Jadran Lenarčič · Oussama Khatib
Editors

Advances in Robot Kinematics

Springer

Advances in Robot Kinematics

Jadran Lenarčič · Oussama Khatib
Editors

Advances in Robot Kinematics

Springer

Editors
Jadran Lenarčič
J. Stefan Institute
Ljubljana
Slovenia

Oussama Khatib
Department of Computer Science
Stanford University
Stanford, CA
USA

ISBN 978-3-319-06697-4 ISBN 978-3-319-06698-1 (eBook)
DOI 10.1007/978-3-319-06698-1
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014938472

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

As it relates to the essential property of robots, the motion of mechanisms, kinematics is the most fundamental aspect of robot design, analysis and control. The series of books on *Advances in Robot Kinematics* reports the latest achievements in this field. The first book in the series was published in 1991. Then, since 1994, these books have been published every 2 years, with the publication of each one being followed by a symposium in which the participants exchange their results and opinions. Books on *Advances in Robot Kinematics* have always been warmly accepted by experts and represent an important and ongoing information about what is happening in this area.

This book is the 12th in the series. All the articles contained within it have been selected on the basis of a peer-review process and describe the newest and most original achievements in the field of robot kinematics. We would like to emphasise that the whole process, from the submission of manuscripts, the reviewing, the selection of the articles, the various revisions, the preparation of the finished articles and the publication of the book, has taken less than 6 months.

Today, robot kinematics still presents an immense number of research challenges and the symposia on Advances in Robot Kinematics manage to bring together the best of the world's researchers and scientists. Since 1992 the symposia have come under the patronage of the International Federation for the Promotion of Mechanism and Machine Science—IFToMM.

The last symposium related with this book was organised by the J. Stefan Institute in Ljubljana, Slovenia. The 56 articles in this book cover the latest topics in the kinematics of robotic systems. We are grateful to the authors for their contributions and to the large team of reviewers for their critical and insightful recommendations. We are also indebted to Dr. Tadej Petrič from the J. Stefan Institute for his valuable technical contribution, and to the staff of Springer who were responsible for putting the whole book together.

Ljubljana, June 2014

Jadran Lenarčič
Oussama Khatib

Contents

Computing Cusps of 3R Robots Using Distance Geometry

Federico Thomas

Abstract The singularities of a 3R robot are usually determined, in terms of its joint angles, from the determinant of its Jacobian which can then be mapped onto the robot's workspace through its forward kinematics. The presence of cusps in these singularity plots permits to change robot's posture without meeting a singularity and hence their relevance. This chapter shows how, using Distance Geometry, the singularities in the workspace of a 3R robot can be represented as an octic curve of the form $4\delta_1\delta_3 - \delta_2^2 = 0$, where δ_i, $i = 1, 2, 3$, are quartic polynomials and, what is more important, its cusps correspond to those points in which $\delta_2 = \delta_3 = 0$. This leads to important simplifications over previous approaches.

Keywords 3R regional robot · Wrist-partitioned robots · Cuspidal robots · Distance geometry

1 Introduction

The presence of cusps in the singularity locus of a 3R robot allows it to change posture without meeting a singularity (see [1] and the references therein). Cusps correspond to points of the workspace where the robot's inverse kinematics admit three equal solutions, as it was first established in [2]. Then, the existence of cusps is usually determined from its fourth-degree inverse kinematics polynomial, though other alternatives are possible [3]. The symbolic condition, in terms of the DH parameters of the robot, for this polynomial to have three equal roots has been considered as intractable [1]. Only the case of orthogonal 3R robots—robots whose consecutive joint axes are mutually orthogonal—has been analyzed in detail [4]. This chapter

F. Thomas (✉)
Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Barcelona, Spain
e-mail: fthomas@iri.upc.edu

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_1,

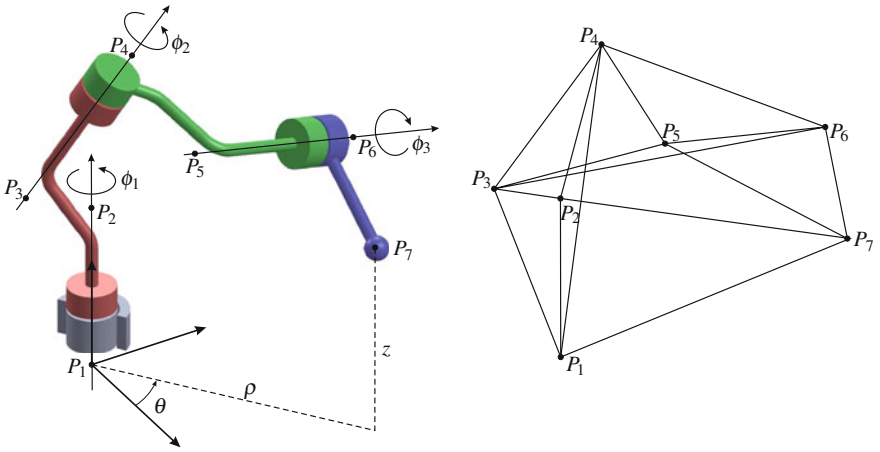


Fig. 1 A 3R regional robot (*left*) and its associated bar-and-joint framework (*right*)

puts forward a new formulation of the problem that could lead to new insights into the general case thanks to the simplicity and symmetry of the resulting algebraic expressions.

This chapter is organized as follows. Section 2 describes how, using Distance Geometry, the inverse kinematics of a 3R robot boils down to compute the intersection of two ellipses. Sections 3 and 4 show how the singularity locus of a 3R robot can be computed from the pencil of conics defined by these two ellipses, and how cusps in this locus correspond to osculating contacts between them. An example is presented in Sect. 5, and some conclusions are drawn in Sect. 6.

2 Distance-Based Formulation

A regional 3R robot is an open chain of four rigid bodies (the links), pairwise articulated through revolute joints [Fig. 1(left)]. The inverse kinematics problem is to find all valid configurations of this chain (ϕ_1, ϕ_2, and ϕ_3) that are compatible with a specified location for P_7, relative to a global reference frame. Then, the first and last links can be seen as articulated through a spherical joint centered at P_7 and the problem is equivalent to that of finding the valid configurations of a closed loop of four pairwise articulated links through three revolute joints and one spherical joint.

A link connecting two revolute axes can be modeled by taking two points on each of these axes, and by connecting them all with rigid bars to form a tetrahedron. A link connecting a revolute axis and a spherical joint can be modeled by taking two points on the revolute joint axis and the center of the spherical joint, and by connecting them all with rigid bars to form a triangle. In this way, a 3R regional robot can be modeled as the bar-and-joint framework shown in Fig. 1(right). In this

conversion we are loosing an important information: the orientation of the two tetrahedra. Nevertheless, given a valid configuration of the obtained bar-and-joint framework, we can derive another valid configuration with the desired orientations for the two tetrahedra. Indeed, observe that P_3, P_4, and P_7 defines a plane that divides the framework in two halves and the mirror projection of any of these two halves with respect to this plane changes the orientation of the corresponding tetrahedron while preserving the bar lengths. As a consequence, any valid configuration for the framework translates into a valid configuration for the 3R robot.

The distances between the set of points $\{P_1, P_2, P_3, P_4, P_7\}$ or $\{P_3, P_4, P_5, P_6, P_7\}$ are not independent because they are embedded in $\mathbb{R}^3$. This dependency, using the theory of Cayley-Menger determinants, translates into the following algebraic conditions:

$$\begin{vmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & s_{1,2} & s_{1,3} & s_{1,4} & s_{1,7} \\ 1 & s_{2,1} & 0 & s_{2,3} & s_{2,4} & s_{2,7} \\ 1 & s_{3,1} & s_{3,2} & 0 & s_{3,4} & s_{3,7} \\ 1 & s_{4,1} & s_{4,2} & s_{4,3} & 0 & s_{4,7} \\ 1 & s_{7,1} & s_{7,2} & s_{7,3} & s_{7,4} & 0 \end{vmatrix} = 0 \text{ and } \begin{vmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & s_{3,4} & s_{3,5} & s_{3,6} & s_{3,7} \\ 1 & s_{4,3} & 0 & s_{4,5} & s_{4,6} & s_{4,7} \\ 1 & s_{5,3} & s_{5,4} & 0 & s_{5,6} & s_{5,7} \\ 1 & s_{6,3} & s_{6,4} & s_{6,5} & 0 & s_{6,7} \\ 1 & s_{7,3} & s_{7,4} & s_{7,5} & s_{7,6} & 0 \end{vmatrix} = 0, \tag{1}$$

where $s_{i,j}$ stands for the squared distance between P_i and P_j. The above two equations are quadratic forms in the unknown distances $s_{3,7}$ and $s_{4,7}$. They actually represent two real ellipses, $\mathscr{A} : \mathbf{x}\mathbf{A}\mathbf{x}^T = 0$ and $\mathscr{B} : \mathbf{x}\mathbf{B}\mathbf{x}^T = 0$, where $\mathbf{x} = (s_{3,7}, s_{4,7}, 1)$ and

$$\mathbf{A} = \begin{pmatrix} a_1 & c_1 & d_1 \\ c_1 & b_1 & e_1 \\ d_1 & e_1 & f_1 \end{pmatrix} \text{ and } \mathbf{B} = \begin{pmatrix} a_2 & c_2 & d_2 \\ c_2 & b_2 & e_2 \\ d_2 & e_2 & f_2 \end{pmatrix}. \tag{2}$$

The entries d_1 and e_1 depend linearly on $s_{1,7}$ and $s_{2,7}$, while f_1 depends on them quadratically. All other entries of $\mathbf{A}$ and $\mathbf{B}$ are constant for a given robot and they can be expressed in terms of determinants of known bar lengths (see [5] for details).

Solving the inverse kinematics of a 3R robot consists in obtaining the sets of joint angles (ϕ_1, ϕ_2, ϕ_3) that provide the desired position of its end-effector given in cylindrical coordinates by (θ, ρ, z). Due to the symmetry with respect to the first joint axis, ϕ_1 is trivially obtained. The other two cylindrical coordinates, ρ and z, determine the squared distances $s_{1,7}$ and $s_{2,7}$. Then, we can compute the intersection of $\mathscr{A}$ and $\mathscr{B}$ to obtain up to four sets of distances $(s_{3,7}, s_{4,7})$compatible with all other distances. It is important to note that, given the orientation of the tetrahedra with vertex sets $\{P_1, P_2, P_3, P_4\}$ and $\{P_3, P_4, P_5, P_6\}$, every solution for $(s_{3,7}, s_{4,7})$ leads to a unique solution for (ϕ_2, ϕ_3). Thus, solving the inverse kinematics of a 3R regional robot reduces to calculate the intersections of two ellipses.

3 Singularities

A singularity occurs when we have a repeated solution of the inverse kinematics, that is, when $\mathscr{A}$ and $\mathscr{B}$ are tangent. The positional relationship between $\mathscr{A}$ and $\mathscr{B}$ can be derived from the study of the pencil of conics they define, that is, from the family of conics defined by $\mathbf{p}^T(\lambda\mathbf{A}+\mathbf{B})\mathbf{p} = 0, \lambda \in \mathbb{R}$ (see [6] for an introductory explanation). The values of λ for which a conic of this pencil is degenerate correspond to those in which

$$f(\lambda) = \det(\lambda\mathbf{A} + \mathbf{B}) = l_3\lambda^3 + 3l_2\lambda^2 + 3l_1\lambda + l_0 = 0, \tag{3}$$

where the coefficients l_i, $i = 0, 1, 2, 3$, can expressed in a neat and elegant way as [7, p. 191]:

$$l_3 = \begin{vmatrix} a_1 & c_1 & d_1 \\ c_1 & b_1 & e_1 \\ d_1 & e_1 & f_1 \end{vmatrix} = \det(\mathbf{A}), \tag{4}$$

$$3l_2 = \begin{vmatrix} a_2 & c_1 & d_1 \\ c_2 & b_1 & e_1 \\ d_2 & e_1 & f_1 \end{vmatrix} + \begin{vmatrix} a_1 & c_2 & d_1 \\ c_1 & b_2 & e_1 \\ d_1 & e_2 & f_1 \end{vmatrix} + \begin{vmatrix} a_1 & c_1 & d_2 \\ c_1 & b_1 & e_2 \\ d_1 & e_1 & f_2 \end{vmatrix}, \tag{5}$$

$$3l_1 = \begin{vmatrix} a_1 & c_2 & d_2 \\ c_1 & b_2 & e_2 \\ d_1 & e_2 & f_2 \end{vmatrix} + \begin{vmatrix} a_2 & c_1 & d_2 \\ c_2 & b_1 & e_2 \\ d_2 & e_1 & f_2 \end{vmatrix} + \begin{vmatrix} a_2 & c_2 & d_1 \\ c_2 & b_2 & e_1 \\ d_2 & e_2 & f_1 \end{vmatrix}, \tag{6}$$

$$l_0 = \begin{vmatrix} a_2 & c_2 & d_2 \\ c_2 & b_2 & e_2 \\ d_2 & e_2 & f_2 \end{vmatrix} = \det(\mathbf{B}). \tag{7}$$

The above polynomial in λ is known as the *generalized characteristic polynomial* of the pencil. By definition, $f(\lambda) = 0$ has a multiple root if, and only if, its discriminant, say Δ, vanishes. Furthermore, it can be shown that $f(\lambda) = 0$ has three simple real roots if $\Delta > 0$, and $f(\lambda) = 0$ has two complex conjugate roots and a real root if $\Delta < 0$. The roots of $\Delta = 0$ give information on the positional relationship between $\mathscr{A}$ and $\mathscr{B}$. Actually, the sign of the discriminant Δ gives information about the order of accessibility of the robot's workspace. It permits to decompose it in the following three regions:

- $\Delta < 0$ corresponds to a two-way accessible region;
- $\Delta = 0$ corresponds to the singularities of the robot; and
- $\Delta > 0$ corresponds to a four-way accessible region or an inaccessible region.

The standard expression of the discriminant of a cubic gives little insight into the structure of our problem. Fortunately, a much more convenient expression, that will reveal very important for the computation of cusps, can be found in [8]. It reads as follows:

$$\Delta = \begin{vmatrix} 2\delta_1 & \delta_2 \\ \delta_2 & 2\delta_3 \end{vmatrix} \tag{8}$$

where

$$\delta_1 = \begin{vmatrix} l_3 & l_2 \\ l_2 & l_1 \end{vmatrix}, \ \delta_2 = \begin{vmatrix} l_3 & l_1 \\ l_2 & l_0 \end{vmatrix}, \ \delta_3 = \begin{vmatrix} l_2 & l_1 \\ l_1 & l_0 \end{vmatrix} \tag{9}$$

Observe that the condition of singularity $\Delta = 0$ is expressed as a determinant of determinants of determinants of determinants (four levels of nested determinants). It can be shown that the elements of the third level of determinants depend quadratically on $s_{3,7}$ and $s_{4,7}$. Then, since the two outer levels of determinants are quadratic with respect to their elements, the singularity locus can be displayed as a curve of order 2^3 in the plane defined by $s_{3,7}$ and $s_{4,7}$.

4 Cusps As Osculating Contacts

If three of the points of intersection between $\mathscr{A}$ and $\mathscr{B}$ coincide, the ellipses are said to osculate each other at this point. In this case, the characteristic polynomial of the pencil they define is a perfect cube [9]. As a consequence,

$$\frac{l_3}{l_2} = \frac{l_2}{l_1} = \frac{l_1}{l_0} = -\frac{1}{r}, \tag{10}$$

where r is the triple root of the characteristic polynomial. Then, observe that

$$\frac{l_3}{l_2} = \frac{l_2}{l_1} \Rightarrow \delta_1 = 0, \ \ \frac{l_3}{l_2} = \frac{l_1}{l_0} \Rightarrow \delta_2 = 0, \ \ \text{and} \ \ \frac{l_2}{l_1} = \frac{l_1}{l_0} \Rightarrow \delta_3 = 0.$$

In other words, in a cusp not only $\Delta = 0$, but all three components of the discriminant vanish. Moreover,

$$\left.\begin{matrix} \delta_1 = 0 \\ \delta_2 = 0 \end{matrix}\right\} \Rightarrow \Delta = 0, \ \text{and} \ \delta_3 = 0 \ \text{iff} \ \frac{l_2}{l_3} \neq \frac{0}{0},$$
$$\left.\begin{matrix} \delta_2 = 0 \\ \delta_3 = 0 \end{matrix}\right\} \Rightarrow \Delta = 0, \ \text{and} \ \delta_1 = 0 \ \text{iff} \ \frac{l_0}{l_1} \neq \frac{0}{0}.$$

Since $l_0 = \det(\mathbf{B}) \neq 0$ by construction, we conclude that we have a cusp if, and only if, $\delta_2 = 0$ and $\delta_3 = 0$.

5 Example

Let us considered the orthogonal 3R robot with the standard DH parameters $d_1 = 1$, $d_2 = 1/2, d_3 = 1, a_1 = 1, a_2 = 4/5$ and $a_3 = 2$. Then, by locating P_1 and P_7 at the origin of the reference frame and at the center of the end-effector, respectively; P_2 and P_3 on the common normal between the first and the second axis; P_4 and P_5 on the

common normal between the second and the third revolute axis; and P_6 as the nearest point to P_7 on the third revolute axis, we have that $s_{1,2} = d_1^2$, $s_{2,3} = a_1^2$, $s_{3,4} = d_2^2$, $s_{4,5} = a_2^2$, $s_{5,6} = d_3^2$, $s_{6,7} = a_3^2$. Then, by Pythagoras' theorem, $s_{1,3} = s_{1,2} + s_{2,3}$, $s_{2,4} = s_{2,3} + s_{3,4}$, $s_{3,5} = s_{3,4} + s_{4,5}$, $s_{4,6} = s_{4,5} + s_{5,6}$, $s_{5,7} = s_{5,6} + s_{6,7}$. Moreover, since $\alpha_2 = \pi/2$ and $\alpha_3 = \pi/2$, $s_{1,4} = s_{1,3} + s_{3,4}$, $s_{3,6} = s_{3,5} + s_{5,6}$.

Substituting the above squared distances in the expressions for $\mathscr{A}$ and $\mathscr{B}$, we obtain:

$$\begin{pmatrix} s_{3,7} & s_{4,7} & 1 \end{pmatrix} \begin{pmatrix} 5 & -4 & d \\ -4 & 4 & e \\ d & e & f \end{pmatrix} \begin{pmatrix} s_{3,7} \\ s_{4,7} \\ 1 \end{pmatrix} = 0, \tag{11}$$

and

$$\begin{pmatrix} s_{3,7} & s_{4,7} & 1 \end{pmatrix} \begin{pmatrix} 2.56 & -2.56 & -0.64 \\ -2.56 & 3.56 & -5 \\ -0.64 & -5 & 21.7296 \end{pmatrix} \begin{pmatrix} s_{3,7} \\ s_{4,7} \\ 1 \end{pmatrix} = 0, \tag{12}$$

respectively, where d, e, and f depend on $s_{1,7}$ and $s_{2,7}$ as follows:

$$\begin{pmatrix} d \\ e \end{pmatrix} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} s_{1,7} \\ s_{2,7} \\ 1 \end{pmatrix} \quad \text{and} \quad f = \begin{pmatrix} s_{1,7} & s_{2,7} & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 & -1 \\ -1 & 2 & 0 \\ -1 & 0 & 2.25 \end{pmatrix} \begin{pmatrix} s_{1,7} \\ s_{2,7} \\ 1 \end{pmatrix}.$$

Now, substituting in (4)–(7), we obtain:

$$\begin{aligned}
l_0 &= -\frac{16384}{625} \\
l_1 &= \frac{18848}{1875} d + \frac{64}{75} f + \frac{263116}{46875} \\
&= \frac{64}{75} s_{1,7}^2 - \frac{128}{75} s_{1,7} s_{2,7} - \frac{128}{75} s_{1,7} + \frac{128}{75} s_{2,7}^2 - \frac{18848}{1875} s_{2,7} - \frac{173116}{46875} \\
l_2 &= \frac{1256}{75} d + \frac{63}{25} f - \frac{89}{75} d^2 + \frac{18274}{1875} \\
&= \frac{63}{25} s_{1,7}^2 - \frac{126}{25} s_{1,7} s_{2,7} - \frac{126}{25} s_{1,7} + \frac{289}{75} s_{2,7}^2 - \frac{1256}{75} s_{2,7} + \frac{115621}{7500} \\
l_3 &= -4d^2 + 8d + 4f - 5 \\
&= 4s_{1,7}^2 - 8s_{1,7} s_{2,7} - 8s_{1,7} + 4s_{2,7}^2 - 8s_{2,7} + 4
\end{aligned}$$

Then, substituting in (9), we finally obtain:

$$\begin{aligned}
\delta_1 = {} & \frac{4814}{1875} d^2 f - \frac{1184996}{15625} f - \frac{70064}{1875} df - \frac{59287072}{140625} d - \frac{7239812}{46875} d^2 \\
& - \frac{2608}{5625} d^3 - \frac{7921}{5625} d^4 - \frac{5507}{1875} f^2 - \frac{235270576}{3515625} \\
\delta_2 = {} & \frac{5696}{5625} d^2 f - \frac{348149476}{3515625} f - \frac{5571872}{140625} df - \frac{250411552}{1171875} d - \frac{82201508}{1171875} d^2 \\
& + \frac{1677472}{140625} d^3 - \frac{1344}{625} f^2 + \frac{16328181784}{87890625}
\end{aligned}$$

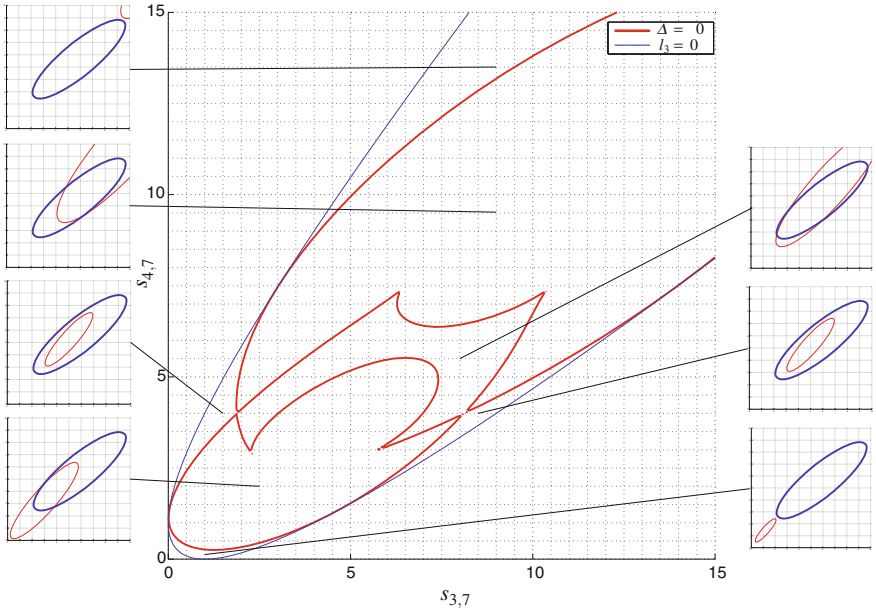


Fig. 2 The singularity locus ($\Delta = 0$) divides the area where $l_3 < 0$ (the region where $\mathscr{A}$ is a real ellipse) into regions where the spatial relationship between $\mathscr{A}$ and $\mathscr{B}$ is the same

$$\delta_3 = -\frac{28665899264}{87890625}d - \frac{198564352}{3515625}f - \frac{2412544}{140625}df - \frac{245883904}{3515625}d^2 - \frac{4096}{5625}f^2 - \frac{630607309456}{2197265625}$$

The robot's singularity locus is given by $\Delta = 4\delta_1\delta_3 - \delta_2^2 = 0$. This locus segments the plane define by $(s_{3,7}, s_{4,7})$ into regions with the same number of inverse kinematics solutions (Fig. 2). It can be mapped onto the robot's workspace by observing that $z = (d_{1,2}^2 - s_{2,7} + s_{1,7})/2d_{1,2}$ and $\rho = +\sqrt{s_{2,7} - (d_{1,2} - z)^2}$, where $d_{1,2} = \sqrt{s_{1,2}} = 1$. The result is represented in Fig. 3.

The curves defined by $\delta_2 = 0$ and $\delta_3 = 0$ are plotted in Fig. 4. Observe how their intersection coincide with the cusps of the singularity locus plotted in Fig. 2. The system of equations $\delta_2 = \delta_3 = 0$ has 12 solutions, 8 of them are complex. The real solutions correspond to the coordinates of the 4 cusps: (5.4364, 2.8669), (10.4444, 7.4444), (2.2975, 2.8669), and (6.4444, 7.4444).

We have analyzed an orthogonal 3R robot and this induces some simplifications in the derived equations. For example, δ_3 is cubic with respect to $s_{1,7}$ and $s_{2,7}$ instead of quartic (as is the case in general). This leads to 12 solutions for the system $\delta_2 = \delta_3 = 0$ while, in general, we would have 16.

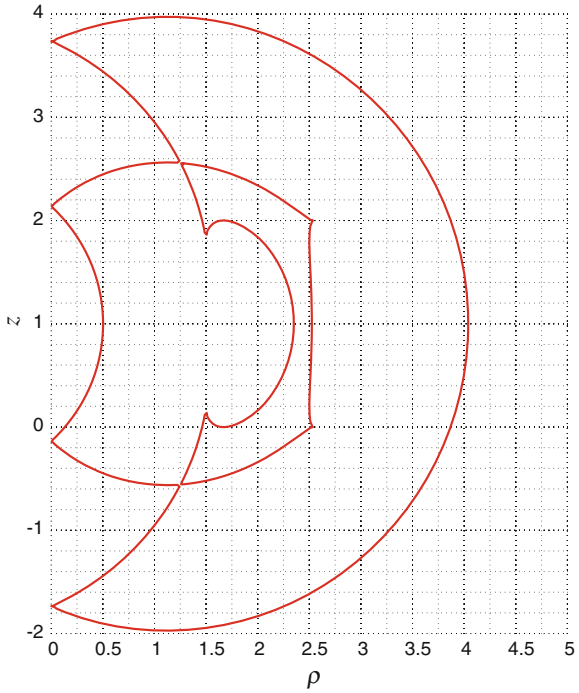


Fig. 3 The singularity locus represented in Fig. 2 mapped onto the robot's workspace. Both plots are in one-to-one correspondence

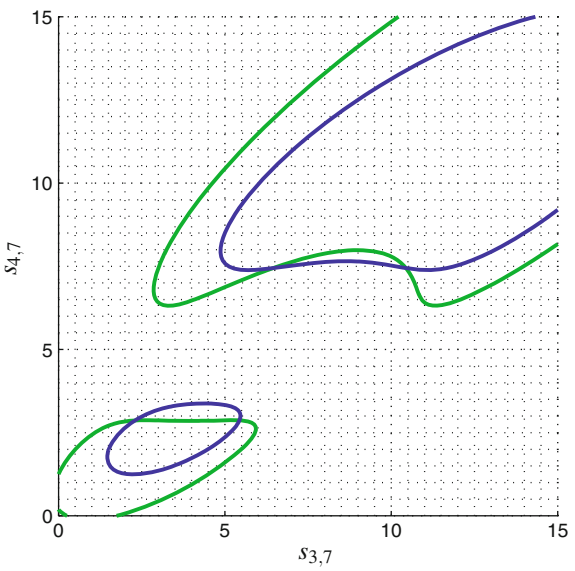


Fig. 4 Plot of $\delta_2 = 0$ (*green*), and $\delta_3 = 0$ (*blue*)

6 Conclusion

We have shown how the kinematics of a 3R robot is determined by the positional relationship between two ellipses, and how cusps in the singularity locus correspond to osculating contacts between these two ellipses. This geometric interpretation translates into an algebraic characterization of cusps as the intersection of two quartic curves, a simpler characterization than the traditional one based on imposing three equal roots to the inverse kinematics polynomial.

