noise were added to the cable length simulating measurement and control errors. The thresholds for termination of the Levenberg–Marquardt algorithm were chosen to be $\varepsilon_1, \varepsilon_2 = 10^{-17}$. The maximum number of

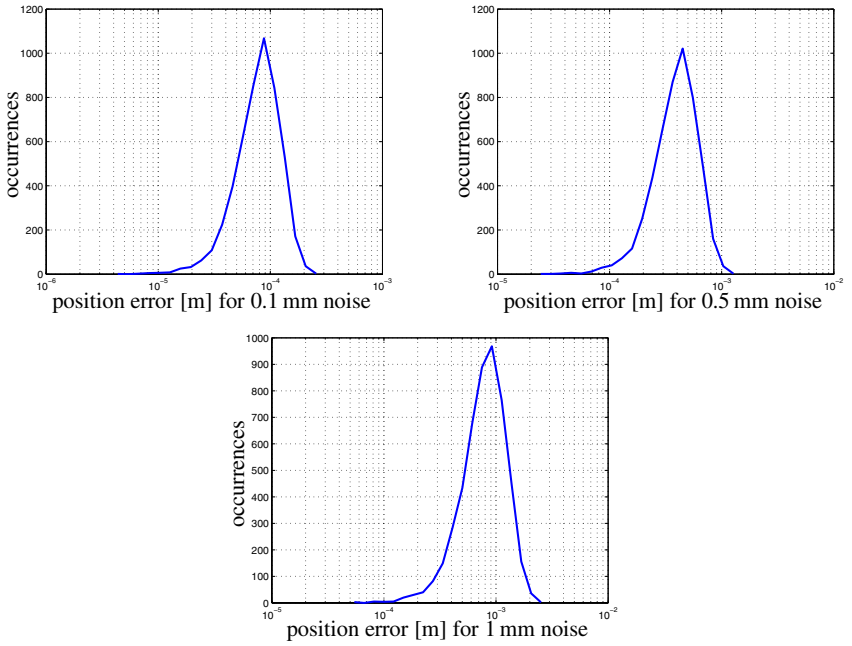


Fig. 5 Histogram of position error for forward kinematics with Levenberg–Marquardt algorithm for noise 0.1 mm, 0.5 mm, 1 mm on the cable length.

iterations was set to 100 which was never reached in practice. On the desktop PC the average computation time per evaluation was determined to be 97 µs. Since the used Windows operating system lacks a high-precision timer with reliable resolutions less than one milliseconds, the worst case computation time could not be determined. However, the measured average time and the distribution of needed iterations (Fig. 4) were encouraging. The number of iterations was between 7 and 20 for all poses tested where a typical number of ten iterations is needed. The determined error between the nominal pose $\mathbf{y}_i$ and the determined pose were correlated to the noise in the cable length (Fig. 5). One can see that the average error of the poses is almost equal to the errors in cable length.

On the real-time controller system it is not possible to measure the exact time which was consumed for the kinematic transformation but one can only measure the overall time consumed by the transformation and all other controller codes for each cycle. During some hours of operations no violations of the cycle-time were reported by the real-time system and the computation time for the whole NC-kernel including the kinematic transformation while moving along smooth trajectories was always less than 1 ms.

5 Conclusions

In this paper an algorithm for real-time forward kinematics of cable-driven parallel robots is presented. The algorithm was designed for usage in a NC-based robot controller. To solve the over-constrained forward kinematics an optimization approach was proposed. In a first step, the solution is bounded using simple interval considerations, which work well with the typical structure of over-constrained cable robots. In the second step, a Levenberg–Marquardt algorithm was used to iterate the sought pose of the cable robot. Experimental evaluation demonstrate a robust behaviour under moderate noise. The successful implementation and operation in the real-time controller system proves the applicability. Ongoing work address the extension of the kinematic equations for forward kinematics to consider pulley mechanisms.