References

1. Wenger, P.: Cuspidal and noncuspidal robot manipulators. Robotica **25**(6), 677–689 (2007)
2. El Omri, J., Wenger, P.: How to recognize simply a non-singular posture changing 3-DOF manipulator. In: Proceedings of 7th ICAR, pp. 215–222 (1995)
3. Husty, M., Ottaviano, E., and Ceccarelli, M.: A geometrical characterization of workspace singularities in 3R manipulators. In: Lenarčič, J., Wenger, P. (eds.) Advances in Robot Kinematics: Analysis and Design, pp. 411–418. Springer, New York (2008)
4. Baili, M., Wenger, P., Chablat, D.: A classification of 3R orthogonal manipulators by the topology of their workspace. In: IEEE International Conference on Robotics and Automation (2004)
5. Thomas, F., Re-examining the singularity loci of 3R manipulators (in preparation)
6. Choi, Y.-K., Wang, W., Liu, Y., Kim, M.-S.: Continuous collision detection for two moving elliptic disks. IEEE Trans. Robot. **22**(2), 213–224 (2006)
7. Richter-Gebert, J.: Perspectives on Projective Geometry: A Guided Tour Through Real and Complex Geometry. Springer, New York (2011)
8. Blinn, J.F.: Polynomial discriminants i. matrix magic. IEEE Comput. Graph. Appl. **20**(6), 94–98 (2002)
9. Salmon, G.: A Treatise on Conic Sections. Chelsea Publishing Co., New York (1869)

Kinematic Mapping of SE(4) and the Hypersphere Condition

Georg Nawratil

Abstract In this chapter we present a novel kinematic mapping for the Euclidean 4-space E^4. We show that there is a bijection between the group SE(4) of Euclidean displacements of E^4 and points on a quadric (sliced along a 3-dimensional generator-space), which is located in a projective 11-dimensional space. These 12 homogeneous motion parameters can be seen as a natural extension of the Blaschke-Grünwald parameters for E^2 and the Study parameters for E^3, respectively. In addition we also study the constraint that a point is located on a hypersphere of E^4. We prove that this hypersphere condition is a homogeneous quadratic equation in the 12 homogeneous motion parameters.

Keywords Kinematic mapping · Euclidean 4-space · Quaternion · Hypersphere

1 Motivation

The study of displacements of the Euclidean 4-space is motivated by Stewart Gough (SG) platforms. These are 6-dof $S_3\underline{P}S_3$ parallel manipulators, as the platform is connected with the base via six $S_3\underline{P}S_3$-legs, where $\underline{P}$ denotes an active prismatic joint and S_3 a passive spherical[1] one. If the centers of the S_3-joints located at the base (resp. platform) are coplanar, then the base (resp. platform) is called planar. A SG manipulator with planar platform and planar base is called planar SG platform. These manipulators are geometrically a lot better understood than their non-planar counterparts (e.g. attachment of additional legs without changing the direct

[1] S_n denote the spherical joint, which admits the group of spherical motions SO(n) of E^n. Note that a S_2-joint equals a rotational joint (R-joint).

G. Nawratil (✉)
Institute of Discrete Mathematics and Geometry, Vienna University of Technology, Vienna, Austria
e-mail: nawratil@geometrie.tuwien.ac.at

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_2,

kinematics [1] and singularity set [2], self-motions [3] and Duporcq's theorem [4], ...).

We hope to gain a deeper geometric insight into the nature of non-planar SG platforms by studying the analogs of planar SG platforms in E^4, which are so-called hyperplanar 10-dof $S_4\underline{P}S_4$ parallel manipulators.[2]

The basic equation for an algebraic kinematical study of 10-dof $S_4\underline{P}S_4$ parallel manipulators is the so-called hypersphere condition which means that the center of the platform S_4-joint is located on a hypersphere centered in the corresponding base S_4-joint. For the formulation of this equation, we need a proper kinematic mapping of SE(4), which is given in Sect. 2. Based on the presented kinematic mapping, we study the hypersphere condition and its derivative (infinitesimal direct kinematics) in Sect. 3. Finally we close the chapter with conclusions and an outlook.

2 Kinematic Mappings of SE(n)

A kinematic mapping of SE(n) is a bijective mapping between the group of displacements of E^n and a set $\mathscr{S}$ of points in a certain space. Well known examples of these mappings are the one of Blaschke [5] and Grünwald [6] for E^2 and the one of Study [7] for E^3, which are reviewed within the next two subsections.

2.1 Study Mapping of SE(3)

$\mathfrak{Q} := q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$ with $q_0, \ldots, q_3 \in \mathbb{R}$ is an element of the skew field of quaternions $\mathbb{H}$, where $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are the so-called quaternion units. The conjugated quaternion to $\mathfrak{Q}$ is given by $\widetilde{\mathfrak{Q}} := q_0 - q_1\mathbf{i} - q_2\mathbf{j} - q_3\mathbf{k}$. Moreover $\mathfrak{Q}$ is called a pure quaternion for $q_0 = 0$ and a unit quaternion for $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$. Finally we can embed the points X of E^3 with Cartesian coordinates (x_1, x_2, x_3) into the set of pure quaternions by the following mapping:

$$\iota_3 : \mathbb{R}^3 \to \mathbb{H} \quad \text{with} \quad (x_1, x_2, x_3) \mapsto \mathfrak{X} := x_1\mathbf{i} + x_2\mathbf{j} + x_3\mathbf{k}. \tag{1}$$

Classically the Study mapping is introduced by the usage of dual quaternions $\mathbb{H} + \varepsilon\mathbb{H}$, where ε is the dual unit with the property $\varepsilon^2 = 0$. An element $\mathfrak{E} + \varepsilon\mathfrak{T}$ of $\mathbb{H} + \varepsilon\mathbb{H}$ is called unit dual quaternion if $\mathfrak{E}$ is an unit quaternion and following condition holds:

$$e_0t_0 + e_1t_1 + e_2t_2 + e_3t_3 = 0. \tag{2}$$

[2] As there are 10 dofs in E^4 (six rotational ones and four translatory dofs), the hyperplanar platform (moving 3-space) and the hyperplanar base (fixed 3-space) have to be connected via ten $S_4\underline{P}S_4$-legs (cf. footnote 1). In this context it should be noted that the lower-dimensional counterparts of planar SG platforms are 3-dof $R\underline{P}R$ parallel manipulators with collinear base points and platform points.

Based on the usage of unit dual quaternions $\mathfrak{E} + \varepsilon\mathfrak{T}$ it can be shown (e.g. Sect. 3.3.2.2 of [8]) that the mapping of points $\mathsf{X} \in E^3$ to $\mathsf{X}' \in E^3$ induced by any element of SE(3), can be written as follows (by using ι_3):

$$\mathfrak{X} \mapsto \mathfrak{X}' \quad \text{with} \quad \mathfrak{X}' := \mathfrak{E} \circ \mathfrak{X} \circ \widetilde{\mathfrak{E}} + (\mathfrak{T} \circ \widetilde{\mathfrak{E}} - \mathfrak{E} \circ \widetilde{\mathfrak{T}}), \tag{3}$$

where $\circ$ denotes the well-known quaternion multiplication. Moreover it can be shown that the mapping of Eq. (3) is an element of SE(3) for any unit dual quaternion $\mathfrak{E} + \varepsilon\mathfrak{T}$. Note that $\mathfrak{X}'$ is again a pure quaternion, where the first summand $\mathfrak{E} \circ \mathfrak{X} \circ \widetilde{\mathfrak{E}}$ is the rotational component, which can be written in vector-representation as $(x_1', x_2', x_3')^T = \mathbf{R}_3(x_1, x_2, x_3)^T$ with

$$\mathbf{R}_3 = \begin{pmatrix} e_0^2 + e_1^2 - e_2^2 - e_3^2 & 2(e_1e_2 - e_0e_3) & 2(e_1e_3 + e_0e_2) \\ 2(e_1e_2 + e_0e_3) & e_0^2 - e_1^2 + e_2^2 - e_3^2 & 2(e_2e_3 - e_0e_1) \\ 2(e_1e_3 - e_0e_2) & 2(e_2e_3 + e_0e_1) & e_0^2 - e_1^2 - e_2^2 + e_3^2 \end{pmatrix}, \tag{4}$$

where $det\mathbf{R}_3 = (e_0^2 + e_1^2 + e_2^2 + e_3^2)^3 = 1$ holds. As the remaining part of Eq. (3) does not depend on X, it corresponds to a translation with vector $\mathbf{s}_3 := (s_1, s_2, s_3)^T$ and

$$\begin{aligned} s_1 &= 2(e_0t_1 - e_1t_0 + e_2t_3 - e_3t_2), \quad s_2 = 2(e_0t_2 - e_1t_3 - e_2t_0 + e_3t_1), \\ s_3 &= 2(e_0t_3 + e_1t_2 - e_2t_1 - e_3t_0). \end{aligned} \tag{5}$$

As both unit dual quaternions $\pm(\mathfrak{E} + \varepsilon\mathfrak{T})$ correspond to the same Euclidean motion of E^3, we consider the homogeneous 8-tuple $(e_0 : \ldots : e_3 : t_0 : \ldots : t_3)$. These so-called Study parameters can be interpreted as a point of a projective 7-dimensional space P^7. Therefore there is a bijection between SE(3) and all real points $\mathscr{S}$ of P^7 located on the so-called Study quadric $\Phi \subset P^7$, which is given by Eq. (2) ($\Rightarrow$ the signature of Φ is $(4_+, 4_-, 0_0)$) and is sliced along the 3-dimensional generator-space $e_0 = e_1 = e_2 = e_3 = 0$, as the corresponding quaternion $\mathfrak{E}$ cannot be normalized.

2.2 Blaschke-Grünwald Mapping of SE(2)

The Blaschke-Grünwald mapping can be obtained from the Study mapping by restricting ourselves to planar Euclidean displacements within a plane α, which corresponds to a 3-dimensional generator-space of Φ. If we choose α as the plane given by $x_1 = 0$, it can easily be seen (cf. Remark 3.38 of [8]) that the corresponding generator-space of Φ is given by $e_2 = e_3 = t_0 = t_1 = 0$. Therefore there is a bijection between SE(2) and all real points $(e_0 : e_1 : t_2 : t_3)$ of a projective 3-dimensional space P^3, with exception of the points located on the line $e_0 = e_1 = 0$.

The vector-representation of planar displacements in dependency of the Blaschke-Grünwald parameters $(e_0 : e_1 : t_2 : t_3)$ can immediately be obtained from Eqs. (4)

and (5) and reads as $(x_2', x_3')^T = \mathbf{R}_2(x_2, x_3)^T + \mathbf{s}_2$ with:

$$\mathbf{R}_2 = \begin{pmatrix} e_0^2 - e_1^2 & -2e_0e_1 \\ 2e_0e_1 & e_0^2 - e_1^2 \end{pmatrix}, \qquad \mathbf{s}_2 = \begin{pmatrix} 2(e_0t_2 - e_1t_3) \\ 2(e_0t_3 + e_1t_2) \end{pmatrix}, \tag{6}$$

where $det\mathbf{R}_2 = (e_0^2 + e_1^2)^2 = 1$ holds.

2.3 Kinematic Mappings of SE(4)

Until now the author is only aware of one kinematic mapping of SE(4), which was given by Klawitter and Hagemann [9]. They presented an unified concept based on Clifford algebras, for constructing kinematic mappings for certain Cayley-Klein geometries. Especially for E^2 and E^3, they demonstrated that their approach yields the Blaschke-Grünwald mapping and the Study mapping, respectively.

According to Sects. 6.3 and 7 of [9], displacements of SE(4) are mapped onto points of a real 15-dimensional projective space P^{15}, which are located in the intersection of nine quadrics R_i $(i = 1, \ldots, 9)$ sliced along the quadric N_1. For the explicit equations of $R_1, \ldots, R_9$ and N_1 we refer to Sect. 7 of [9].

Due to the large number of homogeneous motion parameters as well as the resulting set of quadratic constraints, the Klawitter-Hagemann mapping is not suited for performing computational algebraic kinematics in E^4. Therefore we are interested in a simplified kinematic mapping of SE(4), which is constructed next.

We start by embedding the points X of E^4 with Cartesian coordinates (x_0, x_1, x_2, x_3) into the set of quaternions by the mapping:

$$\iota_4 : \mathbb{R}^4 \to \mathbb{H} \quad \text{with} \quad (x_0, x_1, x_2, x_3) \mapsto \mathfrak{X} := x_0 + x_1\mathbf{i} + x_2\mathbf{j} + x_3\mathbf{k}. \tag{7}$$

Moreover we need the classical quaternion representation theorem for SO(4), which has many fathers (Euler, Cayley, Salmon, Elfrinkhof, Stringham, Bouman) according to Mebius [10] and states the following:

Theorem 1 *The mapping of points $\mathsf{X} \in E^4$ to $\mathsf{X}' \in E^4$ induced by any element of SO(4), can be written as follows (by using ι_4):*

$$\mathfrak{X} \mapsto \mathfrak{X}' \quad \textit{with} \quad \mathfrak{X}' := \mathfrak{E} \circ \mathfrak{X} \circ \mathfrak{F}, \tag{8}$$

where $\mathfrak{E}$ and $\mathfrak{F}$ is a pair of unit quaternions, which is determined uniquely up to the sign. Moreover the mapping of Eq. (8) is an element of SO(4) for any pair of unit quaternions $\mathfrak{E}$ and $\mathfrak{F}$.

Direct computation shows that the mapping given in Eq. (8) can be written in vector-representation as $(x_0', x_1', x_2', x_3')^T = \mathbf{R}_4(x_0, x_1, x_2, x_3)^T$ with $\mathbf{R}_4 = \mathbf{EF}$ and

$$\mathbf{E} = \begin{pmatrix} e_0 & -e_1 & -e_2 & -e_3 \\ e_1 & e_0 & -e_3 & e_2 \\ e_2 & e_3 & e_0 & -e_1 \\ e_3 & -e_2 & e_1 & e_0 \end{pmatrix}, \qquad \mathbf{F} = \begin{pmatrix} f_0 & -f_1 & -f_2 & -f_3 \\ f_1 & f_0 & f_3 & -f_2 \\ f_2 & -f_3 & f_0 & f_1 \\ f_3 & f_2 & -f_1 & f_0 \end{pmatrix}, \tag{9}$$

where $det\mathbf{R}_4 = det\mathbf{E}\, det\mathbf{F} = (e_0^2 + e_1^2 + e_2^2 + e_3^2)^2(f_0^2 + f_1^2 + f_2^2 + f_3^2)^2 = 1$ holds.

Due to the free choice of sign in Theorem 1, the decomposition into a left unit quaternion $\mathfrak{E}$ and a right unit quaternion $\mathfrak{F}$ yields a double cover of SO(4). Therefore we consider again the homogeneous 8-tuple $(e_0 : \ldots : e_3 : f_0 : \ldots : f_3)$, which can be seen as a point in P^7. Hence there is a bijection between SO(4) and all real points $\mathscr{S}$ of P^7, which are located on the quadric $\Psi \subset P^7$ given by

$$(e_0^2 + e_1^2 + e_2^2 + e_3^2) - (f_0^2 + f_1^2 + f_2^2 + f_3^2) = 0, \tag{10}$$

($\Rightarrow$ the signature of Ψ is $(4_+, 4_-, 0_0)$) sliced along the 3-dimensional space $e_0 = e_1 = e_2 = e_3 = 0$, as the corresponding quaternion $\mathfrak{E}$ cannot be normalized. But this 3-space does not have a real intersection with Ψ and therefore no point of Ψ has to be removed. Note that Eq. (10) expresses the fact that $\mathfrak{F}$ is also normalized if $\mathfrak{E}$ is.

Remark 1 If we identify E^3 with the hyperplane $x_0 = 0$, all points of the 3-dimensional generator-space $f_0 = e_0$, $f_i = -e_i$ for $i = 1, 2, 3$ ($\Leftrightarrow \mathfrak{F} = \widetilde{\mathfrak{E}}$) of Ψ, map the hyperplane $x_0 = 0$ onto itself. Therefore this 3-dimensional generator-space is the well-known Euler-Rodrigues parameter space $(e_0 : \ldots : e_3)$ of SO(3).

The extension of this kinematic mapping of SO(4) with respect to translations of E^4 can be done as follows:

Theorem 2 *The mapping of points $\mathsf{X} \in E^4$ to $\mathsf{X}' \in E^4$ induced by any element of SE(4), can be written as follows (by using ι_4):*

$$\mathfrak{X} \mapsto \mathfrak{X}' \quad with \quad \mathfrak{X}' := \mathfrak{E} \circ \mathfrak{X} \circ \mathfrak{F} - (\mathfrak{T} \circ \mathfrak{E} + \mathfrak{E} \circ \widetilde{\mathfrak{T}}). \tag{11}$$

Moreover the mapping of Eq. (11) is an element of SE(4) for any triple of quaternions $\mathfrak{E}, \mathfrak{F}, \mathfrak{T}$, where $\mathfrak{E}$ and $\mathfrak{F}$ are unit quaternions.

Proof Due to Theorem 1, we only have to show that there is a bijection between the coordinates of the translation vector $\mathbf{s}_4 = (s_0, s_1, s_2, s_3)^T$ and the entries $t_0, \ldots, t_3$ of $\mathfrak{T}$ for a given unit quaternion $\mathfrak{E}$. It can easily be seen that s_1, s_2, s_3 equal the expressions given in Eq. (5) and that $s_0 = -2(e_0t_0 + e_1t_1 + e_2t_2 + e_3t_3)$ holds. Now we solve these four equations for $t_0, \ldots, t_3$ which yields:

$$t_0 = -(e_0s_0 + e_1s_1 + e_2s_2 + e_3s_3)/2, \qquad t_1 = (e_0s_1 - e_1s_0 - e_2s_3 + e_3s_2)/2,$$
$$t_2 = (e_0s_2 + e_1s_3 - e_2s_0 - e_3s_1)/2, \qquad t_3 = (e_0s_3 - e_1s_2 + e_2s_1 - e_3s_0)/2.$$

This already proves the theorem. □

As both triples of quaternions $\pm(\mathfrak{E}, \mathfrak{F}, \mathfrak{T})$, where $\mathfrak{E}$ and $\mathfrak{F}$ are unit quaternions, correspond to the same Euclidean motion of E^4, we consider the homogeneous 12-tuple $(e_0 : \ldots : e_3 : f_0 : \ldots : f_3 : t_0 : \ldots : t_3)$. These 12 homogeneous motion parameters for E^4, which are called the *new parameters* for short, can be interpreted as a point of a projective 11-dimensional space P^{11}. Therefore there is a bijection between SE(4) and all real points $\mathscr{S}$ of P^{11} located on the cylinder Ξ over Ψ, which is also given by Eq. (10) ($\Rightarrow$ the signature of Ξ is $(4_+, 4_-, 4_0)$) and is sliced along the 7-dimensional space $e_0 = e_1 = e_2 = e_3 = 0$, as the corresponding quaternion $\mathfrak{E}$ cannot be normalized. But the real intersection of this 7-space and Ξ equals the 3-dimensional generator-space $e_0 = e_1 = e_2 = e_3 = f_0 = f_1 = f_2 = f_3 = 0$ of Ξ. Therefore only the points of this 3-space have to be removed from Ξ.

Remark 2 If we identify E^3 with the hyperplane $x_0 = 0$, all points of the 7-dimensional generator-space $f_0 = e_0$, $f_i = -e_i$ for $i = 1, 2, 3$ (cf. Remark 1) of Ξ, which additionally fulfill the condition that no translation is done in direction of x_0 ($\Leftrightarrow s_0 = 0$), map the hyperplane $x_0 = 0$ onto itself. As the condition $s_0 = 0$ equals the Study condition, the 7-dimensional generator-space of Ξ is the Study parameter space of SE(3). This shows that the Study parameters and subsequently the Blaschke-Grünwald parameters can be obtained from the *new parameters*.

Finally it should be noted that the exceptional quadric of the *new parameter space* is given by $e_0^2 + e_1^2 + e_2^2 + e_3^2 = 0$ and therefore it is also quasi-elliptic (cf. [11]) like the kinematic image spaces named after Study and Blaschke-Grünwald.

3 The Hypersphere Condition

In the following we study the hypersphere condition Ω_n of E^n for $n = 4$, but we formulate everything in a way that it is also valid for the lower-dimensional counterparts, i.e. the sphere condition for $n = 3$ and the circle condition for $n = 2$.

Based on our results of Sect. 2, the mapping $\mathsf{X} \mapsto \mathsf{X}'$ implied by an element of SE(n) can be written in vector-representation as follows:

$$\begin{pmatrix} x'_{4-n} \\ \ldots \\ x'_3 \end{pmatrix} = \frac{1}{N_n} \left[\mathbf{R}_n \begin{pmatrix} x_{4-n} \\ \ldots \\ x_3 \end{pmatrix} + \mathbf{s}_n \right], \tag{12}$$

with $N_2 = e_0^2 + e_1^2$ and $N_3 = N_4 = e_0^2 + e_1^2 + e_2^2 + e_3^2$, respectively, if we neglect the normalizing condition $N_n = 1$. Note that the factor N_n^{-1}, which corresponds to the division by 1, is inserted in order to homogenize Eq. (12).

Now we can write the constraint Ω_n that the point X is located on a hypersphere of E^n with midpoint M and radius ρ as follows:

$$\Omega_n: \quad (x'_{4-n} - m_{4-n})^2 + \ldots + (x'_3 - m_3)^2 - \rho^2 = 0, \tag{13}$$

where $m_{4-n}, \ldots, m_3$ are the coordinates of M and with $x'_{4-n}, \ldots, x'_3$ of Eq. (12). As $N_n \neq 0$ holds the denominator of Ω_n cannot vanish and so we can focus on the nominator, which is a homogeneous polynomial P_n of degree 4 in the motion parameters.

- $n = 2$: A closer look at P_2 shows that N_2 factors out and we remain with a homogeneous quadratic equation in the Blaschke-Grünwald parameters, which is the so-called circle equation Q_2.
- $n = 3$: Interestingly P_3 does not behave like P_2, but Husty [12] showed that N_3 factors out if we add four times the squared Study condition to P_3. The remaining homogeneous quadratic equation in the Study parameters is the so-called sphere equation Q_3, which is the key for solving the direct kinematics of SG platforms (cf. [12]). Note that according to Sect. 2.2, we can obtain Q_2 from Q_3 by setting $m_1 = x_1 = e_2 = e_3 = t_0 = t_1 = 0$.
- $n = 4$: It is not difficult to see that P_4 factors into N_4 and a homogeneous quadratic equation in the *new parameters*. This is the so-called hypersphere equation Q_4, which is the base of any algebraic kinematical study (e.g. solution of the direct kinematics) of 10-dof $S_4\underline{P}S_4$ parallel manipulators.
  According to Remark 2 we can obtain Q_3 from Q_4 by setting $m_0 = x_0 = 0$, $f_0 = e_0$, $f_i = -e_i$ for $i = 1, 2, 3$. This also sheds light onto Husty's tricky addition, as it corresponds to the summand s_0^2 within the *new parameter* approach.

Based on the hypersphere condition, we prove in the next theorem that singular (infinitesimal movable) poses of 10-dof $S_4\underline{P}S_4$ manipulators have an analogous line-geometric characterization as those of their lower-dimensional counterparts.

Theorem 3 *A 10-dof $S_4\underline{P}S_4$ manipulator is in a singular configuration $\mathscr{C}$ if and only if the carrier lines of the ten $\underline{P}$-joints belong to a linear complex of lines of E^4.*

Proof For the proof we consider an arbitrarily given configuration $\mathscr{C}$ of the 10-dof $S_4\underline{P}S_4$ manipulator. Without loss of generality we can assume that the coordinates of the platform anchor points X and the base anchor points M are given with respect to the same reference frame. Therefore $\mathscr{C}$ is given by the identity map of SE(4), which corresponds to the point $I = (1 : 0 : 0 : 0 : 1 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0)$ on Ξ.

We study an arbitrary motion through $\mathscr{C}$, which depends on the time τ, where $\mathscr{C}$ is passed at $\tau = 0$. Now the partial derivative of the normalizing condition $N_4 = 1$ and the equation of the cylinder Ξ with respect to τ evaluated in I yields $\dot{e}_0 = \dot{f}_0 = 0$, where the superior dot denotes the time derivative. Under consideration of this result the partial derivative of Q_4 with respect to τ simplifies to:

$$\frac{1}{2\rho}\left(\sum_{i=1}^{3} Q_{4,e_i}\dot{e}_i + \sum_{i=1}^{3} Q_{4,f_i}\dot{f}_i + \sum_{i=0}^{3} Q_{4,t_i}\dot{t}_i\right) = \dot{\rho} \quad \text{with} \quad Q_{4,v_i} := \frac{\partial Q_4}{\partial v_i} \tag{14}$$

for $v \in \{e, f, t\}$ and

$$Q_{4,e_1} + Q_{4,f_1} = 4(m_0x_1 - m_1x_0) =: 4g_{01} \quad Q_{4,e_1} - Q_{4,f_1} = 4(m_2x_3 - m_3x_2) =: 4g_{23}$$
$$Q_{4,e_2} + Q_{4,f_2} = 4(m_0x_2 - m_2x_0) =: 4g_{02} \quad Q_{4,e_2} - Q_{4,f_2} = 4(m_3x_1 - m_1x_3) =: 4g_{31}$$
$$Q_{4,e_3} + Q_{4,f_3} = 4(m_0x_3 - m_3x_0) =: 4g_{03} \quad Q_{4,e_3} - Q_{4,f_3} = 4(m_1x_2 - m_2x_1) =: 4g_{12}$$
$$Q_{4,t_0} = 4(m_0 - x_0) =: 4g_{04} \qquad Q_{4,t_j} = 4(x_j - m_j) =: 4g_{4j}$$

for $j = 1, 2, 3$. This is a linear relation between the instantaneous motion of the platform and the velocity $\dot{\rho}$ of the $\underline{P}$-joint. Therefore the coefficient matrix with respect to $\dot{e}_1, \dot{e}_2, \dot{e}_3, \dot{f}_1, \dot{f}_2, \dot{f}_3, \dot{t}_0, \dot{t}_1, \dot{t}_2, \dot{t}_3$ of the system of ten linear equations (14), which are induced by the ten $S_4\underline{P}S_4$-legs, is the 10×10 Jacobian matrix $\mathbf{J}$. As a consequence the given configuration $\mathscr{C}$ is singular for $det\mathbf{J} = 0$.

In the following, we consider the projective point coordinates of X and M, i.e.

$$(x_0^\star : \ldots : x_3^\star : x_4^\star) := (x_0 : \ldots : x_3 : 1), (m_0^\star : \ldots : m_3^\star : m_4^\star) := (m_0 : \ldots : m_3 : 1).$$

With this notation the ten Grassmann coordinates $l_{ij} = -l_{ji}$ of the line $\left[\mathsf{M}, \mathsf{X}\right] \in E^4$, which are the analogue to the six Plücker coordinates of lines in E^3, can be computed as $l_{ij} := m_i^\star x_j^\star - m_j^\star x_i^\star$ for $i \neq j$ and $i, j \in \{0, 1, 2, 3, 4\}$ (cf. [13]). As the g_{ij}'s defined in Eq. (14) equal the l_{ij}'s, the theorem is proven. □

4 Conclusion and Outlook

In this chapter we developed all basics (kinematic mapping, hypersphere condition, Jacobian matrix, line-geometric characterization of singular configurations) for a future, deeper study of hyperplanar 10-dof $S_4\underline{P}S_4$ parallel manipulators of E^4, which aims to improve the geometric understanding of non-planar SG platforms.

Moreover the hypersphere condition, written in the novel 12 homogeneous motion parameters for E^4, yielded a deeper insight into Husty's tricky addition for the generation of the sphere condition in terms of Study parameters (cf. [12]), which is the central equation for (computational) algebraic kinematics of SG manipulators.

Klawitter and Hagemann showed in Sect. 5.1 (resp. 5.2) of [9] (see also Sect. 9 of [14]) that the algebraic structure of the Study parameters (resp. Blaschke-Grünwald parameters) corresponds to the Spin group of the even part of the Clifford Algebra with signature $(3_+, 0_-, 1_0)$ (resp. $(2_+, 0_-, 1_0)$), which is isomorph to the group of unit dual quaternions (cf. Sect. 2.1). The algebraic structure behind the *new parameters* of SE(4) is still unknown and dedicated to future research.

Acknowledgments This research is supported by Grant No. I 408-N13 and Grant No. P 24927-N25 of the Austrian Science Fund FWF. Moreover the author would like to thank the reviewers for their suggestions and comments, which have helped to improve the quality of the chapter.

References

1. Mielczarek, S., Husty, M.L., Hiller, M.: Designing a redundant Stewart-Gough platform with a maximal forward kinematics solution set. In: Proceedings of the International Symposion of Multibody Simulation and Mechatronics (MUSME), Mexico City, Mexico (2002)
2. Borras, J., Thomas, F., Torras, C.: Singularity-invariant leg rearrangements in doubly-planar Stewart-Gough platforms. In: Proceedings of Robotics Science and Systems, Zaragoza, Spain (2010)
3. Nawratil, G.: Review and recent results on Stewart Gough platforms with self-motions. Appl. Mech. Mater. **162**, 151–160 (2012)
4. Nawratil, G.: Correcting Duporcq's theorem. Mech. Mach. Theory **73**, 282–295 (2014)
5. Blaschke, W.: Euklidische Kinematik und nichteuklidische Geometrie. Z. Math. Phys. **60**, 61–91, 203–204 (1911)
6. Grünwald, J.: Ein Abbildungsprinzip, welches die ebene Geometrie und Kinematik mit der räumlichen Geometrie verknüpft. Sitz.-Ber. der math.-nat. Klasse der kaiserlichen Akademie der Wissenschaften Wien **120**, 677–741 (1911)
7. Study, E.: Geometrie der Dynamen. Teubner, Leipzig (1903)
8. Husty, M., Karger, A., Sachs, H., Steinhilper, W.: Kinematik und Robotik. Springer, Berlin (1997)
9. Klawitter, D., Hagemann, M.: Kinematic mappings for Cayley-Klein geometries via Clifford algebras. Beiträge zur Algebra und Geometrie **54**(2), 737–761 (2013)
10. Mebius, J.E.: History of the quaternion representation theorem for four-dimensional rotations. http://www.jemebius.home.xs4all.nl/So4hist.htm
11. Giering, O.: Vorlesungen über höhere Geometrie. Vieweg, Braunschweig (1982)
12. Husty, M.: An algorithm for solving the direct kinematics of general Stewart-Gough platforms. Mech. Mach. Theory **31**(4), 365–380 (1996)
13. Pottmann, H, Wallner, J.: Computational Line Geometry. Springer, Berlin (2001)
14. Selig, J.M.: Geometric Fundamentals of Robotics. Springer, New York (2005)

Direct Kinematics of an Orthogonal 6PRRS Parallel Manipulator

Paul Zsombor-Murray

Abstract Direct kinematic (DK) analysis of a fully parallel—i.e., where each identical 6dof serial leg contains a single basal, actuated joint—highly symmetric six-PRRS legged robot is presented. Though not pursued herein, it is easy to see that the inverse (IK) problem is linear. What is remarkable is that this device may be thought of as a parallel architecture equivalent to the classical 6R wrist partitioned serial manipulator because the linear point positioning aspect of the DK problem is distinctly separated from its rotational aspect which is only slightly more complicated. This is accomplished by rotational transformation (rotational kinematic mapping) that places 6 points, the vertices of a regular octahedron originally in an ideally convenient end effector (EE) frame, onto 6 planes explicitly defined by the 6 actuated prismatic P-joints in the fixed frame FF. A numerical example is formulated, solved and results are illustrated in principal view pairs. Some plans for future, more thorough, investigation are laid out, workspace and singularity and a different end effector architecture are discussed briefly.

Keywords Direct kinematics · Parallel manipulator · Kinematic mapping · Six degrees of freedom · Rotationally partitioned

1 Introduction

Recently, the novel six-legged robot shown in Fig. 1 was discovered as a website posting [1]. Three pairs of parallel actuated prismatic or P-joints, in the fixed frame FF, move in three mutually orthogonal directions. Attached to each of the six actuated links is a serial RRS subchain where R and S respectively indicate a free revolute joint and a distal spherical joint attached to the end effector. Going by what can be

P. Zsombor-Murray (✉)
Centre for Intelligent Machines, McGill University, Montreal, Canada
e-mail: paul@cim.mcgill.ca

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_3,

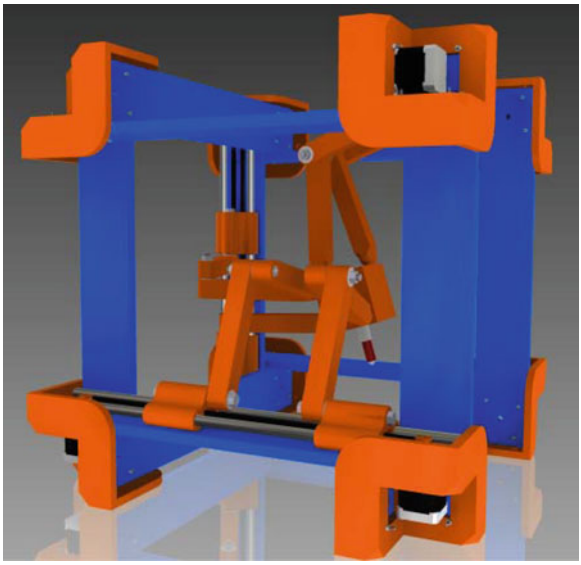


Fig. 1 Orthogonal 6PRRS parallel robot

gleaned from Fig. 1 it seems that the six S-joint centres, attached to the end effector, are distributed on the vertices of a regular octahedron. For the purpose of direct kinematic analysis (DK), that follows, these vertices are assigned homogeneous coordinates

$$A\{1:1:0:0\},\ B\{1:-1:0:0\},\ C\{1:0:1:0\}$$

$$D\{1:0:-1:0\},\ E\{1:0:0:1\},\ F\{1:0:0:-1\},$$

expressed in the moving or end effector frame EE shown on the right in Fig. 2.

Because the axes of the unactuated parallel RR pair in each leg are parallel to the direction of motion caused by that pair's P-joints the S-joint centre, if that joint is detached from the end effector, sweeps a plane normal to the P-joint imposed motion. These planes are labeled on FF, in Fig. 2, in lower case with the same letter in upper case assigned to the S-joint centre that is confined to that corresponding plane. Figure 3 shows the pair of PRRS legs whose S-joint centres A, B, on planes a, b, translate in the planes' normal direction parallel to the x-axis in FF. The three thickest, mutually orthogonal lines in Fig. 3 represent the rails on which the three pairs of actuated P-joints travel. The plane equations and their coefficients are

$$a: x = p \equiv \{-p:1:0:0\},\ b: x = q \equiv \{-q:1:0:0\},\ c: y = r \equiv \{-r:0:1:0\}$$

$$e: y = s \equiv \{-s:0:1:0\},\ e: z = t \equiv \{-t:0:0:1\},\ f: z = u \equiv \{-u:0:0:1\}.$$

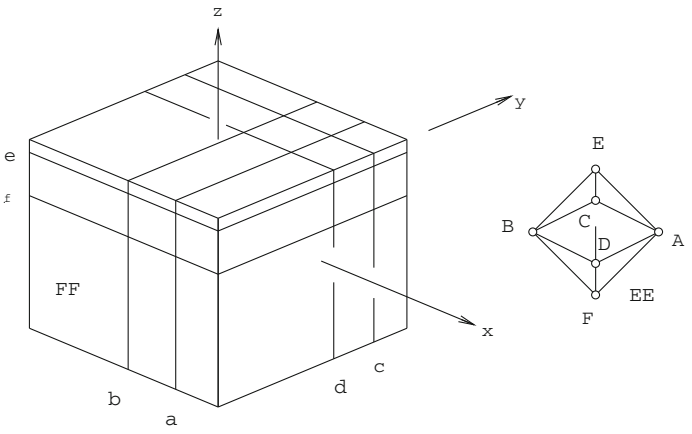


Fig. 2 Three orthogonal pairs of parallel planes and six S-joint centres in regular octahedral array

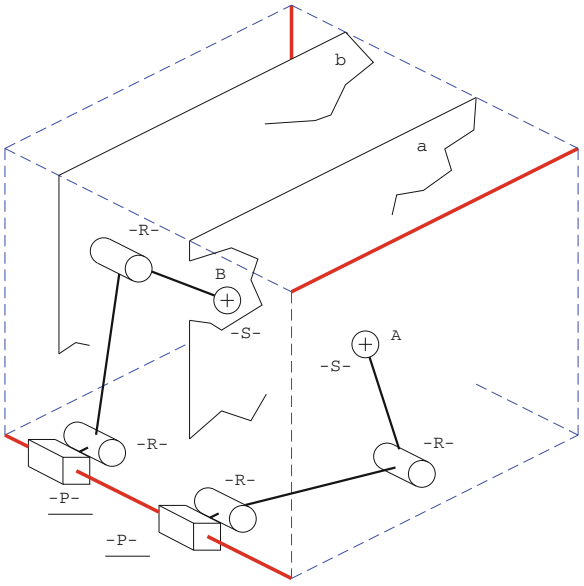


Fig. 3 One of the three PRRS leg pairs

Note how

$$A \in a, B \in b, C \in c, D \in d, E \in e, F \in f$$

have been assigned. Whether or not this corresponds to the layout shown in Fig. 1 the DK approach detailed below is based on the architecture implicit in Figs. 2 and 3. Due to the symmetry of the manipulator and choice of frame EE the solution is simple. Pure translation of the end effector is produced by moving actuated parallel P-joints

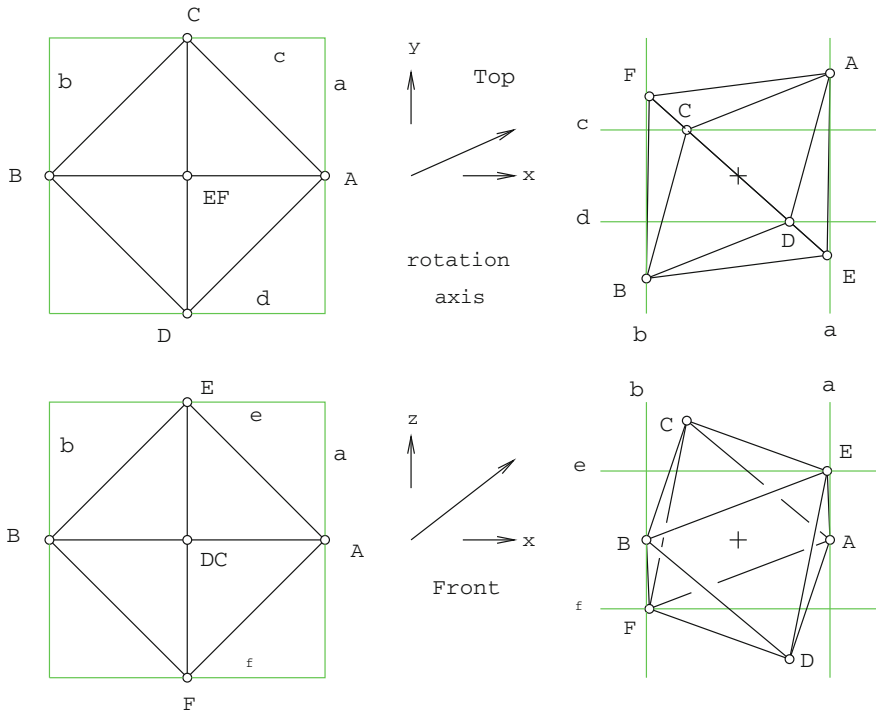


Fig. 4 Home and rotated orientations of end effector

pairs in the same direction at the same rate so as to maintain constant the parallel plane separation distances $p-q, r-s, t-u$. Thus the origin M in EE assumes the midpoint of the rectangular parallelepiped, as shown in Fig. 4, embraced by planes a, b, c, d, e, f; *i.e.*, the point $M\{2 : p+q : r+s : t+u\}$ in FF.

2 Rotational DK Analysis

Analysis is confined to rotation because this manipulator architecture neatly separates it from translation. Pure rotation is effected by moving parallel plane pairs together or apart at the same rate, i.e., in opposite directions so as to maintain their mid-planes on a fixed point M. Operating on the points A, B, C, D, E, F in EE with the rotation matrix populated with Euler parameters (also known as quaternion elements) and equating the appropriate coordinate differences to the three separation distances provides the necessary three–four, if one includes the quaternion norming condition-equations to reveal the rotation, its axis expressed by the unit vector $[\cos\alpha \ \cos\beta \ \cos\gamma]^\top$ and its angle by ϕ taken in a right handed sense with respect to sense given by the direction cosine unit vector, necessary to separate the octahedron

opposite vertex pairs according to the distance between parallel plane pairs in the Cartesian directions peculiar to FF. For example the first operation $[\mathbf{R}]\mathbf{a}_{\mathrm{EE}} = \mathbf{a}_{\mathrm{FF}}$ is written below.

$$\begin{bmatrix} c_0^2+c_1^2+c_2^2+c_3^2 & 0 & 0 & 0 \\ 0 & c_0^2+c_1^2-c_2^2-c_3^2 & 2(c_1c_2-c_0c_3) & 2(c_1c_3+c_0c_2) \\ 0 & 2(c_2c_1+c_0c_3) & c_0^2-c_1^2+c_2^2-c_3^2 & 2(c_2c_3-c_0c_1) \\ 0 & 2(c_3c_1-c_0c_2) & 2(c_3c_2+c_0c_1) & c_0^2-c_1^2-c_2^2+c_3^2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} c_0^2+c_1^2+c_2^2+c_3^2 \\ c_0^2+c_1^2-c_2^2-c_3^2 \\ 2(c_1c_2+c_0c_3) \\ 2(c_1c_3-c_0c_2) \end{bmatrix}$$

Due to the norming condition $c_0^2 + c_1^2 + c_2^2 + c_3^2 = 1$ the second row $c_0^2 + c_1^2 - c_2^2 - c_3^2 = p - q$ defines the x-coordinate distance between points A, B in FF.

3 Constraint Equations and Solution

The four constraint equations are

$$2(c_0^2 + c_1^2 - c_2^2 - c_3^2) - p + q = 0, \quad 2(c_0^2 - c_1^2 + c_2^2 - c_3^2) - r + s = 0,$$

$$2(c_0^2 - c_1^2 - c_2^2 + c_3^2) - p + q = 0, \quad c_0^2 + c_1^2 + c_2^2 + c_3^2 - 1 = 0.$$

These yield the following solutions.

$$c_0^2 = (p - q + r - s + t - u + 2)/8, \quad c_1^2 = (p - q - r + s - t + u + 2)/8,$$
$$c_2^2 = (-p + q + r - s - t + u + 2)/8, \quad c_3^2 = (-p + q - r + s + t - u + 2)/8$$

Note that since the solutions are in terms of the square of Euler parameters all square root values can be positive or negative so as to produce up to eight end effector poses. Why not 16, i.e., $2^4 = 16$? Since the four quaternion elements are homogeneous coordinates the sign of one of them can be chosen.

4 Example

Positioning properties of this manipulator, as regards both DK and IK, are adequately covered above. Therefore this example will be confined to dealing with end effector orientation pertaining to DK. Assume joint coordinates are imposed to produce the following plane parameters.

$$p - q = 4/3, \quad r - s = 2/3, \quad t - u = 1$$

These yield

$$c_0^2 = \frac{4/3 + 2/3 + 3/3 + 6/3}{8} = 5/8, \quad c_1^2 = \frac{4/3 - 2/3 - 3/3 + 6/3}{8} = 5/24,$$
$$c_2^2 = \frac{-4/3 + 2/3 - 3/3 + 6/3}{8} = 1/24, \quad c_3^2 = \frac{-4/3 - 2/3 + 3/3 + 6/3}{8} = 1/8.$$

This produces the following quaternion elements (or Euler parameters), rotation axis direction cosines and rotation half-angle cosines and sines.

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \equiv \begin{bmatrix} \cos(\phi/2) \\ \cos\alpha \sin(\phi/2) \\ \cos\beta \sin(\phi/2) \\ \cos\gamma \sin(\phi/2) \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\sqrt{\frac{5}{2}} \\ \pm\frac{1}{2}\sqrt{\frac{5}{6}} \\ \pm\frac{1}{2}\sqrt{\frac{1}{6}} \\ \pm\frac{1}{2}\sqrt{\frac{1}{2}} \end{bmatrix} \equiv \begin{bmatrix} \sqrt{\frac{5}{8}} \\ \pm\sqrt{\frac{5}{9}}\sqrt{\frac{3}{8}} \\ \pm\sqrt{\frac{1}{9}}\sqrt{\frac{3}{8}} \\ \pm\sqrt{\frac{1}{3}}\sqrt{\frac{3}{8}} \end{bmatrix}$$

It is noted with satisfaction that the values above confirm that

$$c_0^2 + c_1^2 + c_2^2 + c_3^2 = \cos^2\alpha + \cos^2\beta + \cos^2\gamma = 1.$$

Choosing the solution wherein all signs are positive and putting the c_i into the rotation matrix reveals the following displacement operator

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2/3 & -\sqrt{5}/6 & \sqrt{15}/6 \\ 0 & -\sqrt{5}/3 & 1/3 & -\sqrt{3}/3 \\ 0 & 0 & \sqrt{3}/2 & 1/2 \end{bmatrix}$$

whose determinant is 1. Premultiplying the point vectors of A,B,C,D,E,F, given above in EE, by this matrix gives the displaced position vectors.

$$\mathbf{a} = \begin{bmatrix} 2/3 \\ \sqrt{5}/3 \\ 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} -2/3 \\ -\sqrt{5}/3 \\ 0 \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} -\sqrt{5}/6 \\ 1/3 \\ \sqrt{3}/2 \end{bmatrix}$$
$$\mathbf{d} = \begin{bmatrix} \sqrt{5}/6 \\ -1/3 \\ -\sqrt{3}/2 \end{bmatrix}, \quad \mathbf{e} = \begin{bmatrix} \sqrt{15}/6 \\ -\sqrt{3}/3 \\ 1/2 \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} -\sqrt{15}/6 \\ \sqrt{3}/3 \\ -1/2 \end{bmatrix}$$

Referring to Fig. 4 one sees, all plotted to scale, on the left the six S-joint centres A, B, C, D, E, F and the corresponsing planes a, b, c, d, e, f in edge or line view on these points in plan (top) and elevation (front) projections in the so-called "home"

orientation. In the middle appears the same two views of the rotation axis implied by the quaternion. The angle of rotation, in a right hand sense, is about $\phi = 75.52°$. On the right there appear the two views of the six S-joint centres and six constraining planes after the quaternion rotation has been imposed on the home attitude.

5 Conclusion

Extensions to this analysis would logically include

- Examination of workspace and singularity,
- Generalizing by treating DK analysis of various S-joint distributions on the vertices of a scalene octahedron and/or,
- Similarly treating an architecture with the six constraining planes embracing a general convex hexahedral volume.

It should be noted here that the simplicity of the analysis presented comes about by applying the rotational kinematic mapping method, introduced by Husty [2], to mapping six points in EE to six planes in FF, called "Point-on-Plane" constraint equations (PoP) by Zsombor-Murray and Gfrerrer [3]. Upon undertaking analysis of the most general case, of arbitrary octahedral vertices and convex hexahedron, six general PoP constraints will arise. The rotation matrix, shown previously in Sect. 2, Rotational DK Analysis, will, in its first column, acquire dual quaternion elements thus.

$$\begin{bmatrix} c_0^2 + c_1^2 + c_2^2 + c_3^2 & \dots \\ 2(c_0 d_1 - c_1 d_0 + c_2 d_3 - c_3 d_2) & \dots \\ 2(c_0 d_2 - c_2 d_0 + c_3 d_1 - c_1 d_3) & \dots \\ 2(c_0 d_3 - c_3 d_0 + c_1 d_2 - c_2 d_1) & \dots \end{bmatrix}$$

In these six quadrics all d_i appear only in linear form. Eliminating these one is left with the quaternion elements c_i in both linear and quadratic form. Therefore the most general problem will admit eight solutions too. These however must be computed individually and not by simply permuting three $\pm$ signs.

6 Appendix: A Reprise

Here some concerns expressed by a reviewer and DK with a somewhat different layout of the 6 EE points $A', \dots, F'$ are briefly addressed.

Re-examine Figs. 2 and 3 then consider Fig. 5 which shows

- On the left, S-joint points $A, \dots, F$ on regular octahedron vertices and the 6 points $A', \dots, F'$, on midpoints of 6 of the 12 cube edges that circumscribe the octahedron. The latter was supplied by Professor Bonev [4] as the "correct ones" used by the designer [1].

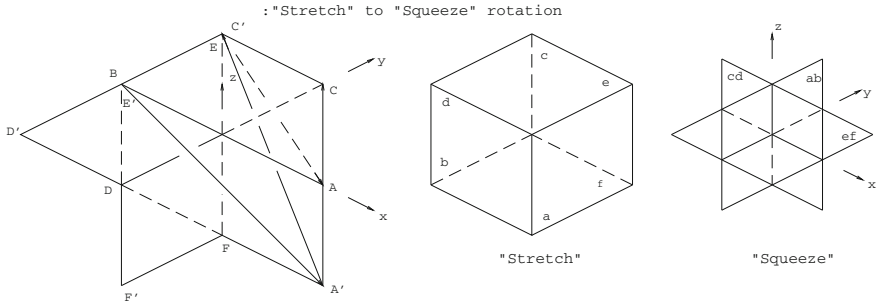


Fig. 5 Two S-joint distributions on end effector

- In the middle at $\frac{1}{2}$-scale, the "stretched" configuration of planes $a, \ldots, f$ which encloses a 2×2 unit cube, the faces of which cannot be further separated.
- On the right at $\frac{1}{2}$-scale, the "squeezed" configuration such that the cube collapses to a point and (a, b), (c, d), (e, f) describe three mutually orthogonal coincident plane pairs.

The S-joint point sets $A, \ldots, F$ and $A', \ldots, F'$ have been compatibly distributed in both stretch and squeeze configuration to demonstrate that the rotational workspace limits are identical with both EE designs. This can be shown by noting the chords of the arcs described by migration of points under a single (quaternion) rotation about a fixed axis during the transition is in the same direction, to wit:

$$\begin{array}{cc} (A \to C) \times (C \to E) & (A' \to C') \times (C' \to E') \\ [1,0,0) \to (0,1,0) \times (0,1,0) \to (0,0,1)] & [(1,0,-1) \to (-1,1,0) \times (-1,1,0) \to (0,-1,1)] \\ \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix} \times \begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} & \begin{bmatrix} -2 \\ 1 \\ 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix} \end{array}$$

But a given intermediate rotation between the two extremes described above is *not* produced by the same placement of planes $a, \ldots, f$. This is evident from the set of four constraint equations obtained for the second EE configuration.

$$\begin{aligned} c_0c_2 + c_1c_3 - d_{ab}/4 = 0, \quad c_0c_3 + c_1c_2 - d_{cd}/4 = 0, \\ c_0c_1 + c_2c_3 - d_{ef}/4 = 0, \quad c_0^2 + c_1^2 + c_2^2 + c_3^2 - 1 = 0 \end{aligned}$$

These produce a monovatiate of degree 6 in squares. It is hoped that this reprise helps to explain some issues raised by **Conclusion** above.

References

1. http://forums.reprap.org/read.php?178,229935,229935#msg-229935 (2013)
2. Husty, M.L.: Kinematic mapping of spherical three-legged platforms. In: Proceedings of 15th Canadian Conference of Applied Mechanics (CANCAM 95), vol. 2, pp. 874–875. ISBN 0920049-06, Victoria, 95-05, -06 (1995)
3. Zsombor-Murray, P.J., Gfrerrer, A.: A unified approach to direct kinematics of some reduced motion parallel manipulators. ASME J. Mech. Robot. **2**(2), 021006 (10 p) (2010). doi:10.1115/1.4001095
4. Bonev, I.: Private communication. 13–08-20 (2013)

The Hidden Robot Concept: A Tool for Control Analysis and Robot Control-Based Design

Sébastien Briot, Victor Rosenzveig and Philippe Martinet

Abstract Exteroceptive sensors can be used to estimate the robot pose in order to suppress inaccuracies coming from the accumulation of modelling errors when using the classical control approach. In some cases, it is impossible to directly observe the end-effector. Thus we can replace it efficiently by the observation of the legs directions. However, with such an approach, unusual results were recorded, namely: (i) the possibility of controlling the robot by observing a limited number of legs, and (ii) in some cases, the robot does not converge to the desired end-effector pose, even if the observed leg did. These results can be explained through the use of the hidden robot concept, which is a tangible visualisation of the mapping between the observed leg direction space and Cartesian space. In the present chapter, it is explained (1) why the tools used in mechanical design can be efficently applied in control analysis via the use of the hidden robot concept which is, in our opinion, a way to unify the analysis of the mechanical and control performances and (2) why we believe that the hidden robot concept must be used to modify the robot design methodologies in order to include control-based performance indices.

Keywords Parallel robots · Control analysis · Robot design

S. Briot (✉) · V. Rosenzveig · P. Martinet
IRCCyN, UMR CNRS 6597, Nantes, France
e-mail: Sebastien.Briot@irccyn.ec-nantes.fr

V. Rosenzveig
e-mail: Victor.Rosenzveig@irccyn.ec-nantes.fr

P. Martinet
e-mail: Philippe.Martinet@irccyn.ec-nantes.fr

S. Briot · V. Rosenzveig · P. Martinet
École Centrale de Nantes, Nantes, France

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_4,

1 Introduction

Sensor-based control approaches have proven to be more efficient than model-based control approaches when accuracy is required in robotized industrial applications [9]. The most usual approach is to observe the robot end-effector pose through the use of a camera. However, for some operations, such as the milling of materials, it is not possible to observe the end-effector. An innovative vision-based control approach that can be used on parallel robots has been proposed in [2]. It is based on the fact that parallel robot links are usually made of rectilinar cylindrical rods that can be easily detected in the camera space. The information being acquired through an external sensor, this technique allows to estimate indirectly the pose of the end-effector from it.

Servoing through the robot leg observation was sucessfully applied to several types of robots, such as the Gough-Stewart (GS) platform, the Adept Quattro and other robots of the same family [2, 3].

However, two unexpected results arose from the use of this technique: (1) it was possible to control the robot by observing a number of legs fewer than the total number of legs; this is surprising because in actuator-based control schemes, each actuated leg has to be controlled to fully servo the robot, and (2) in some cases, the robot did not converge to the desired pose, even if all observed leg directions did.

Not only were these two points inexplicable, but other questions arose too: (3) are we sure there are no local minima (for which the error in the observation space is non zero while the robot platform cannot move [8]) and (4) are we sure that there is no singularity in the mapping between the leg direction and the Cartesian spaces?