References

1. R. Verhoeven. Analysis of the workspace of tendon-based Stewart platforms. PhD thesis, University of Duisburg-Essen (2004).
2. M. Hiller, S. Fang, S. Mielczarek, R. Verhoeven, and D. Franitza. Design, analysis and realization of tendon-based parallel manipulators. *Mechanism and Machine Theory*, **40**(4), 429–445 (2005).
3. M. Gouttefarde, J.-P. Merlet, and D. Daney. Wrench-feasible workspace of parallel cable-driven mechanisms. In *IEEE International Conference on Robotics and Automation*, Roma, Italy, pp. 1492–1497 (2007).
4. J.-P. Merlet and D. Daney. A new design for wire-driven parallel robot. In *2nd Int. Congress, Design and Modelling of Mechanical Systems* (2007).
5. A. Ming and T. Higuchi. Study on multiple degree-of-freedom positioning mechanism using wires (Part 1) – Concept, design and control. *Int. Journal of the Jap. Soc. for Precision Engineering*, **28**(2), 131–138 (1994).
6. M.L. Husty. An algorithm for solving the direct kinematic of Stewart–Gough-type platforms. *Mechanism and Machine Theory*, **31**(4), 365–380 (1996).
7. M.L. Husty, S. Mielczarek, and M. Hiller. A redundant spatial Stewart–Gough platform with maximal forward kinematics solution set. In *Advances in Robot Kinematics: Theory and Applications*, J. Lenarčič and F. Thomas (Eds.), Kluwer Academic Publishers, pp. 147–154 (2002).
8. S. Fang. *Design, Modeling and Motion Control of Tendon-Based Parallel Manipulators*. Fortschritt-Berichte VDI, Reihe 8, Nr. 1076. VDI Verlag, Düsseldorf (2005).
9. J.-P. Merlet. Solving the forward kinematics of a Gough-type parallel manipulator with interval analysis. *The International Journal of Robotics Research*, **23**(3), 221–235 (2004).
10. J.-P. Merlet. Kinematics of the wire-driven parallel robot marionet using linear actuators. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Pasadena, CA, (2008).
11. F. Thomas, E. Ottaviano, L. Ros, and M. Ceccarelli. Uncertainty model and singularities of 3-2-1 wire-based tracking systems. In *Advances in Robot Kinematics: Theory and Applications*, J. Lenarčič and F. Thomas (Eds.), Kluwer Academic Publishers, pp. 107–116 (2002).
12. A. Pott. Forward kinematics and workspace determination of a wire robot for industrial applications. In *Advances in Robot Kinematics: Analysis and Design*, J. Lenarčič and P. Wenger (Eds.). Springer, pp. 451–458 (2008).

13. T. Bruckmann, L. Mikelsons, T. Brandt, M. Hiller, and D. Schramm. Wire robots Part I – Kinematics, analysis and design. In *Parallel Manipulators – New Developments*, ARS Robotic Books, I-Tech Education and Publishing, Vienna, Austria (2008).
14. E. Hansen and G.W. Walster. *Global Optimization Using Interval Analysis*. Marcel Dekker (2004).
15. K. Madsen, H.B. Nielsen, and O. Tingleff. Methods for non-linear least squares problems, lecture notes. Technical Report, Technical University of Denmark (2004).
16. M.I.A. Lourakis. LEVMAR: Levenberg–Marquardt nonlinear least squares algorithms in C/C++. http://www.ics.forth.gr/ lourakis/levmar/+ (2004). [Accessed on 31 January 2005]

Numerical Synthesis of Overconstrained Mechanisms Based on Screw Theory

René Bartkowiak and Christoph Woernle

Department of Mechanical Engineering and Marine Technology, University of Rostock, Germany; e-mail: {rene.bartkowiak, woernle}@uni-rostock.de

Abstract. An approach for the synthesis of overconstrained single-loop mechanisms with helical joints using screw theory is presented. By expressing higher-order derivatives of the screw axes with respect to the joint coordinates a local approximation of the closure conditions is obtained. The approximation is used to define conditions for the finite mobility of overconstrained mechanisms.

Key words: Overconstrained mechanism, screw system, mechanism synthesis, loop closure.

1 Introduction

In general the degree of freedom (DOF) of closed single-loop mechanisms can be determined by means of the Grübler–Kutzbach Criterion (GKC). In the GKC the number of links as well as the degree of freedom of the joints are used. Curiously several single-loop mechanisms are known which have the mobility $f_{\mathrm{M}} = 1$ although their DOF is $f_{\mathrm{GKC}} < 1$ because of their special geometry [1, 2]. These mechanisms are called overconstrained and are often investigated especially with respect to kinematical analysis [3–5].

In recent years general analysis methods based on group theory have been applied [6] but in the field of synthesis special methods are often used [7, 8]. Until now there exists no uniform approach to synthesise spatial overconstrained single-loop mechanisms because of their complexity. In the present paper a numerical synthesis method for overconstrained single-loop mechanisms with helical joints is presented. The well-known principle of transference is used to transfer relations from spherical geometry to spatial line geometry.

Sections 2 and 3 contain fundamentals of screw theory for the analysis of open kinematical chains as well as closed single-loop mechanisms with helical joints. A new synthesis method for overconstrained single-loop mechanisms is described in Section 4. Finally the numerical synthesis algorithm is presented.

J. Lenarčič and M.M. Stanišić (eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, DOI 10.1007/978-90-481-9262-5_58, © Springer Science+Business Media B.V. 2010

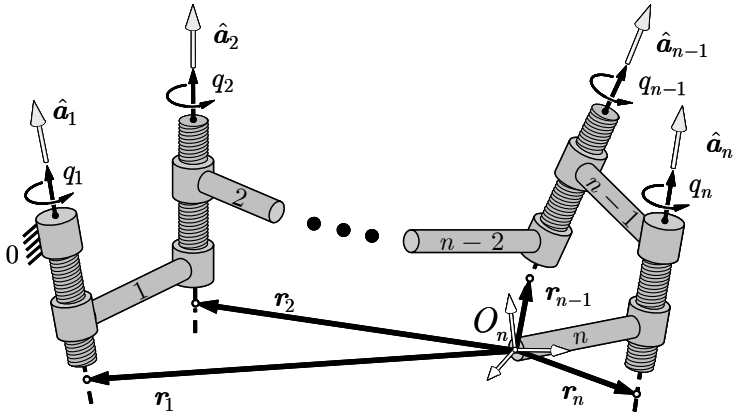


Fig. 1 Open kinematical chain with helical joints.

2 Open Kinematical Chains with Helical Joints

An open kinematic chain with bodies 1 to n connected by helical joints according to Fig. 1 is considered.

2.1 Screw System of the Helical Joints

The actual position of the screw axes of the helical joints is described by unit screws here written in the form

$$\hat{\underline{a}}_i \equiv \begin{bmatrix} \boldsymbol{a}_i \\ \boldsymbol{a}_{\varepsilon i} \end{bmatrix} = \begin{bmatrix} \boldsymbol{a}_i \\ \boldsymbol{r}_i \times \boldsymbol{a}_i + h_i \boldsymbol{a}_i \end{bmatrix}, \quad i = 1, \ldots, n, \tag{1}$$

with the unit vectors $\boldsymbol{a}_i$ of the screw axes, the moments $\boldsymbol{r}_i \times \boldsymbol{a}_i$ of the vectors $\boldsymbol{a}_i$ with respect to a common reference point, here point O_n on body n, and the pitches h_i of the helical joints. In dual-number notation the vectors $\boldsymbol{a}_i$ and $\boldsymbol{a}_{\varepsilon i}$ are the real and the dual parts of a dual vector $\hat{\boldsymbol{a}}_i = \boldsymbol{a}_i + \varepsilon \boldsymbol{a}_{\varepsilon i}$, $\varepsilon^2 = 0$.

For given angular velocities $\dot{q}_i = \frac{\mathrm{d}q_i}{\mathrm{d}t}$ of the helical joints the velocity of the coordinate system $\mathcal{K}_n$ on body n, consisting of the angular velocity $\boldsymbol{\omega}_n$ of $\mathcal{K}_n$ and the velocity $\boldsymbol{v}_n$ of the origin O_n with respect to the fixed system, is given by

$$\begin{bmatrix} \boldsymbol{\omega}_n \\ \boldsymbol{v}_n \end{bmatrix} = \sum_{i=1}^{n} \begin{bmatrix} \boldsymbol{a}_i \\ \boldsymbol{a}_{\varepsilon i} \end{bmatrix} \dot{q}_i \qquad \text{or} \qquad \hat{\underline{\boldsymbol{\omega}}}_n = \sum_{i=1}^{n} \hat{\underline{\boldsymbol{a}}}_i \, \dot{q}_i . \tag{2}$$

The screws $\hat{\underline{\boldsymbol{a}}}_i$, $i = 1, \ldots, n$, span a p-system of screws (*twists*) where p is the rank of the $(6, n)$ Jacobian matrix $\boldsymbol{J} = \left[\hat{\underline{\boldsymbol{a}}}_1 \; \ldots \; \hat{\underline{\boldsymbol{a}}}_n \right]$, thus $p = \mathrm{rank}\,(\boldsymbol{J}) \leq n$. There exist $6 - p$ linearly independent *wrenches*

$$\hat{\underline{\boldsymbol{k}}}_j = \begin{bmatrix} \boldsymbol{f}_j \\ \boldsymbol{\tau}_j \end{bmatrix}, \qquad j = 1, \ldots, 6-p, \tag{3}$$

comprising force vectors $\boldsymbol{f}_j$ with action lines passing through point O_n and torques $\boldsymbol{\tau}_j$ with respect to O_n. If applied to body n the wrenches $\hat{\underline{\boldsymbol{k}}}_j$ do not perform work along the twists $\hat{\underline{\boldsymbol{a}}}_i$, thus $\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{\tau}_j + \boldsymbol{a}_{\varepsilon i}^{\mathrm{T}}\boldsymbol{f}_j = 0$ or in matrix notation

$$0 = \hat{\underline{\boldsymbol{a}}}_i^{\mathrm{T}} \Delta \hat{\underline{\boldsymbol{k}}}_j, \quad i = 1, \ldots, n; \quad j = 1, \ldots, 6-p; \quad \text{with } \Delta \equiv \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} \\ \boldsymbol{I} & \boldsymbol{0} \end{bmatrix}. \tag{4}$$

Here Δ is a permutation matrix. The $(6-p)$-system of screws $\hat{\underline{\boldsymbol{k}}}_j$, $j = 1, \ldots, 6-p$, is called reciprocal to the p-system of screws $\hat{\underline{\boldsymbol{a}}}_i$, $i = 1, \ldots, n$.