Due to the unusual nature of this visual servoing technique, all these points were left unanswered. Indeed, the nature of the mapping was not clearly understood. The answer came only recently, when two of the authors of the present chapter proposed the existence of a virtual robot model "hidden" within the controller. This robot presents singular configurations and assembly modes different from the controlled robot, and it is this hidden robot whose properties are being used through the observation of the real robot's leg directions. This proposition was fully demonstrated and validated through experiments in [13].

The aim of the present chapter is to introduce the generalized concept of hidden robot model and to show why the tools used in mechanical design can be efficently applied in control analysis and how the hidden robot concept must modify the robot design methodologies in order to include control-based performance indices.

2 Leg Observation Based Control

2.1 Cylindrical Leg Observation and Interaction Matrix

The controller proposed in [2] is based on the fact that the robot leg directions ${}^{c}\underline{\mathbf{u}}_i$ (extracted from the projection of the cylindrical link in the image space—Fig. 1) are

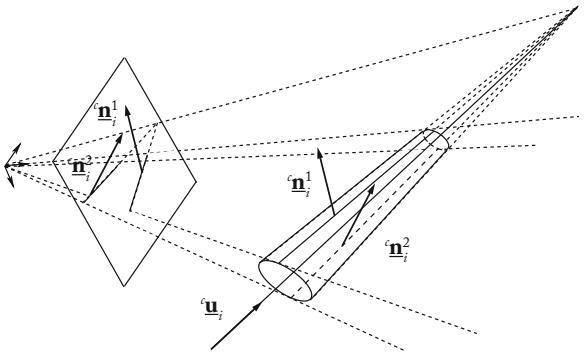


Fig. 1 Projection of a cylinder in the image

controlled and that the leg direction velocities can be linked to the platform twist ${}^c\tau_c$ through the expression (the superscript "c" denotes the camera frame):

$$ {}^c\dot{\underline{\mathbf{u}}}_i = \mathbf{M}_i^{T\,c}\tau_c \tag{1} $$

where $\mathbf{M}_i^T$ is the (3×6) interaction matrix for the leg i.

For spatial parallel robots, matrices $\mathbf{M}_i$ are in general of rank 2 [2]. As a result, for spatial robots with more than 2 *dof*, the observation of several independent legs is necessary to control the end-effector pose. An interaction matrix $\mathbf{M}^T$ can be obtained by stacking k matrices $\mathbf{M}_i^T$ of k legs, which leads to:

$$ {}^c\dot{\underline{\mathbf{u}}} = \left[{}^c\dot{\underline{\mathbf{u}}}_i, {}^c\dot{\underline{\mathbf{u}}}_j, \ldots, {}^c\dot{\underline{\mathbf{u}}}_m\right]^T = \left[\mathbf{M}_i, \mathbf{M}_j, \ldots, \mathbf{M}_m\right]^{T\,c}\tau_c = \mathbf{M}^{T\,c}\tau_c \tag{2} $$

2.2 Control

For the visual servoing of a robot, one achieves exponential decay of an error $e(s, s_d)$ between the current primitive vector s and the desired one s_d using a proportional linearizing and decoupling control scheme. The visual primitives being unit vectors, it is more elegant to use the geodesic error rather than the standard vector difference, i.e. the error grounding the proposed control law will be $\mathbf{e}_i = {}^c\underline{\mathbf{u}}_i \times {}^c\underline{\mathbf{u}}_{di}$, where ${}^c\underline{\mathbf{u}}_{di}$ is the desired value of ${}^c\underline{\mathbf{u}}_i$. A control is then chosen such that $\mathbf{e}$, the vector stacking the errors $\mathbf{e}_i$ associated to of k legs, decreases exponentially (i.e. $\dot{\mathbf{e}} = -\lambda\mathbf{e}$).

Then, introducing $\mathbf{L}_i^T = -\left[{}^c\underline{\mathbf{u}}_{di}\right]_\times \mathbf{M}_i^T$, where $\left[{}^c\underline{\mathbf{u}}_{di}\right]_\times$ is the cross product matrix associated with the vector ${}^c\underline{\mathbf{u}}_{di}$, the control law can be derived:

$$ {}^c\tau_c = -\lambda\mathbf{L}^{T+}\mathbf{e} \tag{3} $$

where $\mathbf{L}^T$ can be obtained by stacking the matrices $\mathbf{L}_i^T$ of k legs and the upperscript "+" corresponds to the matrix pseudo-inverse.

This expression can be transformed into the control joint velocities:

$$\dot{\mathbf{q}} = -\lambda {}^c\mathbf{J}^{inv} \mathbf{L}^{T+} \mathbf{e} \tag{4}$$

where ${}^c\mathbf{J}^{inv}$ is the inverse kinematic matrix of the robot relating the end-effector twist to the actuator velocities, i.e. ${}^c\mathbf{J}^{inv\,c}\tau_c = \dot{\mathbf{q}}$.

It can be proven that the controller can meet numerical issues if:

- the matrix $\mathbf{L}^T$ is rank deficient: in that case, a null error vector $\mathbf{e}$ can lead to a non null platform twist ${}^c\tau_c$. This appears if and only if $\mathbf{M}^T$ is rank deficient.
- the pseudo-inverse $\mathbf{L}^{T+}$ is rank deficient: in that case, we can meet a local minimum of the controller [8] for which the value of the platform twist ${}^c\tau_c$ is zero while the error $\mathbf{e}$ is not. This appears if and only if $\mathbf{M}^{T+}$ is rank deficient.

Clearly, only stacking leg interaction matrices (which is usual in the visual servoing community) is not enough and the singularities of the controller must be studied. The rank-deficiency conditions are difficult to analyze, and indeed, the complexity for the formerly studied robots was so high that they were never obtained. It is shown in the next part that they can be analyzed by taking into account that the matrix $\mathbf{M}^T$ is indeed the inverse kinematic matrix of a virtual robot hidden into the controller.

3 The Concept of Hidden Robot Model

3.1 How to Define the Legs of the Hidden Robots

The concept of hidden robot model comes from the following observation: in the classical control approach, the encoders measure the motion of the actuators and this measure is linked to the forward kinematic problem (*fkp*) $\mathbf{x} = \mathscr{H}(\mathbf{q})$, where $\mathbf{x}$ represents the platform pose and $\mathbf{q}$ the encoder measures; in the previously described control approach, the leg directions are observed. So, in a reciprocal manner, one could wonder to what kind of virtual actuators such observations correspond, i.e. what is the virtual robot hidden below the new *fkp* $\mathbf{x} = \mathscr{G}(\underline{\mathbf{u}})$.

The virtual robot hidden into the controller corresponding to $\mathbf{x} = \mathscr{G}(\underline{\mathbf{u}})$ can be described as follows [13]. First, the observed links mounted to the platform must be disassembled from the rest of the robot and mounted onto active $\underline{U}$ joints (Fig. 2). The $\underline{U}$ joint must be linked to a passive kinematic chain composed of at most 3 orthogonal passive P joints that ensures that the link on which is it attached performs a translation w.r.t. the base frame. This passive chain is also linked to the segments before the observed links so that they do not change their motion. Note that:

- it is necessary to fix the PPP chain on the preceeding leg links because the information given by the vectors $\underline{\mathbf{u}}_i$ is not enough for rebuilding the full platform position

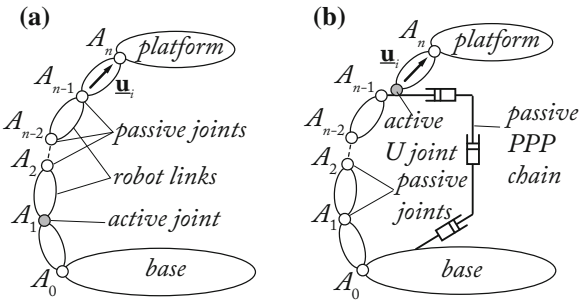


Fig. 2 **a** A general robot leg and **b** its corresponding hidden robot leg when the vector $\underline{\mathbf{u}}_i$ is observed

and orientation: it is also necessary to get information on the location of the anchor point A_{n-1} of the observed segment [1];
- 3 *P* joints are only necessary if and only if the point A_{n-1} describes a motion in the 3D space; if not, the number of *P* joints can be decreased such as in [6];
- when the vector $\underline{\mathbf{u}}_i$ is constrained to move in a plane (e.g. for planar legs), the virtual actuator becomes an $\underline{R}$ joint which must be mounted on the *PPP* chain.

It should be noticed that, in several cases for robots with a lower mobility, the last joint should be changed so that, if the number of observed legs is inferior to the number of real legs, the hidden robot keeps the same mobility (see [12]). Moreover, we have presented above the most general methodology, but not the most elegant. In many cases, a hidden robot leg architecture can be obtained such that less modifications w.r.t the real leg are achieved. For example, for the Quattro [12] made of $\underline{R}$–{2–*UU*} legs for which the parallelogram links are observed ({2–*UU*} subchain links), the *R*–*PPP* chain of the hidden robot leg (which is indeed a{*R*–*PPP*}–{2–$\underline{U}$*U*} leg) could be fully-equivalently replaced by a planar parallelogram (Π joint) [12].

3.2 How Analyzing the Controllability of the Servoed Robots

The aim of this section is to show how to use the hidden robots for answering points 1–4 enumerated in the introduction of the chapter.

Point 1: Let us consider a general parallel robot composed of 6 legs (one actuator per leg) and having six *dof*. *Using the approach proposed in Sect.* 3.1, *each observed leg will lead to a modified hidden robot virtual leg with at least one actuated* $\underline{U}$ *joint that has two degrees of actuation. For controlling 6 dof, only* 6° *of actuations are necessary, i.e. three actuated* $\underline{U}$ *are enough.* Thus, in a general case, only three legs have to be observed to fully control the platform *dof* (e.g. see [6]).

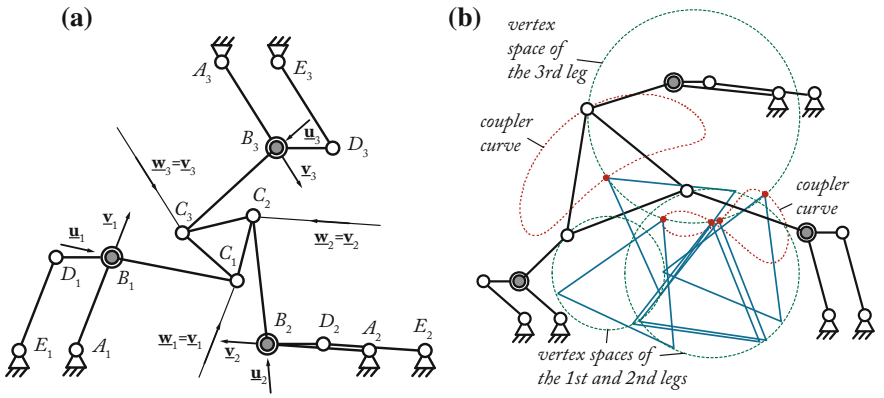


Fig. 3 The hidden robot model of the 3–$\underline{R}RR$ robot and its hidden robot model. **a** 3–$\Pi\underline{R}R$ robot kinematics, **b** the six possible assembly modes

Point 2: In many cases, the hidden robot may have assembly modes and singular configurations different from those of the real robot. If the initial and final robot configurations are not included in the same aspect [11], *the robot will not be able to converge to the desired pose, but to a pose that corresponds to another assembly mode that has the same leg directions as the desired final pose* (e.g. see [6, 12]).

Point 3: The interaction matrix $\mathbf{M}^T$ involved in the controller is the inverse kinematic matrix of the hidden robot (and, consequently, $\mathbf{M}^{T+}$ is the hidden robot kinematic matrix) which, is most of cases, is not free of singularities. Thus, *finding the condition for the rank-deficiency of* $\mathbf{M}^T$ *and* $\mathbf{M}^{T+}$ *is equivalent to find the Type 1 and 2 (also called serial and parallel) singularities of the hidden robot* [10].

Point 4: The robot could converge to *local minima if the matrix* $\mathbf{M}^{T+}$ *is rank deficient, i.e. the hidden robot model encounters a Type 1 singularity.*

Case Study: In the previous chapters, only spatial mechanisms have been studied. For illustrating that section, we have decided to consider the case of the well-known planar 3–$\underline{R}RR$ robot controlled via the observation of the *distal* links (links between the passive *R* joints on each leg). Using the results of previous section, it can be found that its equivalent hidden robot model is a 3–$\Pi\underline{R}R$ robot (Fig. 3—grey joints denote actuated joints). Each of its legs is composed of a passive planar parallelogram (Π joint) which is able to maintain constant the orientation of the links $B_i D_i$ w.r.t. the base and of an $\underline{R}R$ chain which is mounted on the link $B_i D_i$.

Forward kinematics and assembly modes. All the solutions to the *fkp* are at the intersections of the coupler curve (which represents the displacement loci of one platform extremity when one of the leg is disassembled, the actuators of the two other being fixed (see Fig. 3b)—for the present robot, the coupler curve is a sextic curve) with

the vertex space of the disassembled leg (here, a circle) [11]. Thus, the solutions of the *fkp* are at the intersection points (which are at most 6 [11]) between the aforementioned circle and sextic curve (Fig. 3b).

Singular configurations. The Type 1 singularities of 3–$\Pi\underline{R}RR$ robot appear when one leg is fully streched or folded. Its Type 2 singularities appear when the lines of direction $\underline{\mathbf{w}}_i$ ($i = 1, 2, 3$—Fig. 3a) passing through points C_i intersects in the same point (that can be at infinity). Those conditions are different from the Type 2 singular configurations of the 3–$\Pi\underline{R}RR$ robot, for which they appear when all lines passing through C_i of direction $\underline{\mathbf{u}}_i$ intersects in one point [5].

4 Rethinking the Robot Design by Using the Hidden Robot Concept

Through the concept of hidden robot, the tools and methodology used in mechanical design—for solving the *fkp* (e.g. [7, 11], etc.), for analyzing the singularities (e.g. [4, 11], etc.)—can be efficiently used in control design and analysis. However, this concept can also be used to rethink the design process of robots controlled via the use of exteroceptive sensors: *the concept of hidden robot can lead to the definition of control-based performance indices*. We want to mention that, even if we have deliberately limit the approach to the leg-based visual servoing of parallel robots, the hidden robot concept can be extended to any types of robots controlled via the use of exteroceptive sensors (e.g. cameras, lasers, etc.) which are observing robot internal motions in order to estimate the robot external properties (end-effector pose).

Figure 4 illustrates that point. The design process is typically separated into four main phases: (1) the specification of the product requirements coming from the need definition, (2) the phase of conceptual design during which concepts are proposed and evaluated, (3) the embodiement of schemes during which the concepts are developed and analyzed, and (4) the detailed design that leads to the CAD drawing and the manufacturing of prototypes. Phases (1–3) are obviously linked in order to modify/improve the design solutions in case of feasability issues.

If, for a given application, it is specified that a robot must be controlled via the use of exteroceptive sensors observing its legs, the designer has to know that such a requirement leads to the definition, for any kind of robot architecture, to a generic concept of hidden robot (such as the one in Fig. 4 for leg-based visual servoing). Then, during the conceptual design phase, for each proposed robot architecture, the corresponding hidden robots can be defined and their singularities can be analyzed. A feedback can be given to the designer so that he can choose, for example, the robot architectures that have the hidden robot models with the minimal number of singular configurations. If, for any reason, all hidden robots are architecturally singular, the design specifications concerning sensor types and observed elements must be modified (by proposing other sensors, or other observed elements). Then, once the potential architectures are selected, they must be optimized in phase 3. More

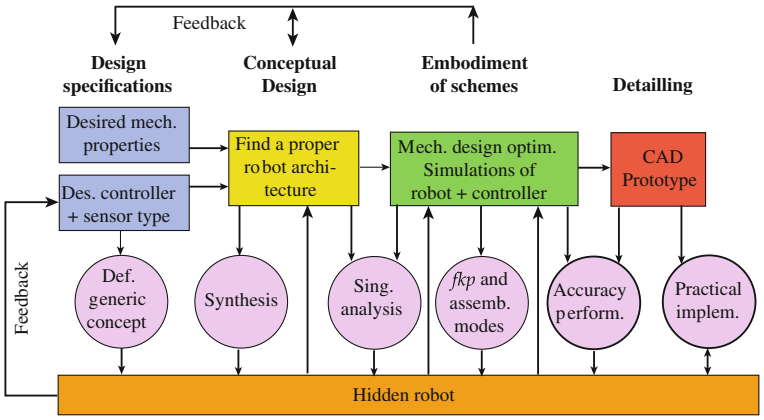


Fig. 4 How the hidden robot concept should be taken into account into the robot design process

detailed kinematic models of the robots and of their corresponding hidden robots are defined and used for extracting performances indices (e.g. w.r.t. accuracy), for defining a controller, and simulating the robot behavior. If the results do not meet the design requirements, previous design steps must be conducted again. Finally, during the implementation of the controller in the manufactured prototype, the hidden robot model equations must be implemented. Our future works will focus on the practical implementation of such a design methodology.

5 Conclusion

In that chapter, it has been shown that some sensor-based control approaches well adapted for servoing parallel robots involve the presence of virtual robots hidden into the controller, which have assembly modes and singularities different from the real robots. The analysis of the robot models hidden into the controllers developped by the sensor-based control community required the use of mathematical tools developped by the mechanical design community. This analysis is crucial to avoid control issues that can arise if interaction matrices are stacked without deeper analysis. It has also been shown that the tools used in mechanical design can be efficently used in control analysis and that the hidden robot concept can be smartly used in the design process of robots in order to include control-based performance indices.

Acknowledgments This work was supported by the French ANR project ARROW (ANR 2011BS3 006 01).

References

1. Andreff, N., Dallej, T., Martinet, P.: Image-based visual servoing of gough-stewart parallel manipulators using legs observation. Int. J. Robot. Res. **26**(7), 677–687 (2007)
2. Andreff, N., Marchadier, A., Martinet, P.: Vision-based control of a Gough-Stewart parallel mechanism using legs observation. In: Proceedings of the IEEE International Conference on Robotics and Automation, ICRA'05, pp. 2546–2551. Barcelona, Spain (2005)
3. Andreff, N., Martinet, P.: Vision-based kinematic modelling of some parallel manipulators for control purposes. In: Proceedings of EuCoMeS. Obergurgl, Austria (2006)
4. Ben-Horin, P., Shoham, M.: Singularity analysis of a class of parallel robots based on grassmanncayley algebra. Mech. Mach. Theory **41**(8), 958–970 (2006)
5. Bonev, I., Zlatanov, D., Gosselin, C.: Singularity analysis of 3-dof planar parallel mechanisms via screw theory. ASME J. Mech. Des. **125**(3), 573–581 (2003)
6. Briot, S., Martinet, P.: Minimal representation for the control of Gough-Stewart platforms via leg observation considering a hidden robot model. In: Proceedings of the 2013 IEEE International Conference on Robotics and Automation (ICRA 2013). Karlsruhe, Germany (2013)
7. Caro, S., Moroz, G., Gayral, T., Chablat, D., Chen, C.: Singularity analysis of a six-dof parallel manipulator using Grassmann-Cayley algebra and Groebner bases. In: Proceedings of the Symposium on Brain, Body and Machine. Montreal, QC, Canada (2010)
8. Chaumette, F.: The Confluence of Vision and Control, chap. Potential problems of stability and convergence in image-based and position-based visual servoing, pp. 66–78. No. 237 in LNCIS. Springer-Verlag (1998)
9. Espiau, B., Chaumette, F., Rives, P.: A new approach to visual servoing in robotics. IEEE Trans. Robot. Autom. **8**(3), 313–326 (1992)
10. Gosselin, C., Angeles, J.: Singularity analysis of closed-loop kinematic chains. IEEE Trans. Robot. Autom. **6**(3), 281–290 (1990)
11. Merlet, J.: Parallel Robots, 2nd edn. Springer, Dordrecht, The Netherlands (2006)
12. Rosenzveig, V., Briot, S., Martinet, P.: Minimal representation for the control of the Adept Quattro with rigid platform via leg observation considering a hidden robot model. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and System. (IROS 2013). Tokyo Big Sight, Japan (2013)
13. Rosenzveig, V., Briot, S., Martinet, P., Ozgur, E., Bouton, N.: A method for simplifying the analysis of leg-based visual servoing of parallel robots. In: Proceedings of 2014 IEEE International Conference on Robotics and Automation (ICRA 2014). Hong Kong, China (2014)

Impact of Perturbation on Wire Tension Vector

Leila Notash

Abstract The effect of uncertainties in design parameters and error in measurement on the Jacobian matrix and its null space basis of wire-actuated parallel manipulators is investigated to ensure positive wire tension. The minimum 2-norm non-negative solution for the vector of wire tensions is calculated utilizing the perturbed and the interval forms of Jacobian matrix. A planar manipulator is simulated to investigate the implementation and effectiveness of these two methodologies while relating their results.

Keywords Positive wire tension · Parameter uncertainty

1 Introduction

In wire/cable-actuated robot manipulators, also known as wire/cable-driven, the mobile platform (end effector) is connected to the base by wires/cables. Because wires act in tension, i.e., their inputs are unidirectional and irreversible, to fully constrain an m degrees of freedom (DOF) rigid body suspended by wires, in the absence of gravity and external force/moment (wrench), the number of wires/actuators should be larger than the DOF of manipulator (Fig. 1). Hence, there are infinite solutions for the wire tension vector for a given platform wrench as $n \geq m + 1$. Applying the generalized inverse (GI) of Jacobian matrix, the minimum 2-norm solution for the vector of wire tensions could result in negative tension for wires which is not acceptable. Homogeneous solution is used to adjust the tension to positive values if the platform position and orientation (pose) is within the wrench closure workspace. Implementation of the methodology of [1] for achieving minimum 2-norm positive

L. Notash (✉)
Department of Mechanical and Materials Engineering, Queen's University, Kingston, Canada
e-mail: notash@me.queensu.ca

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_5,

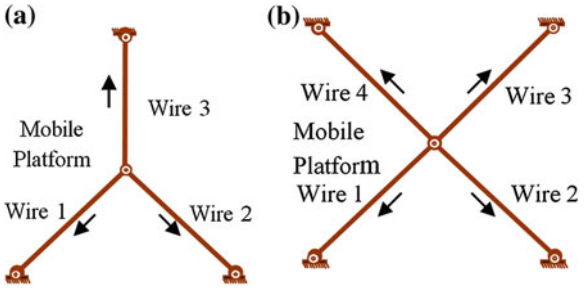


Fig. 1 2 DOF wire-actuated parallel manipulators with **a** three wires; and **b** four wires

wire tension was presented in [2]. The proposed novel method provides closed-form, minimum norm and continuous solution for positive wire tension. In most other publications, positive tension is identified with iterative methods (e.g., [3]), which are not suitable for real time applications and the calculated wire forces could be much larger than necessary. The closed-form procedure of [4] could fail in finding a solution though the solution exists, e.g., wire tension could exceed the upper limit as it is not the minimum norm solution. In Sect. 2, formulation of non-negative minimum 2-norm wire tension vector, in the presence of uncertainties in parameters and error in measurement, is discussed utilizing perturbation theory and interval arithmetic's. Simulation results are reported in Sect. 3. The article concludes with Sect. 4.

2 Non-negative Minimum Norm Wire Tension

2.1 Perturbed Jacobian Matrix

The perturbed transposed Jacobian matrix as a result of uncertainty in parameters is $\tilde{\mathbf{J}}^T = \mathbf{J}^T + \mathbf{E}$. For perfect case, $\mathbf{E} = \mathbf{0}$ and $\tilde{\mathbf{J}}^T = \mathbf{J}^T$. The $n \times 1$ vector of perturbed wire forces $\tilde{\boldsymbol{\tau}} = \left[\, \tilde{\tau}_1 \; \cdots \; \tilde{\tau}_n \,\right]^T$ is related to the $m \times 1$ wrench $\tilde{\mathbf{F}}$ applied by the platform with the $m \times n$ perturbed transposed Jacobian matrix $\tilde{\mathbf{J}}^T$ as

$$\tilde{\mathbf{F}} = \tilde{\mathbf{J}}^T \tilde{\boldsymbol{\tau}} = \left[\tilde{\mathbf{J}}_1^T \; \tilde{\mathbf{J}}_2^T \; \cdots \; \tilde{\mathbf{J}}_i^T \; \cdots \; \tilde{\mathbf{J}}_{n-1}^T \; \tilde{\mathbf{J}}_n^T\right] \tilde{\boldsymbol{\tau}} = \sum_{j=1}^{n} \tilde{\mathbf{J}}_j^T \tilde{\tau}_j \tag{1}$$

The solution for the perturbed linear system when the Moore-Penrose generalized inverse of $\tilde{\mathbf{J}}^T$, $\tilde{\mathbf{J}}^{\#T}$, is a continuous function of its entries is

$$\begin{aligned} \tilde{\boldsymbol{\tau}} &= \tilde{\boldsymbol{\tau}}_p + \tilde{\boldsymbol{\tau}}_h = \tilde{\mathbf{J}}^{\#T}\mathbf{F} + \left(\mathbf{I} - \tilde{\mathbf{J}}^{\#T}\tilde{\mathbf{J}}^T\right)\mathbf{k} = \tilde{\mathbf{J}}^{\#T}\mathbf{F} + \tilde{\mathbf{N}}\boldsymbol{\lambda} \\ \tau_{l\,\min} \le \tilde{\tau}_l &= \tilde{\tau}_{pl} + \tilde{\tau}_{hl} = \tilde{\tau}_{pl} + \sum_{j=1}^{n-m} \tilde{n}_{lj}\lambda_j \le \tau_{l\,\max} \quad \text{for} \quad l = 1, \ldots, n \end{aligned} \tag{2}$$

When $\text{rank}(\mathbf{J}^T) = \text{rank}(\tilde{\mathbf{J}}^T)$ then $\tilde{\mathbf{J}}^T$ is an acute perturbation of $\mathbf{J}^T$. Otherwise, $\|\tilde{\mathbf{J}}^{\#T} - \mathbf{J}^{\#T}\|_2 \geq 1/\|\mathbf{E}\|_2$,$\|\tilde{\mathbf{J}}^{\#T}\|_2 \geq 1/\|\mathbf{E}\|_2$; and $\|\tilde{\mathbf{J}}^T\|_2 = \sqrt{\lambda_{\max}(\tilde{\mathbf{J}}\tilde{\mathbf{J}}^T)} = \sigma_{\max}(\tilde{\mathbf{J}}^T)$ is the spectral norm, $\lambda_{\max}(\tilde{\mathbf{J}}\tilde{\mathbf{J}}^T)$ and $\sigma_{\max}(\tilde{\mathbf{J}}^T)$ are respectively the largest eigenvalue and singular value of $\tilde{\mathbf{J}}\tilde{\mathbf{J}}^T$ and $\tilde{\mathbf{J}}^T$ [5]. For the nonsingular square matrix $\mathbf{J}^T$, a relative perturbation of at least $\|\mathbf{E}\|/\|\mathbf{J}^T\| \geq \kappa^{-1}(\mathbf{J}^T)$ is required for $\tilde{\mathbf{J}}^T$ to become singular, where $\kappa(\mathbf{J}^T) = \|\mathbf{J}^T\| \, \|\mathbf{J}^{-T}\|$ is the condition number of $\mathbf{J}^T$ with respect to a consistent matrix norm, e.g., $\kappa_2(\mathbf{J}^T) = \sigma_{\max}/\sigma_{\min}$.

In Eq. (2), the homogeneous solution $\tilde{\boldsymbol{\tau}}_h = \left(\mathbf{I} - \tilde{\mathbf{J}}^{\#T}\tilde{\mathbf{J}}^T\right)\mathbf{k} = \tilde{\mathbf{N}}\boldsymbol{\lambda}$ vanishes when $\tilde{\mathbf{J}}^T$ is of full column rank as $\tilde{\mathbf{J}}^{\#T}\tilde{\mathbf{J}}^T = \mathbf{I}$. Columns of the $n \times (n-m)$ matrix $\tilde{\mathbf{N}}$, $\tilde{\mathbf{n}}_j$, $j = 1, \ldots, n-m$, correspond to the null space basis of $\tilde{\mathbf{J}}^T$, and $\boldsymbol{\lambda}$ is an $(n-m)$-vector. When one or more entries of the minimum 2-norm (particular) solution $\tilde{\boldsymbol{\tau}}_p = \tilde{\mathbf{J}}^{\#T}\tilde{\mathbf{F}}$ are negative the wire tensions could be adjusted by identifying the correctional tension $\tilde{\boldsymbol{\tau}}_h$ that would set all the wire tensions to positive values provided the manipulator pose is in the wrench closure workspace.The adjusted wire tensions should satisfy the tension limits $\mathbf{0} \leq \boldsymbol{\tau}_{\min} \leq \tilde{\boldsymbol{\tau}}_p + \tilde{\boldsymbol{\tau}}_h \leq \boldsymbol{\tau}_{\max}$.

The possibility of having positive wire tensions (pose being in the wrench closure workspace) could be investigated considering the orthonormal basis of the null space of the $m \times n$ perturbed transposed Jacobian matrix $\tilde{\mathbf{J}}^T$. The sufficient condition for ensuring positive tension is the existence of a null space vector of $\tilde{\mathbf{J}}^T$ with all positive entries. In the presence of external wrench, regardless of the null space vector of $\tilde{\mathbf{J}}^T$, positive wire tension is feasible if there exist a null space vector for the $m \times (n+1)$ augmented transposed Jacobian matrix $\tilde{\mathbf{J}}^T_{aug}$

$$\left[\tilde{\mathbf{J}}_1^T \; \tilde{\mathbf{J}}_2^T \; \cdots \; \tilde{\mathbf{J}}_{n-1}^T \; \tilde{\mathbf{J}}_n^T \;\; -\tilde{\mathbf{F}}\right] \begin{bmatrix} \tilde{\boldsymbol{\tau}} \\ 1 \end{bmatrix} = \tilde{\mathbf{J}}^T_{aug}\tilde{\boldsymbol{\tau}}_{aug} = \mathbf{0} \tag{3}$$

with non-negative values for the first n entries corresponding to wires and positive value for the $(n+1)$th entry corresponding to $\tilde{\mathbf{F}}$. A procedure for calculating the non-negative null space vectors of $\mathbf{J}^T$ and $\mathbf{J}^T_{aug}$ ($\tilde{\mathbf{J}}^T$ and $\tilde{\mathbf{J}}^T_{aug}$) is presented in [2].

When the minimum norm solution results in negative tension for wire i the change in its tension after adjusting the negative value to $\tilde{\tau}_{pi} + \tilde{\tau}_{hi} = \tilde{\tau}_{pi} + \sum_{j=1}^{n-m} \tilde{n}_{ij}\lambda_j = \tau_{ci} \geq \tau_{\min} \geq 0$ is $\left|\tilde{\tau}_{pi} - \tau_{ci}\right|$. When k wires have negative tension, after adjusting the negative tensions, the platform wrench that should be balanced by the remaining wires is $\sum_k \tilde{\mathbf{J}}_i^T(\tilde{\tau}_{pi} - \tau_{ci}) = \tilde{\mathbf{J}}^T(\tilde{\boldsymbol{\tau}}_p - \tilde{\boldsymbol{\tau}}_f)$, and $\tilde{\boldsymbol{\tau}}_f = [\tilde{\tau}_{p\,1} \; \tilde{\tau}_{p\,2} \; \ldots \; \tau_{ci} \; \ldots \; \tilde{\tau}_{p\,n-1} \; \tilde{\tau}_{p\,n}]^T$.

A method for calculating τ_{ci} such that the adjusted wire tensions do not violate the lower and upper limits, i.e., $\tau_{\min} \leq \tilde{\tau}_l \leq \tau_{\max}$, for $l = 1, \ldots, n$ was discussed in [2, 6]. For example, when the entries of the null space vector $\tilde{\mathbf{n}}$ have consistent signs (non-negative), while adjusting the negative tension of wires, e.g., wire i with $\tilde{\tau}_{pi} + \tilde{n}_i\lambda \geq \tau_{\min}$, the tension of wires with positive particular solution will be increased (will remain unchanged if the corresponding entry of $\tilde{\mathbf{n}}$ is zero). When wire k has the smallest $\lambda_k = (\tau_{\max} - \tilde{\tau}_{pk})/\tilde{n}_k$ among all wires with positive

particular solution, the adjusted tensions will not exceed the limit for any values of λ as long as

$$0 < \lambda_{dw} = \frac{\tau_{\min} + |\tilde{\tau}_{pi}|}{\tilde{n}_i} \leq \lambda \leq \lambda_k = \frac{\tau_{\max} - \tilde{\tau}_{pk}}{\tilde{n}_k} \quad \text{for} \quad \tilde{n}_i > 0 \text{ and } \tilde{n}_k > 0 \quad (4)$$

When more than one wire has negative tension, in Eq. (4) wire i corresponds to the dominating wire with λ_{dw}. On the other hand, when the minimum 2-norm solution results in a wire tension larger than the maximum value that tension could be set to the limit and the procedure is repeated. The upper limit will be fulfilled as long as there is sufficient redundancy. The necessary condition for exceeding the upper limit after adjustment is $\tilde{\tau}_{pk} \geq \tau_{\max} - \tilde{n}_k \lambda$. If $\tilde{n}_i$ and $\tilde{n}_k$ have opposite signs, $\tilde{n}_i > 0$ and $\tilde{n}_k < 0$, the upper limit is satisfied if $\tilde{\tau}_{pk} \leq \tau_{\max} + |\tilde{n}_k| \lambda$.

2.1.1 Adjusting Wire Tensions

Following the method in [1] for manipulators with no uncertainty in parameters, for the perturbed case, the minimum 2-norm "correctional" and overall wire force vectors for the calculated τ_{ci} [7] will be

$$\tilde{\boldsymbol{\tau}}_{corr} = \tilde{\mathbf{J}}_f^{\#T} \sum \tilde{\mathbf{J}}_i^T (\tilde{\boldsymbol{\tau}}_{pi} - \tau_{ci}) = \tilde{\mathbf{J}}_f^{\#T} \tilde{\mathbf{J}}^T (\tilde{\boldsymbol{\tau}}_p - \tilde{\boldsymbol{\tau}}_f) \quad (5)$$

$$\tilde{\boldsymbol{\tau}}_{tot} = \tilde{\boldsymbol{\tau}}_f + \tilde{\boldsymbol{\tau}}_{corr} = \tilde{\mathbf{J}}_f^{\#T} \tilde{\mathbf{J}}^T \tilde{\boldsymbol{\tau}}_p + (\mathbf{I} - \tilde{\mathbf{J}}_f^{\#T} \tilde{\mathbf{J}}^T) \tilde{\boldsymbol{\tau}}_f \quad (6)$$

where k columns of $\tilde{\mathbf{J}}^T$, corresponding to the wires with negative tension, are replaced by zeroes resulting in $\tilde{\mathbf{J}}_f^T$. These $\tilde{\boldsymbol{\tau}}_{corr}$ and $\tilde{\boldsymbol{\tau}}_{tot}$ are continuous as long as the GI is continuous, i.e., while $\tilde{\mathbf{J}}^T$ and $\tilde{\mathbf{J}}_f^T$ have locally constant rank. For full row-rank $\tilde{\mathbf{J}}_f^T$, the right-GI of $\tilde{\mathbf{J}}_f^T$ is $\tilde{\mathbf{J}}_f^{\#T} = \tilde{\mathbf{J}}_f (\tilde{\mathbf{J}}_f^T \tilde{\mathbf{J}}_f)^{-1}$ as the vector of wire forces is physically consistent. Otherwise, the weighted left-GI is used.

2.2 Interval Arithmetic's

When data/parameters are given as intervals, the interval Jacobian matrix is $\mathbf{J}^T = [\underline{\mathbf{J}}^T, \overline{\mathbf{J}}^T] = [\mathbf{J}_c^T - \Delta\mathbf{J}^T, \ \mathbf{J}_c^T + \Delta\mathbf{J}^T]$, where $\mathbf{J}_c^T$ is the midpoint transposed Jacobian matrix. $\mathbf{F} = [\underline{\mathbf{F}}, \ \overline{\mathbf{F}}]$ and $\boldsymbol{\tau} = [\underline{\boldsymbol{\tau}}, \ \overline{\boldsymbol{\tau}}]$ are interval vectors due to inexactness of data/parameters. If $\mathbf{J}^T = [\underline{\mathbf{J}}^T, \ \overline{\mathbf{J}}^T]$ does not include a singular matrix then it is regular. The problem then is the computation of the exact/sharp lower and upper bounds of $\boldsymbol{\tau}_p$ using $\mathbf{F} = \mathbf{J}^T \boldsymbol{\tau} = \sum_{j=1}^{n} \mathbf{J}_j^T \tau_j$, along with $\boldsymbol{\tau}_{corr}$ and $\boldsymbol{\tau}_{tot}$, with data/parameters varying independently of each other in the prescribed intervals, e.g., the interval

hull of solution sets. The dependency among entries of $\mathbf{J}^T$ may be addressed by preconditioning, e.g., with the inverse of $\mathbf{J}_c^T$, to tighten the solution set enclosure.

2.2.1 Adjusting Wire Tensions

When the $m \times n$ interval matrix $\mathbf{J}_f^T$ is full-row rank with $n > m$ then $\mathbf{J}_{fc}^T \mathbf{J}_{fc}^{\#T} = \mathbf{I}_m$, where in $\boldsymbol{\tau}_f$ and $\mathbf{J}_f^T$ the corresponding entries and columns relating to the wires with negative tension are replaced with degenerate null interval. The bounds for $\boldsymbol{\tau}_{corr}$ are calculated solving

$$\mathbf{J}_f^T \boldsymbol{\tau}_{corr} = \sum \mathbf{J}_i^T (\tau_{pi} - \tau_{ci}) = \mathbf{J}^T (\boldsymbol{\tau}_p - \boldsymbol{\tau}_f) \tag{7}$$

and $\boldsymbol{\tau}_{tot} = \boldsymbol{\tau}_f + \boldsymbol{\tau}_{corr}$. For $\mathbf{J}_f \mathbf{y} = \boldsymbol{\tau}_{corr}$, $\mathbf{J}_f^T \mathbf{J}_f \mathbf{y} = \mathbf{J}_f^T \tau_{corr} = \mathbf{J}^T (\boldsymbol{\tau}_p - \boldsymbol{\tau}_f)$ is solved for $\mathbf{y}$ then $\boldsymbol{\tau}_{corr} = \mathbf{J}_f \mathbf{y}$. Hence, for the under-determined linear system, the bounds for $\boldsymbol{\tau}_{corr}$ could also be calculated using the $(m+n) \times (m+n)$ extended relation [8]

$$\begin{bmatrix} \mathbf{J}_f & -\mathbf{I} \\ \mathbf{0} & \mathbf{J}_f^T \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \boldsymbol{\tau}_{corr} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{J}^T (\boldsymbol{\tau}_p - \boldsymbol{\tau}_f) \end{bmatrix} \quad \Rightarrow \quad \mathbf{J}_{ext}^T \boldsymbol{\tau}_{ext} = \mathbf{F}_{ext} \tag{8}$$

where $\boldsymbol{\tau}_{corr}$ corresponds to the last n entries of the $(m+n) \times 1$ vector $\boldsymbol{\tau}_{ext}$.

3 Case Study

For planar manipulators, the platform is connected to the base by n wires, each wire with a length of l_i and orientation of α_i (Fig. 2). The attachment points of wire i to the base and platform are denoted as points A_i and B_i, respectively. The angular positions of points B_i on the platform are denoted by θ_i. For a 2 DOF translational manipulator with four wires (Fig. 1b), the coordinates of A_i, $i = 1, \ldots, 4$, are (–2, –1.5), (2, –1.5), (2, 1.5) and (–2, 1.5) meters, respectively. Error bounds of ± 10 mm are considered for these coordinates and for the platform pose (when applicable). The null space bases of the 2×4 matrices $\mathbf{J}^T$ and $\tilde{\mathbf{J}}^T$ are spanned by two 4×1 vectors.

Example 1 At the platform pose of $\mathbf{p} = [0 \quad 0]^T$ in the wrench closure workspace and for $\mathbf{F} = [0 \quad 24]^T$ N, the minimum 2-norm vector of wire forces is $\boldsymbol{\tau}_p = \mathbf{J}^{\#T} \mathbf{F} = [-10 \quad -10 \quad 10 \quad 10]^T$ N with negative tension for wires 1 and 2 and a magnitude of $\|\boldsymbol{\tau}_p\|_2 = 20$. With uniformly distributed random error for A_i,

$$\tilde{\mathbf{J}}^T = \begin{bmatrix} \cos\tilde{\alpha}_1 & \cdots & \cos\tilde{\alpha}_4 \\ \sin\tilde{\alpha}_1 & \cdots & \sin\tilde{\alpha}_4 \end{bmatrix} = \begin{bmatrix} -0.7985 & 0.7977 & 0.8015 & -0.8023 \\ -0.6021 & -0.6030 & 0.5979 & 0.5969 \end{bmatrix} \tag{9}$$

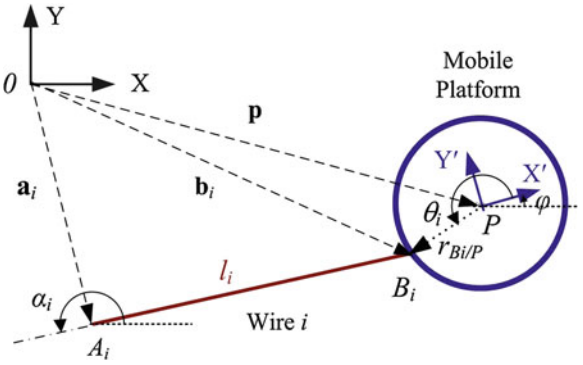


Fig. 2 Parameters of planar wire-actuated parallel manipulators

$\tilde{\mathbf{J}}^T$ is an acute perturbation of non-singular $\mathbf{J}^T$ (reported as Eq. (13) of [2]) since $\|\mathbf{E}\|_2 / \|\mathbf{J}^T\|_2 = 0.0035 < \kappa_2^{-1}(\mathbf{J}^T) = 0.7500$. The minimum 2-norm perturbed wire tension vector also includes negative tension for wires 1 and 2

$$\tilde{\boldsymbol{\tau}}_p = \tilde{\mathbf{J}}^{\#T}\mathbf{F} = [-10.0342 \;\; -10.0510 \;\; 9.9655 \;\; 9.9491]^T \tag{10}$$

The non-negative null space vectors of $\tilde{\mathbf{J}}^T$ and $\tilde{\mathbf{J}}^T_{aug}$ are

$$\tilde{\mathbf{N}} = \left[\tilde{\mathbf{n}}_1 \;\; \tilde{\mathbf{n}}_2\right] = \begin{bmatrix} 0 & 0.9985 \\ 0.9978 & 0 \\ 0.0080 & 1.0000 \\ 1.0000 & 0.0054 \end{bmatrix} \tag{11}$$

$$\tilde{\mathbf{N}}^T_{aug} = [0 \;\; 0 \;\; 0.7053 \;\; 0.7080 \;\; 0.0357] \tag{12}$$

For $\tau_{c1} = \tau_{c2} = 1\text{N}$, $\tilde{\boldsymbol{\tau}}_f = [1 \;\; 1 \;\; 9.9655 \;\; 9.9491]^T$. To produce the wrench $\mathbf{F}$

$$\tilde{\boldsymbol{\tau}}_{corr} = \tilde{\mathbf{J}}_f^{\#T}\tilde{\mathbf{J}}^T(\tilde{\boldsymbol{\tau}}_p - \tilde{\boldsymbol{\tau}}_f) = [0 \;\; 0 \;\; 11.1393 \;\; 11.1353]^T \tag{13}$$

$$\tilde{\boldsymbol{\tau}}_{tot} = [1.0 \;\; 1.0 \;\; 21.1048 \;\; 21.0843]^T \tag{14}$$

with $\|\tilde{\boldsymbol{\tau}}_p\|_2 = 20.0001$, $\|\tilde{\boldsymbol{\tau}}_{corr}\|_2 = 15.7505$, $\|\tilde{\boldsymbol{\tau}}_{tot}\|_2 = 29.8657$.

If the wire tension limits are set at $\tau_{\min} = 1\,\text{N}$ and $\tau_{\max} = 30\,\text{N}$, then $1 \le \tau_{c1} \le 9.8$ and $1 \le \tau_{c2} \le 9.8$ would satisfy the tension limits, e.g., for $\tau_{c1} = \tau_{c2} = 9.8\,\text{N}$

$$\tilde{\boldsymbol{\tau}}_{tot} = [9.8000 \;\; 9.8000 \;\; 29.9885 \;\; 29.9515]^T \tag{15}$$

with $\left\|\tilde{\boldsymbol{\tau}}_{tot}\right\|_2 = 44.5924$. While for $\tau_{c1} = 1$ N,$\tau_{c2} = 9.8$ N, $\|\boldsymbol{\tau}_{tot}\|_2 = 37.9431$ for

$$\tilde{\boldsymbol{\tau}}_{tot} = [1.0 \;\; 9.8 \;\; 21.1750 \;\; 29.9042]^T \tag{16}$$

When A_i are treated as interval arguments, original coordinates are taken as the midpoint of corresponding interval. With an interval radius of 10 mm for A_i coordinates, e.g., with infimum-supremum representation for the coordinates of A_1 $\big([-2.0100, -1.9999]; [-1.5100, -1.4999]\big)$, the interval $\mathbf{J}^T$ is

$$\mathbf{J}^T = \begin{bmatrix} \begin{bmatrix} -0.8085, & -0.7916 \\ -0.6074, & -0.5927 \end{bmatrix} & \begin{bmatrix} 0.7916, & 0.8085 \\ -0.6074, & -0.5927 \end{bmatrix} & \begin{bmatrix} 0.7916, & 0.8085 \\ 0.5927, & 0.6074 \end{bmatrix} & \begin{bmatrix} -0.8085, & -0.7916 \\ 0.5927, & 0.6074 \end{bmatrix} \end{bmatrix} \tag{17}$$

with $\mathbf{J}_c^T = \begin{bmatrix} -0.8000 & 0.8000 & 0.8000 & -0.8000 \\ -0.6000 & -0.6000 & 0.6000 & 0.6000 \end{bmatrix}$ as the Jacobian matrix for perfect case. For degenerate interval $\mathbf{F} = [[0, 0][24, 24]]^T$ N, $\boldsymbol{\tau}_p = \mathbf{J}^{\#T}\mathbf{F} = [[-10.6227, -9.3792] [-10.6227, -9.3792] [9.3792, 10.6227] [9.3792, 10.6227]]^T$ and $\left\|\boldsymbol{\tau}_p\right\|_2 = [18.7584, 21.2454]$. Using the "verifylss" routine with "SharpIVmult" option of INTLAB software, the tighter bounds of the minimum 2-norm solution of under-determined linear system and the minimum norm non-negative solution are

$$\boldsymbol{\tau}_p = [[-10.3646, -9.6340] [-10.3646, -9.6340] [9.6340, 10.3646] [9.6340, 10.3646]]^T \tag{18}$$

$$\boldsymbol{\tau}_{corr} = [[0.0, 0.0] [0.0, 0.0] [8.2286, 13.7790] [8.2286, 13.7790]]^T \tag{19}$$

$$\boldsymbol{\tau}_{tot} = [[1.0, 1.0] [1.0, 1.0] [17.8626, 24.1436] [17.8626, 24.1436]]^T \tag{20}$$

with interval norms $\left\|\boldsymbol{\tau}_p\right\|_2 = [19.2680, 20.7293]$, $\|\boldsymbol{\tau}_{corr}\|_2 = [11.6370, 19.4864]$ and $\|\boldsymbol{\tau}_{tot}\|_2 = [25.3011, 34.1735]$.

As it is evidenced from the results, for the error bounds of ± 10 mm and the interval radius of 10 mm, the interval vectors $\boldsymbol{\tau}_p$, $\boldsymbol{\tau}_{corr}$ and $\boldsymbol{\tau}_{tot}$ encompass the corresponding $\tilde{\boldsymbol{\tau}}_p$, $\tilde{\boldsymbol{\tau}}_{corr}$ and $\tilde{\boldsymbol{\tau}}_{tot}$ vectors identified using the perturbed Jacobian matrix.

Example 2 At the platform pose of $\tilde{\mathbf{p}} = [2.0020 \;\; 0.0010]^T$ meters, with nominal X coordinate identical to those for the anchors of wires 2 and 3,

$$\tilde{\mathbf{J}}^T = \begin{bmatrix} \cos\tilde{\alpha}_1 & \cdots & \cos\tilde{\alpha}_4 \\ \sin\tilde{\alpha}_1 & \cdots & \sin\tilde{\alpha}_4 \end{bmatrix} = \begin{bmatrix} -0.9356 & -0.0030 & -0.0031 & -0.9373 \\ -0.3530 & -0.9999 & 0.9999 & 0.3485 \end{bmatrix} \tag{21}$$

For $\tilde{\mathbf{F}} = [-47.0007 \;\; 20.4097]^T$ N, the minimum 2-norm vector of wire forces is $\tilde{\boldsymbol{\tau}}_p = \tilde{\mathbf{J}}^{\#T}\tilde{\mathbf{F}} = [21.7487 \;\; -9.1042 \; 9.2667 \; 28.2376]^T$ N, with negative tension for wire 2. While the non-negative null space vector of $\mathbf{J}^T$is $\mathbf{n} = [0 \; 1 \; 1 \; 0]^T$, the non-negative $\tilde{\mathbf{n}}$ could not be calculated. The non-negative $\tilde{\mathbf{n}}_{aug}$ is

$$\tilde{\mathbf{n}}_{aug}^{T} = [0.3983 \quad 0.3972 \quad 0.7244 \quad 0.3983 \quad 0.0159]^{T} \tag{22}$$

The tension of wire 2 is set to $\tau_{c_2} = \tau_{\min} = 1$ N. To produce the wrench $\tilde{\mathbf{F}}$

$$\tilde{\boldsymbol{\tau}}_{corr} = \tilde{\mathbf{J}}_{f}^{\#T} \tilde{\mathbf{J}}_{2}^{T} (\tilde{\tau}_{p2} - \tau_{c2}) = [-2.8764 \quad 0 \quad 8.1087, \quad 2.8122]^{T} \tag{23}$$

$$\tilde{\boldsymbol{\tau}}_{tot} = \tilde{\boldsymbol{\tau}}_{f} + \tilde{\boldsymbol{\tau}}_{corr} = [18.8722, \quad 1.0 \quad 17.3754, \quad 31.0497]^{T} \tag{24}$$

and $\left\|\tilde{\boldsymbol{\tau}}_{p}\right\|_{2} = 37.9358$, $\left\|\tilde{\boldsymbol{\tau}}_{corr}\right\|_{2} = 9.0517$, $\left\|\tilde{\boldsymbol{\tau}}_{tot}\right\|_{2} = 40.2884$. If $\tau_{\max} = 30$ N the procedure is repeated for $\tau_{c2} = 1$ and $\tau_{c4} = 30$ with $\left\|\tilde{\boldsymbol{\tau}}_{tot}\right\|_{2} = 40.3224$ for

$$\tilde{\boldsymbol{\tau}}_{tot} = \tilde{\boldsymbol{\tau}}_{f} + \tilde{\boldsymbol{\tau}}_{corr} = [19.9215 \quad 1.0 \quad 18.1117 \quad 30.0]^{T} \tag{25}$$

With an interval radius of 10 mm for A_i coordinates, the interval $\mathbf{J}^{T}$ is

$$\begin{bmatrix} [-0.9468, -0.9260] & [-0.0136, \ 0.0136] & [-0.0136, \ 0.0136] & [-0.9468, -0.9260] \\ [-0.3580, -0.3443] & [-1.0271, -0.9735] & [0.9735, 1.0271] & [\,0.3443, \ 0.3580] \end{bmatrix} \tag{26}$$

Then, the bounds of the minimum 2-norm solutions are

$$\boldsymbol{\tau}_{p} = [[20.6961, 22.8852]\,[-10.2962, -7.9789]\,[7.9789, 10.2962] \\ [27.1116, 29.3008]]^{T} \tag{27}$$

$$\boldsymbol{\tau}_{corr} = [[-7.1935, 1.4658]\,[0.0, \ 0.0]\,[3.3443, 12.9713] \\ [-1.4658, 7.1935]]^{T} \tag{28}$$

$$\boldsymbol{\tau}_{tot} = [[13.5027, 24.3510]\,[1.0, \ 1.0]\,[11.3232, 23.2675] \\ [25.6459, 36.4942]]^{T} \tag{29}$$

with interval norms $\left\|\boldsymbol{\tau}_{p}\right\|_{2} = [35.9263, 39.9285]$, $\|\boldsymbol{\tau}_{corr}\|_{2} = [0.0000, 16.4847]$ and $\|\boldsymbol{\tau}_{tot}\|_{2} = [31.1328, 49.6706]$. For wire tension limits of $\tau_{\min} = 1$ N and $\tau_{\max} = 30$ N, the interval for the tension of wire 4 is redefined as $\tau_{tot_4} = [25.6459, 30.0000]$.

4 Concluding Remarks

The effect of uncertainties in parameters and error in measurement on the minimum 2-norm particular and non-negative solutions for the break wire tension vector of

wire-actuated parallel manipulators was investigated. Matrix perturbation and interval arithmetic's were applied. The perturbation theory integrates a chosen set of distinct uncertainties in each parameter/datum, while the interval arithmetic's provides evidence on the performance of manipulator within the taken uncertainty/error interval for each parameter/datum and possibility of ensuring non-negative wire tension in the presence of uncertainty. An example planar manipulator was used to illustrate the methods, their implementations and results.

References

1. Notash, L.: Failure recovery for wrench capability of wire-actuated parallel manipulators. Robotica **30**(6), 941–950 (2012)
2. Notash, L.: Designing positive tension for wire-actuated parallel manipulators. In: Advances in Mechanisms, Robotics and Design Education and Research, pp. 251–263. Springer (2013)
3. Hassan, M., Khajepour, A.: Optimization of actuator forces in cable-based parallel manipulators using convex analysis. IEEE Trans. Rob. **24**, 736–740 (2008)
4. Pott, A., Bruckmann, T., Mikelsons, L.: Closed-form force distribution for parallel wire robots. In: Computational Kinematics, pp. 25–34. Springer-Verlag, Duisburg (2009)
5. GW, Stewart, Sun, J.-G.: Matrix Perturbation Theory. Academic Press, Boston (1990)
6. Notash, L.: On the minimum 2-Norm positive tension for wire-actuated parallel manipulators. In: Computational Kinematics, pp. 1–9. Springer, Dordrecht (2014)
7. Notash, L.: Wrench recovery for wire-actuated parallel manipulators, RoManSy 19-Robot design. In: Dynamics and Control, pp. 201–208. Springer (2013)
8. Rump, S.M.: Verified bounds for least squares problems and underdetermined linear system. SIAM J. Matrix Anal. Appl. **33**, 130–148 (2012)

A Deployable Parallel Wrist with Simple Kinematics

Raffaele Di Gregorio

Abstract The parallel-wrist (PW) architectures that can be manufactured from a single sheet and, then, packed into a multi-layer configuration are interesting for many applications that range from MEMS to space. If suitably sized, a single-loop RU-RRRS topology allows to devise such a PW. Here, this PW architecture will be presented together with its kinematics analysis.