2.2 Differential Displacement of the Screw Axes

In the actual position of the kinematic chain of Fig. 1 relative differential rotations of the helical joints expressed by the joint angles $\mathrm{d}q_i$, $i = 1, \ldots, n$, are applied. The resulting differential displacement of the joint axis $\mathrm{d}\hat{\underline{\boldsymbol{a}}}_k$ then depends on the rotations $\mathrm{d}q_i$, $i = 1, \ldots, k-1$, of the joints between the base 0 and the joint axis $\hat{\underline{\boldsymbol{a}}}_k$. Transferring the differential rotations of a vector to line geometry by means of the principle of transference $\mathrm{d}\hat{\underline{\boldsymbol{a}}}_k$ can be expressed by means of a sum of the dual vector products [9] according to

$$\mathrm{d}\hat{\boldsymbol{a}}_k = \sum_{i=1}^{k-1} \hat{\boldsymbol{a}}_i \times \hat{\boldsymbol{a}}_k \, \mathrm{d}q_i. \tag{5}$$

With the matrix notation of Eq. (1) this expression can be written as

$$\mathrm{d}\hat{\underline{\boldsymbol{a}}}_k = \sum_{i=1}^{k-1} \tilde{\underline{\hat{\boldsymbol{a}}}}_i \, \hat{\underline{\boldsymbol{a}}}_k \, \mathrm{d}q_i \quad \text{with} \quad \tilde{\underline{\hat{\boldsymbol{a}}}} \equiv \begin{bmatrix} \tilde{\boldsymbol{a}} & \boldsymbol{0} \\ \tilde{\boldsymbol{a}}_\varepsilon & \tilde{\boldsymbol{a}} \end{bmatrix}, \tag{6}$$

where the tilde operator $\tilde{}$ transforms a vector $\boldsymbol{a}$ to the skew-symmetric tensor $\tilde{\boldsymbol{a}}$, thus $\tilde{\boldsymbol{a}}\boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$.

3 Single Loop Spatial Mechanisms with Helical Joints

Connecting body n of the open kinematic chain of Fig. 1 with the base body 0 leads to a closed loop mechanism with n helical joints (nH mechanism).

3.1 Closure Condition at the Velocity Level

From Eq. (2) the loop closure conditions for the joint velocities $\dot{q}_i$ in the actual position are given by

$$\mathbf{0} = \sum_{i=1}^{n} \hat{\underline{\boldsymbol{a}}}_i \, \dot{q}_i . \tag{7}$$

In the following overconstrained mechanisms with $n \leq 6$ and one degree of freedom are considered. Without loss of generality the helical joint axes $\hat{\underline{\boldsymbol{a}}}_1$ and $\hat{\underline{\boldsymbol{a}}}_n$ are assumed to be fixed and the joint angle q_n is defined as the independent coordinate.

The closure condition (7) can be written as

$$\hat{\underline{\boldsymbol{a}}}_n(q_n) = \left[\hat{\underline{\boldsymbol{a}}}_1 \; \dots \; \hat{\underline{\boldsymbol{a}}}_{n-1} \right] \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_{n-1} \end{bmatrix} \equiv \boldsymbol{A}(q_n) \, \boldsymbol{\lambda}(q_n) \quad \text{with } \lambda_i = -\frac{\dot{q}_i}{\dot{q}_n}. \tag{8}$$

From Eq. (8) a *necessary* condition results for the global mobility of a nH mechanism ($n \leq 6$) in the actual position $q_n = q_{n0}$: The screw axes $\hat{\underline{\boldsymbol{a}}}_i$, $i = 1, \dots, n$, must be linearly dependent.

3.2 Differential Displacement of the Screw Axes

To define conditions for finite mobility a Taylor series expansion of the screw axes $\hat{\underline{\boldsymbol{a}}}_i$ with respect to the independent coordinate q_n will be needed in Section 4. With the differential displacements of the screw axes $\hat{\underline{\boldsymbol{a}}}_i$ given by

$$\begin{aligned}
\mathrm{d}\hat{\underline{\boldsymbol{a}}}_1 &= \mathbf{0}, \\
\mathrm{d}\hat{\underline{\boldsymbol{a}}}_2 &= \widetilde{\hat{\underline{\boldsymbol{a}}}}_1 \, \hat{\underline{\boldsymbol{a}}}_2 \, \mathrm{d}q_1, \\
&\vdots \\
\mathrm{d}\hat{\underline{\boldsymbol{a}}}_{n-1} &= \widetilde{\hat{\underline{\boldsymbol{a}}}}_1 \, \hat{\underline{\boldsymbol{a}}}_{n-1} \, \mathrm{d}q_1 + \widetilde{\hat{\underline{\boldsymbol{a}}}}_2 \, \hat{\underline{\boldsymbol{a}}}_{n-1} \, \mathrm{d}q_2 + \dots + \widetilde{\hat{\underline{\boldsymbol{a}}}}_{n-2} \, \hat{\underline{\boldsymbol{a}}}_{n-1} \, \mathrm{d}q_{n-2}, \\
\mathrm{d}\hat{\underline{\boldsymbol{a}}}_n &= \mathbf{0}
\end{aligned} \tag{9}$$