Keywords Parallel wrist · Single-loop · Deployable architecture · Position analysis

1 Introduction

Parallel manipulators' (PMs') possibility of locating actuators on or near to the base makes them lighter and faster than their serial counterparts. This possibility mainly relies on the adoption of multi-loop topologies. Nevertheless, if only three degrees-of-freedom (dof) are required, some single-loop topologies still keep this possibility. Single-loop architectures, with respect to multi-loop ones, usually have a wider workspace and, very often, can easier match some appealing requirements.

Among these requirements, being foldable/deployable makes transportation and in loco installation easier and cheaper, which is highly appreciated in space applications; also, it sometimes allows the machine to be manufactured from a single layer of suitable material as MEMS technology usually requires.

Parallel wrists (PWs) are 3-dof PMs where the platform can only change its orientation with respect to the base. The majority of the PW architectures proposed in the literature have two or three loops even though PWs' single-loop topologies have been also listed (see [4], for details and references). Only a limited number of the proposed PW architectures have been analyzed in depth.

R. Di Gregorio (✉)
Department of Engineering, University of Ferrara, Ferrara, Italy
e-mail: raffaele.digregorio@unife.it

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_6,

Here, a single-loop PW architecture which can be folded/deployed is presented and studied. The proposed architecture has two actuators on the base and the third one adjacent to the base. Also, it can be manufactured from a single sheet since it can be deployed on a plane and contains only revolute (R) pairs.

2 Single-Loop RU-RRRS

Figure 1 shows a particular RU-RRRS[1] architecture at a flattened configuration, where the U joint and the S pair are obtained through R pairs in series with orthogonal and intersecting axes. This architecture is a single-loop 9-link/9-(R) pair mechanism that, at the flattened configuration, has 6 dof: the 3 finite dof it has at non-flattened and non-folded configurations, plus 3 transitory dof [1] due to the possibility of folding the mechanism with finite rotations around the two vertical dash-dot lines, by keeping the three actuated R pairs locked, and to the possibility of exiting out of the plane up or down the axis of the passive (i.e., non-actuated) R pair of the RRRS subchain has. Moreover, if the actuated R pair with horizontal axis is not locked, a third folding possibility arises since finite rotations around the horizontal dash-dot line are allowed. According to how it is folded, the mobility of this mechanism changes. Such a property makes this mechanism a kinematotropic linkage [2, 3, 5].

The existence of a flattened configuration is a necessary condition both to manufacture the machine starting from a sheet of suitable material, and to have foldable RU-RRRS architectures. Different RU-RRRS architectures which lose foldability still keeping a flattened configuration, or lose even the flattened configuration can be obtained by suitably changing the sizes of the links.

Figure 2 shows a more general RU-RRRS architecture at a non-flattened configuration together with the notations that will be used. In this figure, the three R pairs that constitute the S pair centered at B (see Fig. 1) are not represented. In Figs. 1 and 2, points O, A and B, and the two links denoted base and platform are the same reference points/links and allow to recognize which R-pair axis corresponds to which other in the two figures. In particular, a Cartesian reference system $Ox_by_bz_b$,[2] fixed to the base, has been introduced with origin at O and the x_b (z_b) coordinate axis that coincides with the horizontal (vertical) actuated R-pair axis embedded in the base. The unit vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ indicate the directions of the two R-pair axes of the U joint; whereas, the unit vector $\mathbf{v}$ indicates the direction of the two parallel R-pair axes of the RRRS subchain. The angles θ_0, φ_0 and ψ_0 are the three actuated-joint variables; the angles θ_1, θ_2 and ψ_1 are the passive-joint variables; whereas, a, b, c, d, γ, and ε are links' geometric constants.

[1] R, U and S stand for revolute pair, universal joint and spherical pair, respectively. The underscore denotes actuated pairs.

[2] Hereafter, $\mathbf{i}_b$, $\mathbf{j}_b$ and $\mathbf{k}_b$ will denote the unit vectors of the coordinate axes x_b, y_b and z_b, respectively.

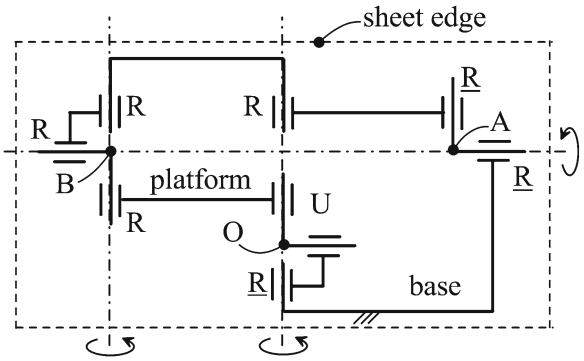


Fig. 1 Fodable/Deployable RU-RRRS wrist

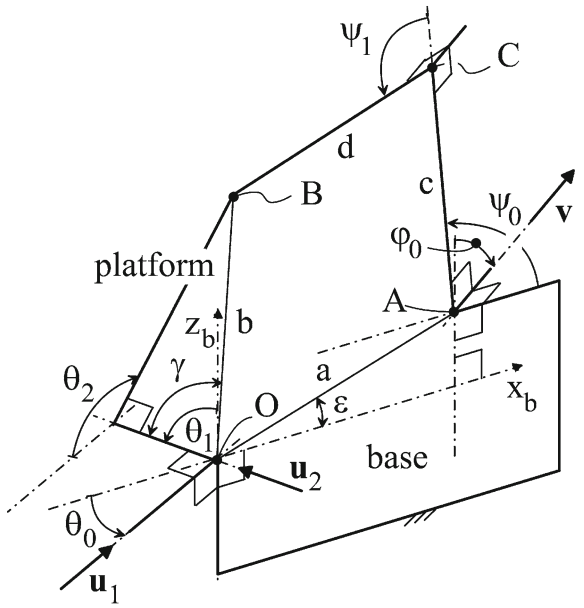


Fig. 2 RU-RRRS wrist: notations

With reference to Fig. 2, if the links are so sized that $a \sin \varepsilon = b \cos \gamma$, $a \cos \varepsilon = c$ and $b \sin \gamma = d$, the RU-RRRS architecture can assume the flattened and foldable configuration shown in Fig. 1; instead, if $a \sin \varepsilon = b \cos \gamma$, $a \cos \varepsilon + b \sin \gamma = c + d$ and the axes of the three R pairs that constitute the S pair centered at B are not mutually orthogonal, it can assume a flattened configuration that cannot be folded.

Eventually, since point O is embedded both in the platform and in the base, when the actuated-joint variables change their values only the platform orientation can vary; hence, this mechanism is a wrist.

3 Position Analysis

The kinematics behavior of the RU-RRRS wrist shown in Fig. 2 can be easily understood by realizing that points A, B and C always lie on the plane perpendicular to the unit vector $\mathbf{v}$ that contains the line (A, $\mathbf{i}_b$).[3]

The adopted notations (see Fig. 2) allows the following relationships to be written

$$\mathbf{v} = \mathbf{k}_b \cos\varphi_0 - \mathbf{j}_b \sin\varphi_0, \qquad \mathbf{u}_1 = \mathbf{i}_b \cos\theta_0 + \mathbf{j}_b \sin\theta_0, \tag{1a}$$

$$\mathbf{u}_2 = \mathbf{k}_b \cos\theta_1 + (\mathbf{k}_b \times \mathbf{u}_1)\sin\theta_1, \quad (\mathbf{B}-\mathbf{O}) = (\mathbf{B}-\mathbf{C}) + (\mathbf{C}-\mathbf{A}) + (\mathbf{A}-\mathbf{O}), \tag{1b}$$

$$(\mathbf{A}-\mathbf{O}) = a(\mathbf{i}_b \cos\varepsilon + \mathbf{k}_b \sin\varepsilon), \quad (\mathbf{C}-\mathbf{A}) = \mathbf{i}_b c\cos\psi_0 + (\mathbf{v}\times\mathbf{i}_b)c\sin\psi_0, \tag{1c}$$

$$(\mathbf{B}-\mathbf{C}) = \mathbf{i}_b d\cos(\psi_0+\psi_1) + (\mathbf{v}\times\mathbf{i}_b)d\sin(\psi_0+\psi_1) \tag{1d}$$

and, also,

$$(\mathbf{B}-\mathbf{O}) = \mathbf{u}_2 b\cos\gamma + b\sin\gamma[(\mathbf{u}_1\times\mathbf{u}_2)\sin\theta_2 - \mathbf{u}_1\cos\theta_2] \tag{2}$$

The position analysis refers to the solution of two problems: inverse position analysis (IPA) and direct position analysis (DPA). Here, the IPA is the determination of the actuated-joint variables, θ_0, φ_0 and ψ_0, for an assigned platform orientation; whereas, the DPA is the determination of the platform orientations compatible with assigned values of the actuated-joint variables.

In the IPA, since the platform orientation is known, the coordinates, $(\mathrm{x}_B, \mathrm{y}_B, \mathrm{z}_B)^{\mathrm{T}}$, of point B and the components, $(\mathrm{u}_{2x}, \mathrm{u}_{2y}, \mathrm{u}_{2z})^{\mathrm{T}}$, of unit vector $\mathbf{u}_2$ in $Ox_by_bz_b$ are known, too. Therefore, the first two actuated-joint variables, θ_0 and φ_0, can be computed as follows

$$\mathbf{u_1} = \pm\frac{\mathbf{k}_b\times\mathbf{u}_2}{\|\mathbf{k}_b\times\mathbf{u}_2\|} \equiv (\mathrm{u}_{1x}, \mathrm{u}_{1y}, 0)^{\mathrm{T}}, \qquad \theta_0 = \mathrm{atan2}(\mathrm{u}_{1y}, \mathrm{u}_{1x}) \tag{3a}$$

$$\mathbf{v} = \pm\frac{\mathbf{i}_b\times(\mathbf{B}-\mathbf{A})}{\|\mathbf{i}_b\times(\mathbf{B}-\mathbf{A})\|} \equiv (0, \mathrm{v}_y, \mathrm{v}_z)^{\mathrm{T}}, \qquad \varphi_0 = \mathrm{atan2}(-\mathrm{v}_y, \mathrm{v}_z) \tag{3b}$$

It is worth noting that the double determination of $\mathbf{u}_1$ and $\mathbf{v}$ in Eq. (3) yields two values of θ_0 and of φ_0 that differ of π radians.

Then, the third actuated-joint variable, ψ_0, can be computed by considering that point C must simultaneously lies on two coplanar circles of the plane points A, B and C belong to, one centered at B with radius d and the other centered at A with radius c. From an analytic point of view, the two values of ψ_0 that solve this problem (see Fig. 3) can be determined by writing $(\mathbf{B}-\mathbf{C})\cdot(\mathbf{B}-\mathbf{C}) = d^2$ where $\mathbf{B} = (\mathrm{x}_B, \mathrm{y}_B, \mathrm{z}_B)^{\mathrm{T}}$ and, according to Eq. (1c), $\mathbf{C} = (a\cos\varepsilon + c\cos\psi_0, c\sin\psi_0\cos\varphi_0, a\sin\varepsilon +$

[3] Here, a line passing through point P and with the direction of the vector $\mathbf{w}$ is denoted (P, $\mathbf{w}$).

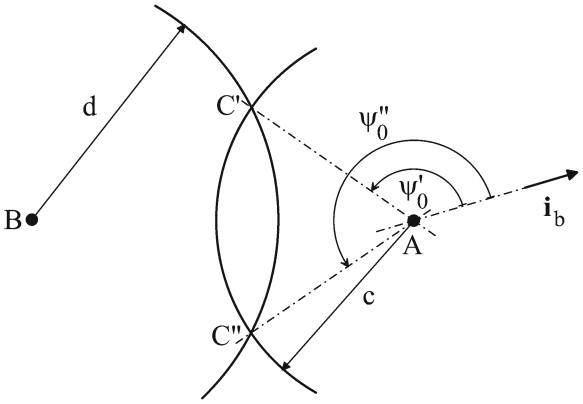


Fig. 3 Point C must simultaneously lies on two coplanar circles

$c \sin\psi_0 \sin\varphi_0)^{\mathrm{T}}$. So doing, after some rearrangements, the following scalar equation in ψ_0 is obtained

$$\begin{aligned} &\mathrm{x}_B(a\cos\varepsilon + c\cos\psi_0) + \mathrm{y}_B c\sin\psi_0\cos\varphi_0 + \mathrm{z}_B(a\sin\varepsilon + c\sin\psi_0\sin\varphi_0) \\ &\quad - ac(\cos\varepsilon\cos\psi_0 + \sin\varepsilon\sin\psi_0\sin\varphi_0) - \frac{a^2+b^2+c^2-d^2}{2} = 0 \end{aligned} \tag{4}$$

Equation (4) can be transformed into a quadratic equation in $\tan(\frac{\psi_0}{2})$ which gives two values of ψ_0 for each value of φ_0 computed through Eq. (3b).

In short, since Eq. (3) provide two values both of θ_0 and of φ_0 and, then, Eq. (4) yields two values of ψ_0 for each φ_0 value, the IPA solutions, (θ_0, φ_0, ψ_0), are at most eight.

In the DPA, since the actuated-joint variables, θ_0, φ_0 and ψ_0, are known, the vectors $\mathbf{v}$, $\mathbf{u}_1$ and $(\mathbf{C}-\mathbf{A})$ are known (see Eq. (1)), too. Also, the platform orientation can be determined by computing the coordinates of point B and the components of unit vector $\mathbf{u}_2$ in $Ox_by_bz_b$, which, according to Eq. (1), simply means calculating the values of the angles ψ_1, for B's coordinates, and θ_1, for $\mathbf{u}_2$'s components.

The values of the angle ψ_1 can be computed by writing $(\mathbf{B}-\mathbf{O})\cdot(\mathbf{B}-\mathbf{O}) = b^2$ where the right-hand side of the second Eq. (1b) is substituted for $(\mathbf{B}-\mathbf{O})$. So doing, after some rearrangement, the following scalar equation in ψ_1 is obtained

$$\begin{aligned} &d\cos(\psi_0+\psi_1)(a\cos\varepsilon + c\cos\psi_0) + d\sin(\psi_0+\psi_1)(c\sin\psi_0 + a\sin\varphi_0\sin\varepsilon) \\ &\quad + ac(\cos\varepsilon\cos\psi_0 + \sin\varepsilon\sin\psi_0\sin\varphi_0) + \frac{a^2+c^2+d^2-b^2}{2} = 0 \end{aligned} \tag{5}$$

Equation (5) can be transformed into a quadratic equation in $\tan(\frac{\psi_0+\psi_1}{2})$ which gives two values of ψ_1 whose introduction into the right-hand side of the second Eq. (1b) yields as many values of $(\mathrm{x}_B, \mathrm{y}_B, \mathrm{z}_B)^{\mathrm{T}}$.

Once the two positions point B can assume have been computed, the values of θ_1 that solve the DPA can be calculated through the equation $(\mathbf{B}-\mathbf{O})\cdot\mathbf{u}_2 = b\cos\gamma$ (see Eq. (2) and Fig. 2) written as follows

$$\sin\theta_1(\mathrm{y}_B\cos\theta_0 - \mathrm{x}_B\sin\theta_0) + \mathrm{z}_B\cos\theta_1 - b\cos\gamma = 0 \tag{6}$$

where the expression given by the first Eq. (1b) is used for the unit vector $\mathbf{u}_2$.

Equation (6) can be transformed into a quadratic equation in $\tan(\frac{\theta_1}{2})$. The solution of such an equation yields at most two values of θ_1, for each possible position of B, whose introduction into the right-hand side of the first Eq. (1b) allows the determination of as many $\mathbf{u}_2$'s directions. Therefore, there are at most four platform orientations compatible with assigned values of the actuated-joint variables (i.e., there are at most four DPA solutions).

The way the DPA has been solved highlights that the position of point B is controlled only by the two actuated-joint variables of the RRRS subchain (i.e., φ_0 and ψ_0); whereas, the changes of the remaining actuated-joint variable, θ_0 (i.e., the one of the RU subchain), just make the platform rotate around the line passing through points O and B (see Fig. 2). This feature makes the proposed wrist an ideal candidate to be a mechanical shoulder where the platform is the humerus with the line through O and B as axis, the RRRS subchain that controls the position of the elbow and the RU subchain that controls the rotation of the whole arm around the humerus axis.

From a geometric point of view, point B is constrained to simultaneously lie on the plane through A and perpendicular to $\mathbf{v}$, due to the RRRS subchain, and on the sphere centered at O with radius b, due to the RU subchain. Hence, it must lie on the intersection curve between these two surfaces, that is, on the circle with radius $\mathrm{r} = \sqrt{\mathrm{b}^2 - [(\mathbf{A}-\mathbf{O})\cdot\mathbf{v}]^2} \equiv \sqrt{\mathrm{b}^2 - \mathrm{a}^2\sin^2\varepsilon\cos^2\varphi_0}$ (see Eq. (1)), centered at the foot, D, of the perpendicular through O to the above-mentioned plane. If φ_0 does not change (i.e., the direction of $\mathbf{v}$ does not change), an arc of such a circle is the path point B describes during the variation of ψ_0. It coincides with the path of the free ending of a rocker hinged to the frame at D whose length is r. Thus, in this case, a planar four-bar linkage can be identified with points A and D as hinge centers at the frame, and the segments AC, CB and DB as crank, coupler and rocker, respectively, with ψ_0 that plays the role of crank angle (see Fig. 4) and with the plane through A and perpendicular to $\mathbf{v}$ that is the plane of motion.

With reference to this four-bar linkage, since φ_0 and ψ_0 are assigned in the DPA, the positions of points D and C are at rest with respect to the frame, and the determination of the B positions that solve the DPA reduces itself to the determination of the configurations that a four-bar linkage can assume for an assigned value of the crank angle. It is well known that such configurations are at most two and can be easily determined through the graphical construction reported in Fig. 5.

In a few words, the motion of the proposed PW architecture reduces itself to the one of a four-bar linkage where the length of the rocker and of the frame bar together

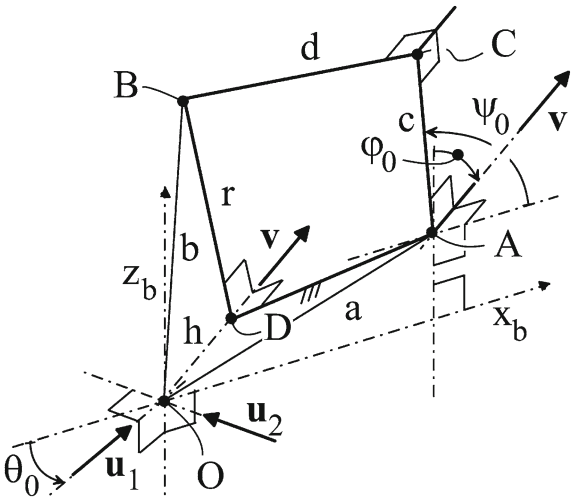


Fig. 4 Four-bar linkage lying on the plane that passes through A and is perpendicular to $\mathbf{v}$ ($h = (\mathbf{A} - \mathbf{O}) \cdot \mathbf{v} \equiv a \sin\varepsilon \cos\varphi_0$)

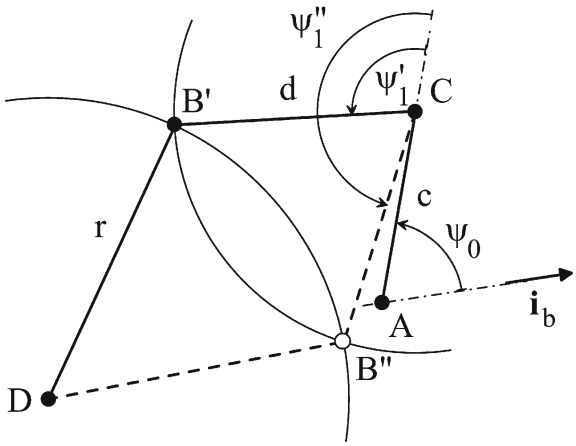


Fig. 5 DPA solution: determination of the positions point B can assume

with the attitude of the motion plane can be varied in a controllable way. The fact that the motion of this mechanism is so clearly defined through well-identified geometric relationships between actuated-joint variables and motion characteristics is certainly of interest for applications that goes over the wrist.

4 Conclusions

Even though systematic synthesis techniques have been used to find all the PW topologies, and such topologies have been practically all identified, many kinematic aspects still remain to be discovered on PWs since only a few of the identified PW types have been studied in depth.

This consideration brought the author to investigate the kinematics of single-loop PWs, with actuators on or near to the base, till to identify some interesting PW architectures among which the RU-RRRS one presented here.

The presented PW architecture can match requirements that range from MEMS to space applications. Its position analysis problems can be solved in closed form with relationships between actuated-joint variables and resulting platform motion that allows to visualize the effect on the platform motion of each variable.

Acknowledgments This work has been developed at the Laboratory of Advanced Mechanics (MECH-LAV) of Ferrara Technopole, supported by UNIFE funds and by Regione Emilia Romagna (District Councillorship for Productive Assets, Economic Development, Telematic Plan) POR-FESR 2007-2013, Attività I.1.1.

References

1. Davidson, J.K., Hunt, K.H.: Robots and Screw Theory: Applications of Kinematics and Statics to Robotics. Oxford University Press, Oxford, UK (2004)
2. Galletti, C., Fanghella, P.: Single-loop kinematotropic mechanisms. Mechanism and Machine Theory **36**(6), 743–761 (2001)
3. Huang, H.H.: Graph representation of the variable chain mechanisms with sequential movement. Proc. of the 13th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI/ISAS 2009), Orlando, Florida (USA), June 10–13, 2009, url: http://www.iiis.org/CDs2009/CD2009SCI/SCI2009/PapersPdf/S670OM.pdf
4. Kong, X., Gosselin, C.M.: Type Synthesis of Parallel Mechanisms. Springer-Verlag, Berlin Heidelberg, Germany (2007)
5. Wohlhart, K.: Kinematotropic linkages. Recent Advances in Robot Kinematics, Lenarčič, J. and Parenti-Castelli, V. (eds.), Kluwer Academic Publishers, Dordrecht, pp. 359–368 (1996)

Geometric Derivation of 6R Linkages with Circular Translation

Chung-Ching Lee and Jacques M. Hervé

Abstract A mathematical explanation on the circular translation is briefly presented and the planar-hinged parallelogram is introduced as the simplest generator of circular translational motions without using prismatic pairs. Based on the geometric properties of circular translation and spherical translation, a kind of 6R paradoxical linkage with a relative motion of circular translation is derived by a new purely geometric approach. The already published linkage having a link with the translation property is verified and other existing ones are distinguished and compared by the use of the parameterization of linkage architecture too.

Keywords Circular translation · Parallelogram · Paradoxical linkage · Geometric approach · Overconstrained

1 Introduction

Several 6R paradoxical linkages were proposed by Bricard [1], Myard [2], Goldberg [3], Waldron [4], Baker [5], Wohlhart [6, 7], etc. A list of the known 6R linkages was appended in [8]. Until now, finding new overconstrained 6R linkages whose architectures are characterized by Euclidean metric constraints [9] is still an attractive topic. We qualified that category of overconstrained linkages as paradoxical. Our aim in this chapter is to derive movable 6R linkages in a visualized geometric way. Recently, three types of parallel 6R linkages with three couples of parallel joint-axes for all

C.-C. Lee (✉)
Department of Mold and Die Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung 80782, Taiwan, ROC
e-mail: cclee@cc.kuas.edu.tw

J. M. Hervé
Ecole Centrale des Arts et Manufactures, Châtenay-Malabry, France
e-mail: jacques.herve07@orange.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_7,

possible configurations were proposed [10–12] using dual quaternions and algebraic computation. The first type has two couples of two adjacent parallel joint-axes and one couple of opposite parallel joint-axes. Its architecture is addressed in [13] but its full geometric description is not provided. Our new geometric derivation will result in exactly the same parallel 6R linkage without using algebraic computation.

The planar four-revolute parallelogram generates circular translation between two opposite bars. Circular translation is a one-dimensional (1D) submanifold of a 2D group of planar translations. This fact was revealed in [14, 15] and was further applied to type synthesis of 4D motion generator in [16]. In the following, our investigation begins with the circular translational motion produced in a hinged parallelogram. A mechanical generator of two-degree-of-freedom (2-DoF) translation along a spherical surface is synthesized by combining several generators of circular translation. The 2-DoF spherical translation contains all the 1-DoF translational motions along curves drawn on the sphere. Special curves on a sphere can be planar circles. From that, we newly derive a family of movable 6R linkages in a purely geometric approach. The chapter is organized as follows. Section 2 makes a brief explanation of circular translation and its planar parallelogram generator. In Sect. 3, the planar-hinged parallelogram is employed to synthesize the 6R linkage with circular translation. Section 4 compares and verifies the already published 6R linkages with our derivation.

2 Circular Translation and the Planar-Hinged Parallelogram

Spatial translation is a motion type with no change of the orientation of the moving rigid body. In a 1-DoF curvilinear translation, each point of the translating body moves on a curve. The trajectories of any two points are congruent by the constant translation transforming one point into the other one. The four-revolute (4R) hinged parallelogram is the simplest way to mechanically generate translation without resorting to the prismatic pair. As shown in Fig. 1, a planar parallelogram generator A_0ABB_0 has four bars jointed by four revolute R pairs and the opposite bars have equal lengths, $\mathbf{A}_0\mathbf{A} = \mathbf{B}_0\mathbf{B}$ and $\mathbf{A}_0\mathbf{B}_0 = \mathbf{AB}$. The trajectory of point A is congruent with that of point B through the translation of vector $(\mathbf{AB})$, which transforms point A into point B.

A circular translation means a 1-DoF curvilinear translational motion with trajectories that are congruent arcs of circles. The point A has a circular trajectory that is congruent to the trajectory of B by the constant translation of vector $(\mathbf{AB})$ and the centers of the circles are A_0 and B_0 respectively. In fact, the planes of the circles for any point are parallel but are not necessarily coplanar. The bound vectors $(\mathbf{A}_0\mathbf{A})$ and $(\mathbf{B}_0\mathbf{B})$ are equipollent (the free vectors are equal) and

$$(\mathbf{A}_0\mathbf{A}) = (\mathbf{B}_0\mathbf{B}) = r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j}) \tag{1}$$

where r is a given radius and θ a variable angle. The angle θ is the canonic parameter of the 1-DoF translational motion along a circle. The point position A is obtained

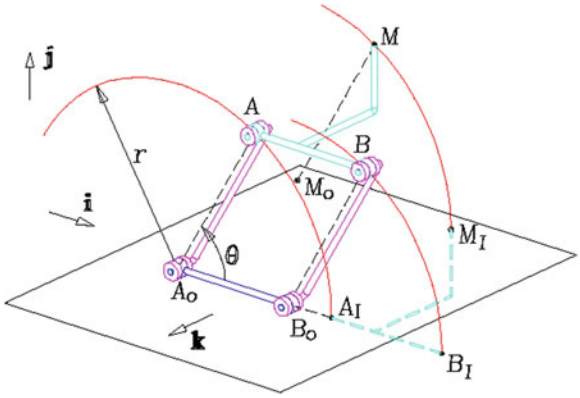


Fig. 1 Circular translation and the planar-hinged parallelogram

from a given initial position A_I. The point position A_I is chosen to satisfy $(\mathbf{A}_0\mathbf{A}_I) = r\mathbf{i}$. By translation, $(\mathbf{A}_0\mathbf{A}_I) = r\mathbf{i}$ becomes

$$(\mathbf{A}_0\mathbf{A}) = r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j}) \Rightarrow (\mathbf{A}_I\mathbf{A}) = r[(\cos\theta - 1)\mathbf{i} + \sin\theta\mathbf{j}] \tag{2}$$

In a similar way, $(\mathbf{B}_I\mathbf{B}) = r[(\cos\theta - 1)\mathbf{i} + \sin\theta\mathbf{j}]$.

Let M denote any point belonging to the translating body; then

$$(\mathbf{M}_I\mathbf{M}) = r[(\cos\theta - 1)\mathbf{i} + \sin\theta\mathbf{j}] = M - M_I \tag{3}$$

Any point at its initial position M_I is transformed into M, namely

$$\forall M_I, M_I \rightarrow M = M_I + r[(\cos\theta - 1)\mathbf{i} + \sin\theta\mathbf{j}] \tag{4}$$

The foregoing point transformation mathematically characterizes a circular translation. From that expression, one can establish that any point M moves on a circle of radius r. The center M_0 can be obtained from M_I using $(\mathbf{M}_I\mathbf{M}_0) = -r\mathbf{i}$ or from M using $(\mathbf{M}\mathbf{M}_0) = -r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j})$. The circular trajectory of M is the set of points $M(\theta) = M_0 + r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j})$.

A property originates from the fact that a circle is globally invariant by any rotation around its axis. An arc of the circle and an arc obtained by rotation around the axis are not congruent by translation. The conjugate by a rotation of a subset of circular translations [9] is also a subset of circular translations but the two subsets are not generally equal and the set intersection is the identity group.

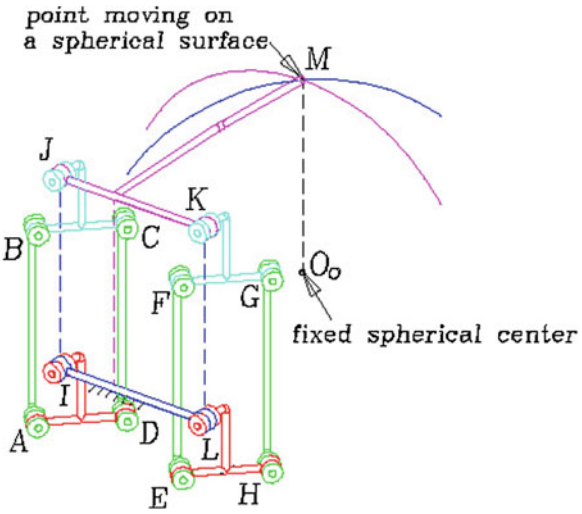


Fig. 2 2-DoF 12R mechanism with spherical translation

3 Mechanical Generator of Spherical Translation

Refer to Fig. 2. In a planar 4R parallelogram *IJKL*, the motion of the bar *JK* with respect to the bar *IL* assumed to be fixed is considered. The opposite bars *IJ* and *LK* are replaced respectively by two sided 4R planar parallelograms *ABCD* and *EFGH*, which are located on two parallel planes. Moreover, in the parallelogram *ABCD*, (**AB**) = (**DC**) = (**IJ**) = (**LK**) and in the parallelogram *EFGH*, (**EF**) = (**HG**) = (**LK**) = (**IJ**). The two bars *JK* and *IL* of the original *IJKL* 4R planar parallelogram have a relative motion of circular translation when the added *ABCD* and *EFGH* 4R parallelograms are locked at any of their feasible postures. The four R joint axes of the *IJKL* 4R planar parallelogram keep their parallelism when both side planar parallelograms move. The radius of the circular translation of *IJ* changes when motions in the side planar parallelograms modify the distances between the two parallel R axes at points *J* and *I* as well as between those at points *K* and *L*. As a result, the motion of the bar *JK* opposite to the fixed bar *IL* of the original 4R planar parallelogram is a one-dimensional set of circular translations and therefore is a 2-DoF translational motion. The point distance between the point *J* of the translating body and the fixed point *I* is constant, and consequently the point *J* moves on a sphere of center *I*. The motion of the body *JK* is a 2-Dof translation in which one point moves on a sphere; it is a translation along a sphere or spherical translation. All trajectories are congruent spheres. In the obtained multi-loop 12R linkage depicted in Fig. 2, any point *M* attached to the moving body *JK* moves on a sphere whose center O_0 is given by ($\mathbf{MO}_0$) = (**JI**).

If the point *M* is also constrained to move on a fixed plane, then it will move on the circular intersection of its spherical trajectory and the plane. This can be achieved

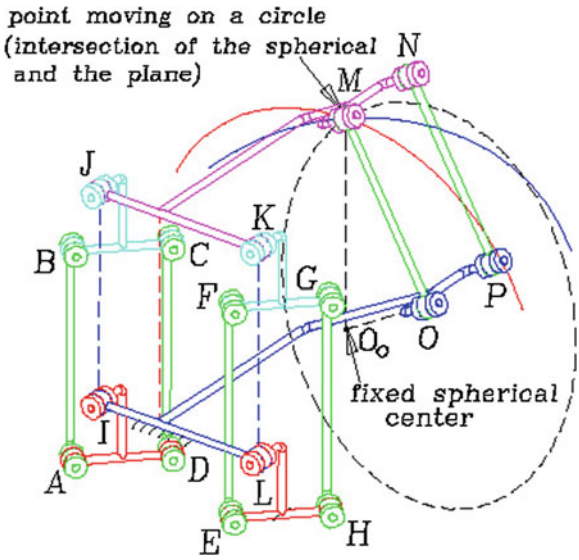


Fig. 3 A spatial 12R-4R mechanism with one-DoF circular translation

by adding one more 4R planar parallelogram *OMNP*. As shown in Fig. 3, the motion of rigid body *JK* will be a subset of the group of spatial translations having a point *M* with a circular trajectory whose fixed center is located at point *O*; it is a circular translation with a radius *OM*. The vector (**OM**) is the orthogonal projection of the vector ($\mathbf{O_0M}$) on the foregoing fixed plane. Hence, one obtains a 12R-4R linkage with 1-DoF circular translation, Fig. 3.

4 Single Loop 6R Linkages

Removing any three bars chosen from links AB, CD, EF, and GH and any one bar from links OM and NP in the multi-loop linkage of Fig. 3, 6R single-loop linkages can be derived. In fact, eight distinct forms of linkages are obtained but they all have the same architectural type and differ only on the particular values of their structural parameters. For brevity, only one form is depicted in Fig. 4. In this architecture, the non-parallel and generally non-intersecting RR links (1) and (3) are congruent and have the same length, $a_1 = a_3 = a$, which can be arbitrarily chosen. The length a_2(=b) of the RR link (2) with parallel axes is the radius of the spherical trajectories and can be arbitrarily chosen. The lengths a_2 and a_5 of link (2) and link (5) are not independent. The length a_5 of link (5) with two parallel axes is the radius of the circular trajectories and is the orthogonal projection of the sphere radius on the plane arbitrarily chosen for the circular translation. It obeys the relation: $a_5 = a_2 \cos\lambda = b \cos\lambda$, in which the $\lambda = \delta$ is the incline angle of projection plane. In addition, angles α, β, μ, and η are the twist angles between two R axes in links

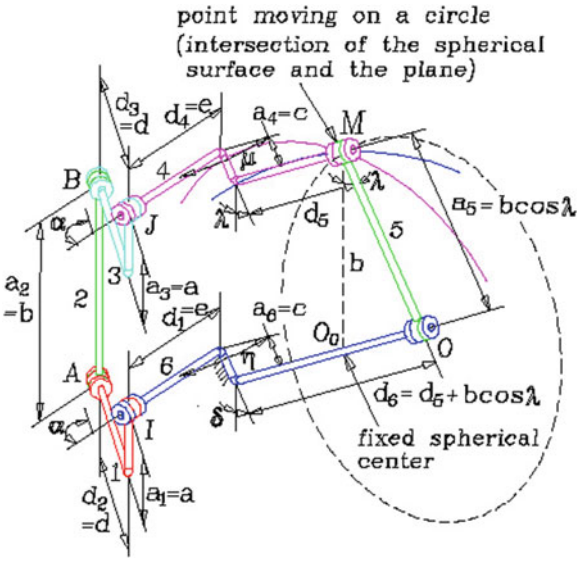


Fig. 4 A movable 6R linkage with one-DoF circular translation

(1), (3), (4), and (6), respectively. Finally, the parameterization of the 6R linkage leads to the following metric constraints:

$$\begin{gathered} a_1 = a_3 = a, a_2 = b, a_5 = b\cos\lambda, a_4 = a_6 = c, \\ \alpha = \beta, \lambda = \delta, \mu = \eta, d_1 = d_4 = e, d_5 - d_6 = b\sin\lambda \end{gathered} \tag{5}$$

in which a, b, c, e, α , λ , and μ are arbitrary parameters of linkage structure.

In the single loop 6R linkage of Fig. 4, the parallelograms shown in Fig. 3 virtually exist due to the special metric geometric constraints. Consequently, the angle of rotation of the body (3) with respect to (2) is equal to the angle of rotation of (1) with respect to (2). In the rotation (4)/(3) the angle is equal to the angle in the rotation (6)/(1). The rotations (4)/(5) and (6)/(5) have equal angles.

In a special case, the two revolute R axes in links (1) and (3) intersect. Its architecture is depicted in Fig. 5a. In a more special case, the revolute R axis between links 4 and 5 passes through the intersection of the R axes between links 2 and 3, and links 3 and 4, respectively. Then the R axis between links 5 and 6 passes through the intersection of the axes between links 6 and 1, and links 1 and 2. The closed loop type is (RRR)–(RRR) with two sets of three intersecting R axes as described in Fig. 5b. It is a special bi-spherical (RRR)–(RRR) linkage with exceptional mobility and the body (5) rotates with respect to the body (2) around the axis which connects the two centers of the (RRR) spherical sub-chains.

When the constant angle α in the links (1) and (3) is a right angle, the bispherical linkage includes two universal (RR) = U joints, and the series of two U joints realizes a constant velocity transmission between the two parallel R axes of link (5).

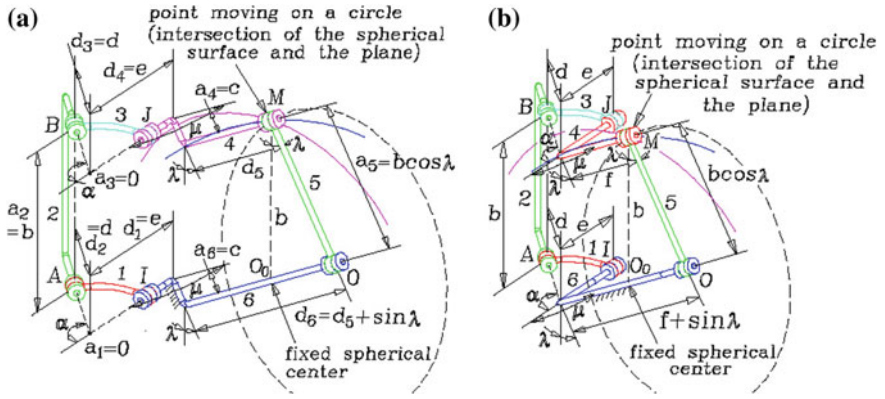


Fig. 5 A special movable 6R linkage with zero common perpendiculars. **a** $a_1 = a_3 = 0$, **b** a spherical bi-spherical space

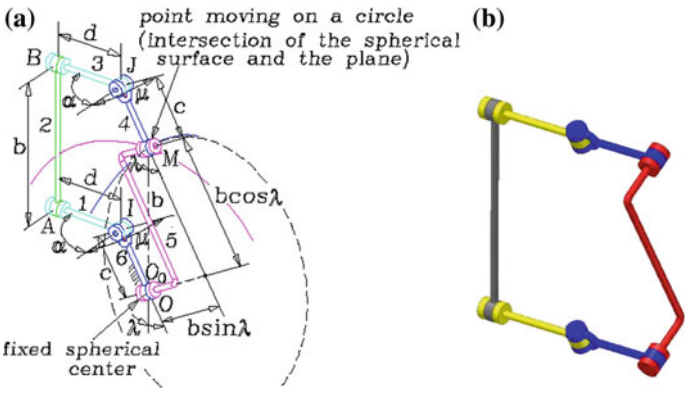


Fig. 6 Gfrerrer 6R linkage. **a** Gfrerrer's 6R, **b** Gfrerrer's 6R configuration

5 Already Published 6R Linkages

Li and Schicho [10], Li and Schicho [11] and Li [12] have introduced angle-symmetric 6R linkages using the representation of rigid-body displacement by dual quaternion. The synthesis is done by algebraic computation. The synthesized linkages are not fully characterized by their geometric descriptions. One type is proposed by Gfrerrer in [12] and termed as parallel 6R linkage. It is a special case of 6R with circular translation. When twist angles between joint axes in links 1 and 3 and twist angles between joint axes in links 4 and 6 are right angles (i.e. $\alpha = \beta = 90°$, $\mu = \eta = 90°$) and the common perpendiculars between joint axes in links 1 and 3 are zeros ($a_1 = a_3 = 0$), one obtains the special architecture of Fig. 6a. The configuration of this linkage is further verified in Fig. 6b.

6 Conclusions

In the chapter, a family of movable 6R linkages with circular translation is synthesized in a visualized geometric way. The synthesis method is based on the fact that the intersection of a spherical translational motion and a planar motion is circular translational motion. The 2-DoF spherical translation is generated by means of a specific arrangement of three parallelograms. The planar hinged-parallelogram produces 1-DoF circular translation. Although parallel 6R linkages were already derived by dual quaternion method, our work is expected to provide a preliminary insight into a possible geometric derivation of other potential parallel linkages that have three couples of opposite parallel joint-axes. It is also hoped that, extending the idea of this work, a new category of 6H linkages with circular translation property could be achieved in the near future.

Acknowledgments The authors are very thankful to the National Science Council for supporting this research under grants NSC 101-2221-E-151-017 and NSC 102-2221-E-151-012.

References

1. Bricard, R.: Leçons de Cinématique, vol. 2. Gauthier-Villars, Paris (1927)
2. Myard, F.E.: Contribution à la géométrie des systémes articulés. Bulletin de la Société Mathématique de France **59**, 183–210 (1931)
3. Goldberg, M.: New five-bar and six-bar linkages in three dimensions. ASME Trans. **65**, 649–661 (1943)
4. Waldron, K.J.: Hybrid overconstrained linkages. J. Mech. **3**(2), 73–78 (1968)
5. Yu, H.-C., Baker, J.E.: On the generation of new linkages from Bennett loops. Mech. Mach. Theory **16**(5), 473–485 (1981)
6. Wohlhart, K.: A new 6R space mechanism. In: Proceeding of 7th World Congress on Mechanism and Machine Theory, No. 5, pp. 193–198. Seville (1987)
7. Wohlhart, K.: Merging two general Goldberg 5R linkages to obtain a new 6R space mechanism. Mech. Mach. Theory **27**(7), 659–668 (1991)
8. Baker, J.E.: Displacement-closure equations of the unspecialised double-Hooke's-joint linkage. Mech. Mach. Theory **10**, 1127–1144 (2002)
9. Lee, C.-C., Hervé, J.M.: Synthesize new 5-bar paradoxical chains via the elliptic cylinder. Mech. Mach. Theory **46**(6), 784–793 (2011)
10. Li, Z., Schicho, J.: There types of parallel 6R linkages. In: Thomas, F, Perez Gracia A (eds.) Computational Kinematics: Proceedings of the 6th International Workshop on Computational Kinematics (CK2013), pp. 111–119. Springer (2013)
11. Li, Zijia, Schicho, J.: Classification of angle-symmetric 6R linkages. Mech. Mach. Theory **70**, 372–379 (2013)
12. Li, Z.: PowerPoint presentation for three types of parallel 6R linkage. Doctral program, Johan Radon institute for Computational and Applied mathematics, Linz, Austria (DK9)
13. Baker, J.E.: Overconstrained six-bar with parallel adjacent joint-axes. Mech. Mach. Theory **38**(2), 103–117 (2003)
14. Hervé, J. M., Sparacino, F.: Structural synthesis of parallel robots generating spatial translation. In: Proceeding of 5th IEEE International Conference on Advanced Robotics, Pisa, Italy, pp. 808–813 (1991)

15. Wohlhart, K.: Displacement analysis of the general spatial parallelogram manipulator. In: Proceedings of the 3rd International Work shop on Advances in Robot Kinematics, pp. 104–111. Ferrara, Italy, 7–9 Sept 1992
16. Lee, C.-C., Hervé, J.M.: Type synthesis of primitive Schoenflies-motion generators. Mech. Mach. Theory **44**(10), 1980–1997 (2009)

Function Synthesis of the Planar 5R Mechanism Using Least Squares Approximation

Gökhan Kiper, Barış Bağdadioğlu and Tunç Bilgincan

Abstract In this chapter, the problem of function generation synthesis of planar 5R mechanism is studied using the least squares approximation method with equal spacing of the design points. The study represents a case study for analytical function generation of multi-degrees-of-freedom systems. The planar 5R mechanism is designed with a fixed input joint and a moving input joint adjacent to the first input, whereas the remaining fixed joint is the output joint. The input/output relationship of the mechanism is expreseed as an objective function in polynomial form with four unknown construction parameters. The objective function involves nonlinearities, however the problem is linearized using Lagrange variables. The linear system is solved and finally the construction parameters of the mechanism are determined. A numerical example is presented as a case study.

Keywords Function generation · Planar 5R mechanism · Least squares approximation · Equal spacing

1 Introduction

One of the research areas in Rasim Alizade Mechatronics Laboratory in IzTech addresses analytical kinematic synthesis of multi degrees-of-freedom (dof) systems. Although analytical kinematic synthesis methods for single-dof mechanisms are widely studied [1, 2], usually numerical optimization methods are utilized for

G. Kiper (✉) · B. Bağdadioğlu · T. Bilgincan
İzmir Institue of Technology, 35430 İzmir, Turkey
e-mail: gokhankiper@iyte.edu.tr

B. Bağdadioğlu
e-mail: barisbagdadioglu@iyte.edu.tr

T. Bilgincan
e-mail: tuncbilgincan@iyte.edu.tr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_8,

dimensioning multi-dof mechanisms (ex. see Alizade [3]). There are a few studies on analytical synthesis methods for multi-dof systems [4, 5]. Recently we worked on function synthesis of a planar 5R mechanism [6] and a spherical 5R mechanism [7]. The present study differs from [6] in selection of one of the inputs, and also a different synthesis method is utilized.

Three widely used approximation methods for kinematic synthesis are the interpolation, least squares and Chebyshev approximation methods [8]. In this study we employ the least squares method [8–10]. In any of these methods, first an objective function is defined using the input/output (I/O) relationship of the mechanism and then an approximation in polynomial form is sought for the objective function. The approximation is performed for a certain number of design points on the domain of the inputs. For the interpolation approximation, these design points are precision points at which the mechanism exactly generates the required function value, while for the least squares and Chebyshev approximations the aim is to minimize the nonzero errors at the design points. The approximation error depends on the selection of design points and optimizing the selection is a big issue [11]. In case of single-dof mechanisms the domain for the design points is just a line segment, while in general for an n-dof mechanism the domain will be an n-dimensional compact space. Specifically, for a 2-dof mechanism the domain is typically rectangular.

In Kiper et al. [6, 7] we employed Chebyshev approximation. In Kiper et al. [6] we had five design points, four of which we located on the sides of the rectangular domain. In Kiper and Bilgincan [7] we tried a different type of spacing for the design points, which we called regional spacing. In regional spacing the design points are located in distinct regions in the domain and since the Chebyshev approximation method involves iterations, the design points are relocated in each iteration step, but they are forced to remain in their respective regions. In least squares approximation there is no iteration. As the name implies, the aim is to minimize the sum of the squares of the errors at the design points. In this study we make use of equal spacing for the design points. For single-dof mechanisms generally Chebyshev spacing gives superior results [8, 12], however the comparison of the methods for 2-dof mechanisms is yet to be done. Although we obtained quite good results with Chebyshev approximation [6, 7], we expect that least squares approximation may yield better results, at least for some functions and some mechanisms.

2 Formulation

In this study, the input variables θ and ϕ of the planar 5R mechanism are associated with one of the fixed joints and the adjacent floating joint. The output variable ψ is associated with the remaining fixed joint. In practice, an extra parallelogram loop can be employed in order to actuate the mechanism at fixed joints (Fig. 1). Since the scale of the mechanism does not affect the I/O relationship, without loss of generality we may assume that the fixed link length is 1. The construction parameters are a, b, d and e shown in Fig. 1.

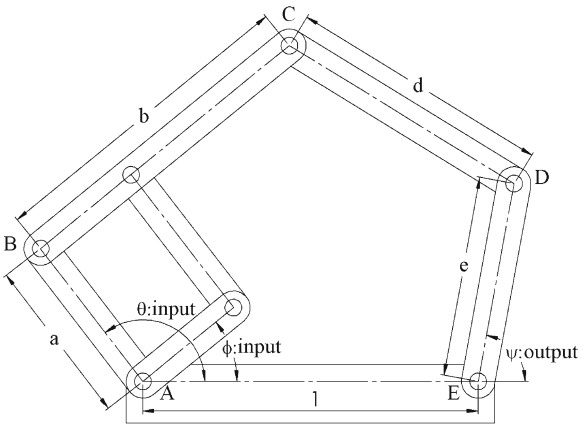


Fig. 1 The construction parameters and joint variables of the 5R mechanism

The I/O relationship for the mechanism is obtained as follows:

$$\begin{aligned} \left|\overrightarrow{CD}\right| &= \left|\overrightarrow{AE} + \overrightarrow{ED} - \overrightarrow{AB} - \overrightarrow{BC}\right| \\ \Rightarrow (aC\theta + bC\phi - 1 - eC\psi)^2 + (aS\theta + bS\phi - eS\psi)^2 &= d^2 \end{aligned} \tag{1}$$

where C and S represent the cosine and sine functions, respectively. Rearranging Eq. (1) in polynomial form:

$$\sum_{j=1}^{6} P_j f_j(\mathbf{x}) - F(\mathbf{x}) = 0 \tag{2}$$

where $\mathbf{x} = \{\theta, \phi, \psi\}$, whereas $\left\{P_j\right\}_1^6$,$\left\{f_j(\mathbf{x})\right\}_1^6$ and $F(\mathbf{x})$ are defined as:

$$P_1 = \frac{-1 - a^2 - b^2 + d^2 - e^2}{2e}, \quad P_2 = a, \quad P_3 = b, \quad P_4 = \frac{a}{e},$$

$$P_5 = P_3 P_4 = \lambda_1 = \frac{ab}{e}, \quad P_6 = \frac{P_5}{P_2} = \lambda_2 = \frac{b}{e}, \tag{3}$$

$$f_1(\mathbf{x}) = 1,\ f_2(\mathbf{x}) = \cos(\theta - \psi),\ f_3(\mathbf{x}) = \cos(\phi - \psi),\ f_4(\mathbf{x}) = \cos\theta,$$
$$f_5(\mathbf{x}) = -\cos(\theta - \phi),\ f_6(\mathbf{x}) = \cos\phi \text{ and } F(\mathbf{x}) = \cos\psi$$

There are four construction parameters, but six P_j's. Therefore P_5 and P_6 are defined in terms of the other P_j's and two Lagrange parameters $P_5 = \lambda_1$ and $P_6 = \lambda_2$ are introduced as two more construction parameters. In order to linearize the system, let $P_j = \ell_j + m_j\lambda_1 + n_j\lambda_2$ for $j = 1, 2, 3, 4$. Eq. (2) becomes

$$\sum_{j=1}^{4}(\ell_j + m_j\lambda_1 + n_j\lambda_2) f_j(\mathbf{x}) + \lambda_1 f_5(\mathbf{x}) + \lambda_2 f_6(\mathbf{x}) - F(\mathbf{x}) = 0 \tag{4}$$

Equation (4) should be satisfied for all $\mathbf{x}$, hence the coefficients of λ_1, λ_2 and the rest can be dissected as follows:

$$\sum_{j=1}^{4}\ell_j f_j(\mathbf{x}) - F(\mathbf{x}) = 0 \tag{5}$$

$$\sum_{j=1}^{4} m_j f_j(\mathbf{x}) + f_5(\mathbf{x}) = 0 \tag{6}$$

$$\sum_{j=1}^{4} n_j f_j(\mathbf{x}) + f_6(\mathbf{x}) = 0 \tag{7}$$

In least squares approximation the number of design points, N, is necessarily greater than the number of construction parameters and we aim to minimize the sum of the squares of the errors at these design points. At each design point $\mathbf{x}_i$ for $i = 1, ..., N$, the sum of the squares of the errors corresponding to Eqs. (5–7) are defined as

$$S_\ell = \sum_{i=1}^{N}\left[\sum_{j=1}^{4}\ell_j f_{ji} - F_i\right]^2 \tag{8}$$

$$S_m = \sum_{i=1}^{N}\left[\sum_{j=1}^{4} m_j f_{ji} + f_{5i}\right]^2 \tag{9}$$

$$S_n = \sum_{i=1}^{N}\left[\sum_{j=1}^{4} n_j f_{ji} + f_{6i}\right]^2 \tag{10}$$

where $f_{ji} = f_j(\mathbf{x}_i)$, $f_{5i} = f_5(\mathbf{x}_i)$, $f_{6i} = f_6(\mathbf{x}_i)$ and $F_i = F(\mathbf{x}_i)$. In order to find the minimum of the sum of the squares, we set the derivatives of Eqs. (8–10) with respect to ℓ_j, m_j, n_j to zero for $j = 1, 2, 3, 4$ to obtain

$$\frac{1}{2}\frac{\partial S_\ell}{\partial \ell_j} = \sum_{i=1}^{N}[f_{1i}\ell_1 + f_{2i}\ell_2 + f_{3i}\ell_3 + f_{4i}\ell_4 - F_i] f_{ji} = 0 \tag{11}$$

$$\frac{1}{2}\frac{\partial S_m}{\partial m_j} = \sum_{i=1}^{N} [f_{1i}m_1 + f_{2i}m_2 + f_{3i}m_3 + f_{4i}m_4 + f_{5i}]\, f_{ji} = 0 \tag{12}$$

$$\frac{1}{2}\frac{\partial S_n}{\partial n_j} = \sum_{i=1}^{N} [f_{1i}n_1 + f_{2i}n_2 + f_{3i}n_3 + f_{4i}n_4 + f_{6i}]\, f_{ji} = 0 \tag{13}$$

For $j = 1, 2, 3, 4$ Eq. (11) constitute a linear set of 4 equations in unknowns $\ell_1, \ell_2, \ell_3, \ell_4$ and similarly Eqs. (12) and (13) are respectively linear in m_1, m_2, m_3, m_4 and n_1, n_2, n_3, n_4. Writing Eqs. (11–13) in matrix form:

$$[A_{jk}][\ell_j] = [b_j] \tag{14}$$

$$[A_{jk}][m_j] = [c_j] \tag{15}$$

$$[A_{jk}][n_j] = [d_j] \tag{16}$$

where $[A_{jk}]$ is the 4×4 coefficient matrix with $A_{jk} = \sum_{i=1}^{N} f_{ki} f_{ji}$ for $j, k = 1, 2, 3, 4$, $[\ell_j] = [\ell_1\ \ell_2\ \ell_3\ \ell_4]^T$, $[m_j] = [m_1\ m_2\ m_3\ m_4]^T$, $[n_j] = [n_1\ n_2\ n_3\ n_4]^T$ and $[b_j]$, $[c_j]$ and $[d_j]$ are 4×1 matrices with $b_j = \sum_{i=1}^{N} F_i f_{ji}$, $c_j = -\sum_{i=1}^{N} f_{5i} f_{ji}$, $d_j = -\sum_{i=1}^{N} f_{6i} f_{ji}$ for $j = 1, 2, 3, 4$.