the derivatives $\hat{\boldsymbol{a}}_i' = \frac{\mathrm{d}\hat{\boldsymbol{a}}_i}{\mathrm{d}q_n}$ are expressed with $\lambda_i = -\frac{\mathrm{d}q_i}{\mathrm{d}q_n}$ by

$$\begin{aligned}
\hat{\underline{\boldsymbol{a}}}_1' &= \mathbf{0}, \\
\hat{\underline{\boldsymbol{a}}}_2' &= -\widetilde{\hat{\underline{\boldsymbol{a}}}}_1 \, \hat{\underline{\boldsymbol{a}}}_2 \, \lambda_1, \\
&\vdots \\
\hat{\underline{\boldsymbol{a}}}_{n-1}' &= -\widetilde{\hat{\underline{\boldsymbol{a}}}}_1 \, \hat{\underline{\boldsymbol{a}}}_{n-1} \, \lambda_1 - \widetilde{\hat{\underline{\boldsymbol{a}}}}_2 \, \hat{\underline{\boldsymbol{a}}}_{n-1} \, \lambda_2 - \dots - \widetilde{\hat{\underline{\boldsymbol{a}}}}_{n-2} \, \hat{\underline{\boldsymbol{a}}}_{n-1} \, \lambda_{n-2}, \\
\hat{\underline{\boldsymbol{a}}}_n' &= \mathbf{0}.
\end{aligned} \tag{10}$$

Thus the derivative of the matrix $\boldsymbol{A}$ from Eq. (8) with respect to q_n is given by $\boldsymbol{A}'(q_n, \boldsymbol{\lambda}(q_n)) = [\,\hat{\underline{\boldsymbol{a}}}'_1 \ \ldots \ \hat{\underline{\boldsymbol{a}}}'_{n-1}\,]$. The second-order derivative of $\boldsymbol{A}$ is obtained by means of the chain rule,

$$\boldsymbol{A}''(q_n, \boldsymbol{\lambda}(q_n)) = \frac{\partial \boldsymbol{A}'}{\partial q_n} + \sum_{i=1}^{n-1} \frac{\partial \boldsymbol{A}'}{\partial \lambda_i} \lambda'_i.$$

4 Synthesis of Overconstrained Spatial Mechanisms

In the following a numerical algorithm for synthesising overconstrained single-loop mechanisms with $n \leq 6$ helical joints is described for the example of mechanisms with $n = 6$ helical joints.

The independent joint variable is q_6, and the actual position is q_{60}. The axes $\hat{\underline{\boldsymbol{a}}}_1$ and $\hat{\underline{\boldsymbol{a}}}_6$ are assumed to be fixed. With $n = 6$ the loop closure condition (8) in the actual position reads $\hat{\underline{\boldsymbol{a}}}_6 = \boldsymbol{A}(q_{60})\boldsymbol{\lambda}(q_{60})$. For the mobility of the mechanism it is necessary, that the joint axes in the actual position form a 5-system of twists, thus $\operatorname{rank}([\,\boldsymbol{A}(q_{60})\ \hat{\underline{\boldsymbol{a}}}_6\,]) = 5$. According to Eq. (4) there exists a reciprocal 1-system of wrenches $\hat{\underline{\boldsymbol{k}}}(q_{60})$ for which the reciprocity condition

$$\left[\,\boldsymbol{A}(q_{60})\ \hat{\underline{\boldsymbol{a}}}_6\,\right]^{\mathrm{T}} \Delta\, \hat{\underline{\boldsymbol{k}}}(q_{60}) = \boldsymbol{0} \tag{11}$$

holds. In the considered case of a 6H mechanism the six twist axes belong to a linear line complex with the main axis $\hat{\underline{\boldsymbol{k}}}$. For the global mobility it is sufficient that the global dimension of the configuration space is greater than zero [10]. Equivalently this means that every possible configuration in the neighbourhood of q_{60} has infinitesimal mobility. Thus the closure condition (8) must be valid for any q_6 near q_{60}

$$\hat{\underline{\boldsymbol{a}}}_6 = \boldsymbol{A}(q_6)\,\boldsymbol{\lambda}(q_6). \tag{12}$$

To prove this theorem Eq. (12) is locally approximated. The Taylor series expansion of the joint axes (twists) $\hat{\underline{\boldsymbol{a}}}_1, \ldots, \hat{\underline{\boldsymbol{a}}}_5$ in matrix $\boldsymbol{A}$ for the variable $q_6 = q_{60} + \delta q_6$ around the actual position q_{60} is given by

$$\boldsymbol{A}(q_6) = \boldsymbol{A}_0 + \frac{\boldsymbol{A}'_0}{1!}\,\delta q_6 + \frac{\boldsymbol{A}''_0}{2!}\,\delta q_6^2 + \frac{\boldsymbol{A}'''_0}{3!}\,\delta q_6^3 + \cdots \tag{13}$$

and for the ratios of the angular velocities

$$\boldsymbol{\lambda}(q_6) = \boldsymbol{\lambda}_0 + \frac{\boldsymbol{\lambda}'_0}{1!}\,\delta q_6 + \frac{\boldsymbol{\lambda}''_0}{2!}\,\delta q_6^2 + \frac{\boldsymbol{\lambda}'''_0}{3!}\,\delta q_6^3 + \cdots, \tag{14}$$

where all values at the actual position q_{60} are expressed according to $\boldsymbol{A}|_{q_{60}} = \boldsymbol{A}_0$. Inserting Eqs. (13) and (14) into the closure condition (12) yields

$$\hat{\underline{\boldsymbol{a}}}_6 = \left(\boldsymbol{A}_0 + \frac{\boldsymbol{A}_0'}{1!}\,\delta q_6 + \frac{\boldsymbol{A}_0''}{2!}\,\delta q_6^2 + \cdots \right) \left(\boldsymbol{\lambda}_0 + \frac{\boldsymbol{\lambda}_0'}{1!}\,\delta q_6 + \frac{\boldsymbol{\lambda}_0''}{2!}\,\delta q_6^2 + \cdots \right). \quad (15)$$

Carrying out the multiplications in Eq. (15) leads to a polynomial in δq_6. As Eq. (15) must hold for arbitrary values of δq_6 the comparison of coefficients of the powers of δq_6 yields independent conditions. The first conditions are

$$\begin{aligned}
\text{coeff. of } \delta q_6^0 : \quad & \hat{\underline{\boldsymbol{a}}}_6 = \boldsymbol{A}_0\,\boldsymbol{\lambda}_0 && (16)\\
\text{coeff. of } \delta q_6^1 : \quad & \boldsymbol{0} = \boldsymbol{A}_0'\,\boldsymbol{\lambda}_0 + \boldsymbol{A}_0\,\boldsymbol{\lambda}_0', && (17)\\
\text{coeff. of } \delta q_6^2 : \quad & \boldsymbol{0} = \frac{1}{2}\left(\boldsymbol{A}_0''\,\boldsymbol{\lambda}_0 + 2\boldsymbol{A}_0'\,\boldsymbol{\lambda}_0' + \boldsymbol{A}_0\,\boldsymbol{\lambda}_0''\right). && (18)\\
& \vdots
\end{aligned}$$

These equations are the higher-order closure conditions at the actual position q_{60}. Since the derivatives of the screw axes in $\boldsymbol{A}_0$ can be expressed algebraically according to Eq. (10) the derivatives of $\boldsymbol{\lambda}$ with respect to q_6 at q_{60} can be determined. For example the first-order derivative of $\boldsymbol{\lambda}$ at q_{60} is obtained from the overdetermined inhomogeneous system of linear equations (17) in $\boldsymbol{\lambda}_0'$. If this system is solvable the solution can be written as

$$\boldsymbol{\lambda}_0' = -\boldsymbol{A}_0^{+}\,\boldsymbol{A}_0'\,\boldsymbol{\lambda}_0. \quad (19)$$

If Eq. (17) is not solvable the Moore–Penrose-pseudoinverse $\boldsymbol{A}_0^{+}$ minimizes the least squares of the residuals of the system of equations.

Additionally the reciprocity condition (11) must hold for the Taylor series expansion of the closure condition (15)

$$0 = \boldsymbol{\lambda}_0^{\mathrm{T}}\boldsymbol{A}_0'^{\mathrm{T}}\,\Delta\,\hat{\underline{\boldsymbol{k}}}_0\,\delta q_6 + \frac{1}{2}\left(\boldsymbol{\lambda}_0^{\mathrm{T}}\boldsymbol{A}_0''^{\mathrm{T}} + 2\boldsymbol{\lambda}_0'^{\mathrm{T}}\boldsymbol{A}_0'^{\mathrm{T}}\right)\Delta\,\hat{\underline{\boldsymbol{k}}}_0\,\delta q_6^2 + \cdots. \quad (20)$$

The individually vanishing coefficients of the polynomial in δq_6 are the following implicit scalar equations,

$$\begin{aligned}
0 &= \boldsymbol{\lambda}_0^{\mathrm{T}}\boldsymbol{A}_0'^{\mathrm{T}}\,\Delta\,\hat{\underline{\boldsymbol{k}}}_0,\\
0 &= \left(2\boldsymbol{\lambda}_0'^{\mathrm{T}}\boldsymbol{A}_0'^{\mathrm{T}} + \boldsymbol{\lambda}_0^{\mathrm{T}}\boldsymbol{A}_0''^{\mathrm{T}}\right)\Delta\,\hat{\underline{\boldsymbol{k}}}_0, \qquad (21)\\
&\vdots\ .
\end{aligned}$$

With the derivatives of $\boldsymbol{\lambda}$ at q_{60} from the higher order closure conditions (17), (18), etc., the conditions (21) can be converted into

$$\begin{aligned}
0 &= \boldsymbol{\lambda}_0^{\mathrm{T}}\boldsymbol{A}_0'^{\mathrm{T}}\,\Delta\,\hat{\underline{\boldsymbol{k}}}_0 && \equiv g_1\left(\hat{\underline{\boldsymbol{k}}}_0, \boldsymbol{\lambda}_0, \boldsymbol{A}_0'\right)\\
0 &= \boldsymbol{\lambda}_0^{\mathrm{T}}\left(-2\boldsymbol{A}_0'^{\mathrm{T}}\boldsymbol{A}_0^{+\mathrm{T}}\boldsymbol{A}_0'^{\mathrm{T}} + \boldsymbol{A}_0''^{\mathrm{T}}\right)\Delta\,\hat{\underline{\boldsymbol{k}}}_0 && \equiv g_2\left(\hat{\underline{\boldsymbol{k}}}_0, \boldsymbol{A}_0^{+}, \boldsymbol{\lambda}_0, \boldsymbol{A}_0', \boldsymbol{\lambda}_0', \boldsymbol{A}_0''\right) \qquad (22)\\
&\vdots\\
0 &= \ldots && \equiv g_m\left(\hat{\underline{\boldsymbol{k}}}_0, \boldsymbol{A}_0^{+}, \ldots, \boldsymbol{\lambda}_0^{(m-1)}, \boldsymbol{A}_0^{(m)}\right),
\end{aligned}$$

with the order m of the Taylor series expansion which corresponds to the $\kappa = (m+1)$th order closure condition of the mechanism. These are m implicit conditions for the determination of the six screw axes $\hat{\underline{\boldsymbol{a}}}_{i0}$, $i = 1, \ldots, 6$.

It can be shown that the same results are obtained from the approximations of the reciprocity condition (4) around the actual position q_{60}. In every way it is necessary to fulfil the higher-order closure conditions.

But which order κ of fulfilled closure conditions is sufficient for the finite mobility of the mechanism? For the local analysis of overconstrained mechanisms this question has been answered [11]: Sufficient for the local analysis is a Taylor series expansion of the closure conditions with a maximal order $\kappa_{\max} = 4$.

5 Numerical Synthesis Algorithm

The numerical search algorithm is described for the example of a 6H mechanism. The five linear independent joint screws are each given by their six screw coordinates $\hat{\underline{\boldsymbol{a}}}_{i0} = [\, a_{ix0}\ a_{iy0}\ a_{iz0}\ a_{\varepsilon ix0}\ a_{\varepsilon iy0}\ a_{\varepsilon iz0}\,]^{\mathrm{T}}$, $i = 1, \ldots, 5$. Without loss of generality the screw $\hat{\underline{\boldsymbol{a}}}_{10}$ coincides with the x-axis of the reference coordinate frame, thus $\hat{\underline{\boldsymbol{a}}}_{10} = [\,1\ 0\ 0\ h_1\ 0\ 0\,]^{\mathrm{T}}$. With five independent screw coordinates for each of the screws $\hat{\underline{\boldsymbol{a}}}_{20}, \ldots, \hat{\underline{\boldsymbol{a}}}_{50}$ there are altogether 21 unknowns. Additional unknowns are the five ratios of the rotational velocities $\boldsymbol{\lambda}_0$. For these altogether 26 unknowns randomized start values are generated. With the five screws $\hat{\underline{\boldsymbol{a}}}_{10}, \ldots, \hat{\underline{\boldsymbol{a}}}_{50}$ and $\boldsymbol{\lambda}_0$ the first-order loop closure conditions (12) yield a screw $\hat{\underline{\boldsymbol{a}}}_{60} = \boldsymbol{A}_0\,\boldsymbol{\lambda}_0$. The next steps are listed in the following scheme:

$$
\begin{array}{lll}
\rightarrow & \boldsymbol{A}_0 = \left[\hat{\underline{\boldsymbol{a}}}_{10}\ \hat{\underline{\boldsymbol{a}}}_{20}\ \hat{\underline{\boldsymbol{a}}}_{30}\ \hat{\underline{\boldsymbol{a}}}_{40}\ \hat{\underline{\boldsymbol{a}}}_{50}\right] & \text{initialization of } \boldsymbol{A}_0 \\
\rightarrow & \hat{\underline{\boldsymbol{k}}}_0 = \Delta\ \mathrm{null}\left(\boldsymbol{A}_0^{\mathrm{T}}\right) & \text{reciprocity condition} \\
\rightarrow & \boldsymbol{A}_0^{+} = \left(\boldsymbol{A}_0^{\mathrm{T}}\boldsymbol{A}_0\right)^{-1}\boldsymbol{A}_0^{\mathrm{T}} & \text{pseudo inverse} \\
\rightarrow & \boldsymbol{A}_0' = \left[\hat{\underline{\boldsymbol{a}}}_{10}'\ \hat{\underline{\boldsymbol{a}}}_{20}'\ \hat{\underline{\boldsymbol{a}}}_{30}'\ \hat{\underline{\boldsymbol{a}}}_{40}'\ \hat{\underline{\boldsymbol{a}}}_{50}'\right] & \text{differential screw displ.} \\
\Rightarrow & g_1\left(\hat{\underline{\boldsymbol{k}}}_0, \boldsymbol{\lambda}_0, \boldsymbol{A}_0'\right) & \text{1st implicit cond. residual} \\
\rightarrow & \boldsymbol{\lambda}_0' = -\boldsymbol{A}_0^{+}\boldsymbol{A}_0'\,\boldsymbol{\lambda}_0 & \\
\rightarrow & \boldsymbol{A}_0'' = \frac{\partial \boldsymbol{A}_0'}{\partial q} + \sum_i \frac{\partial \boldsymbol{A}_0'}{\partial \lambda_i}\lambda_{i0}' & \\
\Rightarrow & g_2\left(\hat{\underline{\boldsymbol{k}}}_0, \boldsymbol{A}_0^{+}, \boldsymbol{\lambda}_0, \boldsymbol{A}_0', \boldsymbol{\lambda}_0', \boldsymbol{A}_0''\right) & \text{2nd implicit cond. residual} \\
\rightarrow & \vdots & \\
\rightarrow & \boldsymbol{\lambda}_0^{(m-1)} = \ldots & \\
\rightarrow & \boldsymbol{A}_0^{(m)} = \ldots & \\
\Rightarrow & g_m\left(\hat{\underline{\boldsymbol{k}}}_0, \boldsymbol{A}_0^{+}, \boldsymbol{\lambda}_0, \boldsymbol{A}_0', \ldots, \boldsymbol{\lambda}_0^{(m-1)}, \boldsymbol{A}_0^{(m)}\right) & m\text{th implicit cond. residual}
\end{array}
$$

The aim of the algorithm is to find numerical values for $\hat{\underline{\boldsymbol{a}}}_{10}$ to $\hat{\underline{\boldsymbol{a}}}_{50}$ and $\boldsymbol{\lambda}_0$ in such a way that the residuals $g_1, \ldots, g_m$ are zero. This is achieved by minimizing the norm of the residuals, whereby for example the Levenberg–Marquardt method is applied.

6 Results and Outlook

The described algorithm converges in over 80% of the randomized start values. With the algorithm any numerical configurations are found which represent movable nH mechanisms. As a next step the geometrical properties has to be derived from the numerical results for instance by using sensitivity analysis.

In the special case of revolute joints the empirical results have shown that the closure conditions up to the 2nd order (in sum three equations) are sufficient to find mobile 4R configurations. The closure conditions up to the 4th order (in sum six equations) are sufficient to find mobile 5R configurations. With the described algorithm the well known Bennett 4R and Goldberg 5R mechanisms have been already found.

References

1. Gogu, G.: Mobility of mechanisms: A critical review. *Mechanism and Machine Theory* **40**, 1068–1097 (2005).
2. Zhen, H., JingFang, L., DaXing, Z.: A general methodology for mobility analysis of mechanisms based on constraint screw theory. *Science in China Series E: Technological Sciences* **52**, 1337–1347 (2009).
3. Waldron, K.J.: The mobility of linkages. PhD thesis, Department of Mechanical Engineering, Stanford University (1969).
4. Mavroidis, C., Roth, B.: Analysis and synthesis of overconstrained mechanisms. In: *Proceedings of the 1994 ASME Design Technical Conferences*, Minneapolis (1994).
5. Jin, Q., Yang, T.: Overconstraint analysis on spatial 6-link loops. *Mechanism and Machine Theory* **37**, 267–278 (2002).
6. Martinez, J.M.R., Ravani, B.: On mobility analysis of linkages using group theory. *Transactions of the ASME* **125**, 70–80 (2003).
7. Dietmaier, P.: A new spatial 6R linkage. In: *Proceedings 9th World Congress IFToMM*, Milano, Vol. 1, pp. 52–56 (1995).
8. Hon-Cheung, Y., Baker, J.E.: On the generation of new linkages from Bennett loops. *Mechanism and Machine Theory* **16**, 473–485 (1981).
9. Hiller, M., Woernle, C.: A unified representation of spatial displacements. *Mechanism and Machine Theory* **19**, 477–486 (1984).
10. Selig, J.M.: *Geometric Fundamentals of Robotics*. Springer (2005).
11. Müller, A.: Geometric characterization of the configuration space of rigid body mechanisms in regular and singular points. In: *Proceedings of IDETC 2005 ASME 2005, International Design Engineering Technical Conferences*, Long Beach, California, USA, September 22–28 (2005).

Author Index

Subject Index