ℓ_j, m_j, n_j are uniquely solved from Eqs. (14–16). Once ℓ_j, m_j, n_j are determined, λ_1 and λ_2 are solved as follows:

$$\begin{aligned}\lambda_1 = P_3P_4 = (\ell_3 + m_3\lambda_1 + n_3\lambda_2)(\ell_4 + m_4\lambda_1 + n_4\lambda_2) \Rightarrow m_3m_4\lambda_1^2 + n_3n_4\lambda_2^2 \\ +(m_3n_4 + m_4n_3)\,\lambda_1\lambda_2 + (\ell_3m_4 + \ell_4m_3 - 1)\lambda_1 + (\ell_3n_4 + \ell_4n_3)\lambda_2 + \ell_3\ell_4 = 0\end{aligned} \tag{17}$$

$$\lambda_2 = \frac{P_5}{P_2} = \frac{\lambda_1}{(\ell_2 + m_2\lambda_1 + n_2\lambda_2)} \Rightarrow n_2\lambda_2^2 + m_2\lambda_1\lambda_2 - \lambda_1 + \ell_2\lambda_2 = 0 \tag{18}$$

We can solve for λ_1 from Eq. (18):

$$\lambda_1 = \frac{n_2\lambda_2^2 + \ell_2\lambda_2}{(1 - m_2\lambda_2)} \tag{19}$$

Substituting Eq. (19) in Eq. (17):

$$\begin{aligned}m_3m_4(n_2\lambda_2^2 + \ell_2\lambda_2)^2 + n_3n_4\lambda_2^2(1 - m_2\lambda_2)^2 + (\ell_3n_4 + \ell_4n_3)\lambda_2(1 - m_2\lambda_2)^2 \\ +(m_3n_4 + m_4n_3)(n_2\lambda_2^2 + \ell_2\lambda_2)(1 - m_2\lambda_2)\lambda_2 + \ell_3\ell_4(1 - m_2\lambda_2)^2 \\ +(\ell_3m_4 + \ell_4m_3 - 1)(n_2\lambda_2^2 + \ell_2\lambda_2)(1 - m_2\lambda_2) = 0\end{aligned} \tag{20}$$

Equation (20) is a 4th order polynomial equation in λ_2 and can be solved analytically. There may be 4, 2 or no real solutions for λ_2. If exists, once one

of the solutions for λ_2 is selected, λ_1 is determined uniquely from Eq. (19). $P_j = \ell_j + m_j\lambda_1 + n_j\lambda_2$ for $j = 1, 2, 3, 4$ are determined and the construction parameters are solved uniquely from Eq. (3) as

$$a = P_2, b = P_3, e = \frac{a}{P_4}, d = \sqrt{1 + a^2 + b^2 + e^2 + 2eP_1}$$

3 The Function Synthesis Problem

Let the function to be generated be $z = f(x, y)$ for $x_{\min} \leq x \leq x_{\max}$ and $y_{\min} \leq y \leq y_{\max}$. The independent variables x and y should be related to the mechanism inputs θ and ϕ and the dependent variable z should be related to the mechanism output ψ. θ, ϕ and ψ are in ranges $\theta_{\min} \leq \theta \leq \theta_{\max}, \phi_{\min} \leq \phi \leq \phi_{\max}$, $\psi_{\min} \leq \psi \leq \psi_{\max}$ and the limits can be arbitrarily chosen. We shall linearly relate x to input θ, y to input ϕ and z to output ψ as [1]

$$\begin{aligned} \frac{x - x_{\min}}{x_{\max} - x_{\min}} &= \frac{\theta - \theta_{\min}}{\theta_{\max} - \theta_{\min}} \\ \frac{y - y_{\min}}{y_{\max} - y_{\min}} &= \frac{\phi - \phi_{\min}}{\phi_{\max} - \phi_{\min}} \\ \frac{z - z_{\min}}{z_{\max} - z_{\min}} &= \frac{\psi - \psi_{\min}}{\psi_{\max} - \psi_{\min}} \end{aligned} \tag{21}$$

Then input values θ and ϕ and the desired output values ψ of the mechanism can be solved from Eq. (21) as follows:

$$\begin{aligned} \theta &= \frac{x - x_{\min}}{x_{\max} - x_{\min}}(\theta_{\max} - \theta_{\min}) + \theta_{\min} \\ \phi &= \frac{y - y_{\min}}{y_{\max} - y_{\min}}(\phi_{\max} - \phi_{\min}) + \phi_{\min} \\ \psi &= \frac{z - z_{\min}}{z_{\max} - z_{\min}}(\psi_{\max} - \psi_{\min}) + \psi_{\min} \end{aligned} \tag{22}$$

and conversely

$$\begin{aligned} x &= \frac{\theta - \theta_{\min}}{\theta_{\max} - \theta_{\min}}(x_{\max} - x_{\min}) + x_{\min} \\ y &= \frac{\phi - \phi_{\min}}{\phi_{\max} - \phi_{\min}}(y_{\max} - y_{\min}) + y_{\min} \end{aligned} \tag{23}$$

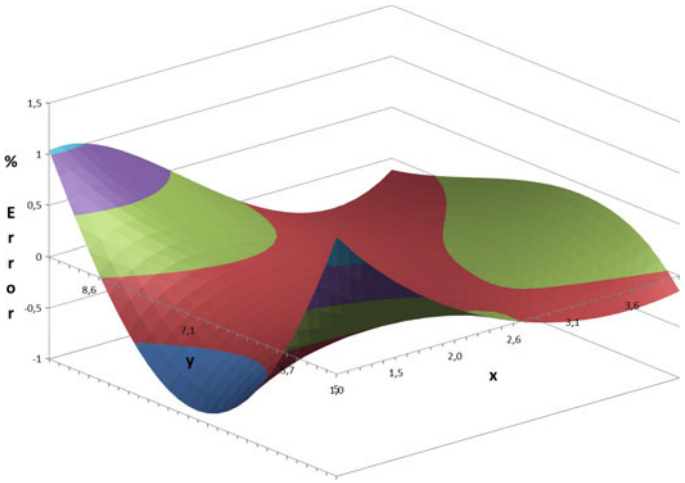


Fig. 2 The percentage error variation over the domain of x and y

$$z = \frac{\psi - \psi_{\min}}{\psi_{\max} - \psi_{\min}}(z_{\max} - z_{\min}) + z_{\min}$$

We make use of Eq. (22) when determining the design points $\{\theta_i\}_1^N$, $\{\phi_i\}_1^N$ and $\{\psi_i\}_1^N$ in terms of $\{x_i\}_1^N$, $\{y_i\}_1^N$ and $\{z_i\}_1^N = \{f(x_i, y_i)\}_1^N$. We select the design points (x_i, y_i) with equal spacing on the rectangular domain given by $x_{\min} \le x \le x_{\max}$ and $y_{\min} \le y \le y_{\max}$, i.e. $x_1 = x_{\min}, y_1 = y_{\min}$, $x_i = x_{\min} + \frac{i-1}{N-1}(x_{\max} - x_{\min})$ and $y_i = y_{\min} + \frac{i-1}{N-1}(y_{\max} - y_{\min})$ for $i = 2, ..., N$.

We make use of Eq. (23) after the synthesis is performed, to check the error in between the desired $z = f(x, y)$ and z generated by the mechanism. At this step, one shall determine the output values of the mechanism for several given input values by solving the I/O relationship.

4 Case Study

The formulations in the previous sections were implemented in MS Excel® and a case study was worked out for a function $z = x^{1.1}y^{1.4}$ for $5 \le x \le 9$ and $1 \le y \le 4$. Limits for the mechanism input and output angles are selected as $75° \ge \theta \ge 30°$, $80° \le \phi \le 130°$ and $120° \le \psi \le 170°$. Actually several different limit values were employed, but the final selection is done until we obtain a small maximum error and good link length ratios. The design points are selected with equal spacing of 30 intervals for both x and y. That is, there are totally 900 design points.

As a result of computations, two real solutions for λ_2 are found. For one of the solutions the link lentgh ratios were not appropriate, whereas for the other

solution a nice 5R mechanism is obtained. The maximum percentage error $100\times \left|\frac{\psi_{desired}-\psi_{computed}}{\psi_{desired}}\right|$ is found as 1,33 %. The variation of the percentage error over the domain of x and y is illustrated in Fig. 2. The construction parameters are calculated as a = 2.382, b = 1.636, d = 2.671, e = 1.577.

5 Conclusions

In this chapter, the function generation problem for a planar 5R mechanism is addressed using analytical solution for given set of design point. The problem is formulated starting with expressing the I/O relationship in polynomial form. The equations are linearized by introducing Lagrange variables. The linear set of equations and the Lagrange variables are solved analytically. A computational example is presented and the error variation is given.

As future studies we plan to work out the same problem using Chebyshev approximation and compare the results. Also we plan to apply these analytical and semi-analytical approximation methods to other multi-dof mechanisms.

References

1. Erdman, A., Sandor, G.N.: Mechanism Design: Analysis and Synthesis. Prentice Hall, Englewood Cliffs (1984)
2. McCarthy, J.M., Soh, G.S.: Geometric Desgin of Linkages, 2nd edn. Springer, New York (2011)
3. Alizade, R., Rao, A.V.M., Sandor, G.N.: Optimum synthesis of two-degree-of-freedom planar and spatial function generating mechanisms using the penalty function approach. J. Eng. Ind. **97**(2), 629–634 (1975)
4. Svoboda, A.: Computing Mechanisms and Linkages, 2nd edn. Dover, New York (1965)
5. Kim, H.S., Tsai, L.-W.: Kinematic synthesis of a spatial 3-RPS parallel manipulator. J. Mech. Des. **125**, 92–97 (2003)
6. Kiper, G., Bilgincan, T., Dede, Mİ.C.: Function generation synthesis of planar 5R mechanism. Probl. Mech. **2**(51), 28–31 (2013)
7. Kiper, G., Bilgincan, T.: Function generation synthesis of spherical 5R mechanism. Mech. Mach. Theory (2014)
8. Alizade, R., Gezgin, E.: Synthesis of function generating spherical four bar mechanism for the six independent parameters. Mech. Mach. Theory **46**, 1316–1326 (2011)
9. Levitskii, N.I., Sarkisian, E.Y.: On the special properties of Lagrange's multipliers in the least-square synthesis of mechanisms. J. Mech. **3**(1), 3–10 (1968)
10. Alizade, R., Can, F.C., Kilit, Ö.: Least square approximate motion generation synthesis of spherical linkages by using Chebyshev and equal spacing. Mech. Mach. Theory **61**, 123–135 (2013)
11. Suixian, Y., Hong, Y., Tian, G.Y.: Optimal selection of precision points for function synthesis of spherical 4R linkage. Proc. Inst. Mech. Eng. Part C: J. Mech. Eng. Sci. **223**, 2183–2189 (2009)
12. Alizade, R., Kiper, G., Bağdadioğlu, B., Dede, M.İ.C.: Function synthesis of Bennett 6R mechanisms using Chebyshev approximation. Mech. Mach. Theory (2014)

Some Remarks on the *RRR* Linkage

J. M. Selig

Abstract The variety of rigid-body displacements of the final link of a $3R$ kinematic chain are investigated. In most cases the variety generated is a Segre manifold; the Cartesian product of three projective lines. The homology of this variety as a subvariety of the Study quadric is found and simple applications to some enumerative problems in kinematics are given. The conditions for the variety to fail to be a Segre variety are investigated in full and the case where the linkage forms the first three joints of a Bennett mechanism is examined.

Keywords $3R$ linkages · Segre variety · Homology

1 Introduction

Husty et al. [2], recognised that the possible displacements achievable by a $3R$ linkage could be viewed as a Segre variety in the Study quadric. In this work several results on the geometry of this Segre variety and its significance to kinematics are collected together. Some of these results have been published elsewhere, but to the author's knowledge, most have not.

2 The Segre Variety $\mathbb{P}^1 \times \mathbb{P}^1 \times \mathbb{P}^1$

Consider a point in $\mathbb{P}^1 \times \mathbb{P}^1 \times \mathbb{P}^1$ to be given by $(c_1, s_1) \times (c_2, s_2) \times (c_3, s_3)$ where c_i and s_i are the homogeneous coordinates of the ith projective line of the product; $i = 1, 2, 3$. For the $3R$ linkage, these will be the cosine and sine of the joint half-angles, see below. The Segre embedding is the map,

J. M. Selig (✉)
London South Bank University, London SE1 0AA, UK
e-mail: seligjm@lsbu.ac.uk

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_9,

$$\mathbb{P}^1 \times \mathbb{P}^1 \times \mathbb{P}^1 \longrightarrow \mathbb{P}^7$$

given explicitly in terms of homogeneous coordinates by,

$$\begin{aligned}(c_1, s_1) \times (c_2, s_2) \times (c_3, s_3) \longmapsto \\ (c_1c_2c_3, s_1c_2c_3, c_1s_2c_3, c_1c_2s_3, s_1s_2s_3, -c_1s_2s_3, s_1c_2s_3, -s_1s_2c_3). \end{aligned} \quad (1)$$

This particular ordering and the inclusion of the negative signs are arbitrary. Notice that each degree three monomial, separately homogeneous in the coordinates of each $\mathbb{P}^1$, appears in the image. The image is the Segre variety which forms the central subject of this work.

Returning to the $3R$ linkage, the possible displacements of the end-effector can be found by multiplying dual quaternions representing displacement about the 3 R-joints,

$$\begin{aligned}(c_1 + s_1\ell_1)(c_2 + s_2\ell_2)(c_3 + s_1\ell_3) = {} & s_1c_2c_3\ell_1 + c_1s_2c_3\ell_2 + c_1c_2c_3 + c_1c_2s_3\ell_3 \\ & + c_1s_2s_3\ell_2\ell_3 + s_1c_2s_3\ell_1\ell_3 + s_1s_2c_3\ell_1\ell_2 \\ & + s_1s_2s_3\ell_1\ell_2\ell_3,\end{aligned}$$

where ℓ_i is the dual quaternion representing the line of the ith joint axis in the home position of the linkage.

Taking the c_i and s_j as parameters, this parametrises a rational variety in $\mathbb{P}^7$. This variety is, in general, projectively equivalent to the Segre variety defined above. To see this consider the parameterisation given above as a matrix equation, $\mathbf{g} = M\mathbf{w}$. Here the vector $\mathbf{g} = (a_0, a_1, a_2, a_3, c_0, c_1, c_2, c_3)^T$, is the vector of dual quaternion components and

$$\mathbf{w} = (c_1c_2c_3, s_1c_2c_3, c_1s_2c_3, c_1c_2s_3, s_1s_2s_3, -c_1s_2s_3, s_1c_2s_3, -s_1s_2c_3)^T.$$

The matrix M has columns given by the components of the dual quaternions $1, \ell_1, \ell_2, \ldots, -\ell_1\ell_2$. To be definite, take the lines determined by the joints of an arbitrary $3R$ linkage which can always be opened to give the form shown in Fig. 1. Placing the origin of coordinates as shown in the figure, at the foot of the common perpendicular between the first and second joint axes and with the axes aligned as shown, the matrix M can be written explicitly, the (transposed) columns will be,

$$\begin{aligned}1^T &= \big(1, 0, 0, 0, 0, 0, 0, 0\big) \\ \ell_1^T &= \big(0, 0, \sin\alpha_1, \cos\alpha_1, 0, 0d_1\cos\alpha_1, -d_1\sin\alpha_1\big) \\ \ell_2^T &= \big(0, 0, 0, 1, 0, 0, 0, 0\big) \\ \ell_3^T &= \big(0, 0, -\sin\alpha_2, \cos\alpha_2, 0, h\sin\alpha_2, -d_2\cos\alpha_2, -d_2\sin\alpha_2\big) \\ (\ell_1\ell_2\ell_3)^T &= \big(0, 0, \sin(\alpha_1 - \alpha_1), -\cos(\alpha_2 - \alpha_1\big), -h\sin\alpha_1\sin\alpha_2, -h\cos\alpha_1\sin\alpha_2, \\ &\qquad (d_2 - d_1)\cos(\alpha_2 - \alpha_1), (d_2 - d_1)\sin(\alpha_2 - \alpha_1)\big)\end{aligned}$$

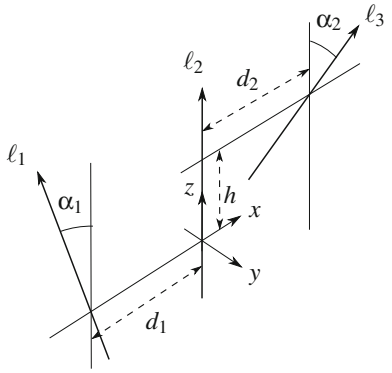


Fig. 1 The standard position for a general $3R$ linkage

$$
\begin{aligned}
-(\ell_2\ell_3)^T &= \big(\cos\alpha_2, -\sin\alpha_2, 0, 0, -d_2\sin\alpha_2, -d_2\cos\alpha_2, -h\sin\alpha_2, 0\big)\\
(\ell_1\ell_3)^T &= \big(-\cos(\alpha_1+\alpha_2), \sin(\alpha_1+\alpha_2), 0, 0, (d_1+d_2)\sin(\alpha_1+\alpha_2),\\
&\qquad (d_1+d_2)\cos(\alpha_1+\alpha_2), h\cos\alpha_1\sin\alpha_2,\ -h\sin\alpha_1\sin\alpha_2\big)\\
-(\ell_1\ell_2)^T &= \big(\cos\alpha_1, -\sin\alpha_1, 0, 0, -d_1\sin\alpha_1, -d_1\cos\alpha_1, 0, 0\big)
\end{aligned}
$$

When the matrix M is non-singular, it can be viewed as a projective change of coordinates in $\mathbb{P}^7$. The determinant of M can be computed and written as,

$$
\begin{aligned}
\det(M) = &-h^4\sin^4\alpha_1\sin^4\alpha_2 - 2h^2\sin^2\alpha_1\sin^2\alpha_2(d_1^2\sin^2\alpha_2 + d_2^2\sin^2\alpha_1)\\
&-(d_1^2\sin^2\alpha_2 - d_2^2\sin^2\alpha_1)^2. \qquad (2)
\end{aligned}
$$

This is clearly non-singular in general. Condition for this expression to vanish will be investigated below.

2.1 Nine Quadrics

Classically the implicit equations for a Segre variety comprise a number of quadrics, each a simple 2×2 determinant. Take $(X_0 : X_1 : X_2 : X_3 : Y_0 : Y_1 : Y_2 : Y_3)$ as homogeneous coordinates for $\mathbb{P}^7$. In these coordinates the Segre map will be written,

$$X_0 = c_1c_2c_3, \quad X_1 = s_1c_2c_3, \quad X_2 = c_1s_2c_3, \quad X_3 = c_1c_2s_3$$

and

$$Y_0 = s_1s_2s_3, \quad Y_1 = -c_1s_2s_3, \quad Y_2 = s_1c_2s_3, \quad Y_3 = -s_1s_2c_3.$$

Consider these coordinates as the label for the vertices of a cube, see Fig. 2. The vertices are labelled so that vertices are joined by an edge if and only if the corresponding monomials differ in only one position. Each of the six faces gives a 2×2

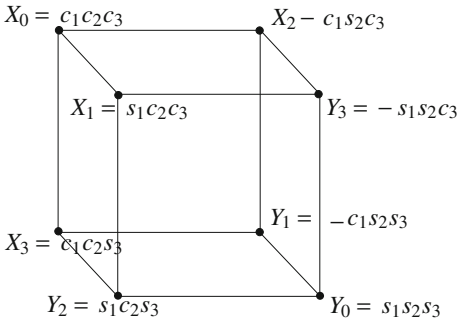


Fig. 2 The monomials defining the Segre embedding

determinant,

$$\begin{aligned} Q_1 &: X_0Y_1 + X_2X_3 = 0, \quad Q_4 : X_1X_0 + Y_2Y_3 = 0, \\ Q_2 &: X_0Y_2 - X_1X_3 = 0, \quad Q_5 : X_2Y_0 - Y_1Y_3 = 0, \\ Q_3 &: X_0Y_3 + X_2X_3 = 0, \quad Q_6 : X_3Y_0 + Y_1Y_2 = 0. \end{aligned}$$

The six diagonal planes also give 2×2 determinants, but only three of these are linearly independent,

$$\begin{aligned} Q_7 &: X_0Y_0 + X_1Y_1 = 0, \quad Q_7 - Q_8 : X_1Y_1 + X_2Y_2 = 0, \\ Q_8 &: X_0Y_0 - X_2Y_2 = 0, \quad Q_7 - Q_9 : X_1Y_1 - X_3Y_3 = 0, \\ Q_9 &: X_0Y_0 + X_3Y_3 = 0, \quad Q_9 - Q_8 : X_2Y_2 + X_3Y_3 = 0. \end{aligned}$$

A general quadric in $\mathbb{P}^7$ can be specified by a symmetric 8×8 matrix. Such a matrix will have 36 independent entries. Substituting the parameterisation for the Segre variety given above into the general quadric gives a linear equation in the entries of the symmetric matrix. For the Segre variety to lie entirely on a quadric the coefficients of the monomials must all vanish. There are 27 such terms and hence 27 linear homogeneous equations in the 36 matrix entries. So we should have expected that the Segre variety lies in a linear system of $36 - 27 = 9$ quadrics.

The degree of the Segre variety is well known to be six, see for example [7]. This fact can also be easily computed by slightly extending the arguments given in [1] for the degree of two factor Segre varieties.

2.2 Line Symmetric 6R Mechanisms

A line symmetric motion is defined by a ruled surface, called the base surface of the motion. Given a fixed position and orientation for the rigid body, the motion is produced by performing a π-rotation of the body about successive generating lines in the base surface.

If the end-effector of a $3R$ linkage performs a line-symmetric motion then at each instant the linkage can be reflected in the generating line of the base surface to produce a reflected linkage. Since a π-rotation is a proper rigid-body displacement the reflected linkage will have the same design-parameters (link-lengths, twist angles and offsets), as the original. Hence, if such a motion is possible, it will be produced by joining the end-effectors of the original and reflected linkages.

In [5] it was shown that a line symmetric rigid motion lies in the intersection of the Study quadric with a certain type of 5-plane in $\mathbb{P}^7$. The 5-plane is determined by the initial line in the base surface of the motion, ℓ_0 and can be written as the dual quaternion equation, $g\ell_0^- + \ell_0 g^- = 0$, where g is a variable dual quaternion. The intersection of the Segre variety with such a 5-plane will be a curve of degree 6, in general. We may conclude that such a motion is therefore possible and that the coupler bar of a general line symmetric $6R$ mechanism will follow a degree 6 line symmetric motion.

This curve has genus one, that is the curve is elliptic. To see this note that a curve of degree $n+1$ in $\mathbb{P}^n$ is always either genus 0 or 1, see [6, Chap. 2 Sect. 4.1]. Our curve cannot be rational since it lies on 9 quadrics. It is not hard to show that a general rational sextic in a $\mathbb{P}^5$ lies on a maximum of 8 quadrics. This is a simple extension of the well known fact that a rational quartic curve in $\mathbb{P}^3$ can only lie in one quadric but an elliptic quartic is generally the intersection of two quadrics [8].

The above argument can be extended to other cases. Recall, [4], that a Schoenflies motion is determined by the intersection of the Study quadric with a 5-plane. Hence if we join the end-effector of a $3R$ linkage to a Schoenflies motion generator, a SCARA robot for example, we should expect that the coupler bar of this closed loop linkage will follow motion represented by an elliptic sextic curve in the Study quadric.

2.3 Homology in the Study Quadric

A detailed account of the homology of the Study quadric may be found in [3, Sect. 11.5]. Here only the homology in dimension 3 is required. This is generated by the homology classes of arbitrary A-planes and B-planes. Recall that in the Study quadric there are two families of 3-plane called A-planes and B-planes. The class of a general A-plane will be called σ_A and σ_B will denote the class of a general B-plane. In general, two A-planes or two B-planes are disjoint, hence in homology we have the relations, $\sigma_A \cap \sigma_A = \sigma_B \cap \sigma_B = 0$. However a general A-plane will meet a general B-plane in a point: $\sigma_A \cap \sigma_B = 1$. A 3-dimensional subvariety of the Study quadric will have a homology class $m\sigma_A + n\sigma_B$ where the integers m and n give the generic number of intersections with a B-plane and an A-plane respectively. The subvariety will also be a subvariety of $\mathbb{P}^7$ and hence will have a degree—the number of intersections with a general 4-plane. Such a 4-plane intersects the Study quadric in subvariety of class $\sigma_A + \sigma_B$. The degree of the original subvariety is thus given by,

$$d = (\sigma_A + \sigma_B) \cap (m\sigma_A + n\sigma_B) = m + n.$$

In general the intersection of two subvarieties of the Study quadric with homology classes $m_1\sigma_A + n_1\sigma_B$ and $m_2\sigma_A + n_2\sigma_B$ will be given by

$$(m_1\sigma_A + n_1\sigma_B) \cap (m_2\sigma_A + n_2\sigma_B) = m_1n_2 + n_1m_2$$

points. This is analogous to Halphen's theorem for line congruences.

The homology class of the general $3R$ linkage was given in [3, Sect. 11.5.1], here a simpler proof is given. Since the degree of the Segre variety is 6 the homology class of the 3-dimensional variety must be $m\sigma_A + n\sigma_B$ where $m + n = 6$. So we only need to find the intersection of the Segre variety with an general A-plane (or B-plane) . Recall that the motions allowed by an S joint are simply an A-plane in the Study quadric. Hence the number of intersections between the Segre variety and a general A-plane can be counted as the number of assembly configurations of an $RRRS$ loop. This loop can also be decomposed as the intersection of an RR dyad and an RS linkage. The RS linkage produces rigid displacements lying on a $\mathbb{P}^1 \times \mathbb{P}^3$ Segre variety. This variety lies entirely within the Study quadric and by general arguments can be shown to have degree, $\binom{1+3}{1} = 4$, see [1]. On the other hand the variety of displacements produced by the RR linkage is well known to be the intersection of the Study quadric with a 3-plane, [3, Sect. 11.4]. The intersection of these two varieties thus generally consists of 4 points. Hence the homology class of the variety produced by the $3R$ linkage is $2\sigma_A + 4\sigma_B$.

In a very simple manner this result gives the general number of postures of a $6R$ serial robot, and equivalently the general number of assembly modes for a single loop $6R$ mechanism. The number of postures is given by the intersection of two such varieties,

$$(2\sigma_A + 4\sigma_B) \cap (2\sigma_A + 4\sigma_B) = 2 \times 4 + 4 \times 2 = 16.$$

It is also possible to find the general number of postures for a $6R$ robot with a $3R$ wrist. The wrist behaves like an S joint, but there are generally two postures of the $3R$ wrist for any rotation. Thus the homology class of the $3R$ wrist is $2\sigma_A$. Hence the general number of postures for the general $6R$ serial robot with a $3R$ wrist will be $2\sigma_A \cap (2\sigma_A + 4\sigma_B) = 2 \times 4 = 8$.

3 The Bennett *RRR* Linkage

The condition for the determinant of the matrix M to vanish will be studied here. Recall the significance of this relation; when $\det(M) = 0$ the variety of rigid displacements of the end-effector of the $3R$ linkage will not be the Segre variety $\mathbb{P}^1 \times \mathbb{P}^1 \times \mathbb{P}^1$. Rather, it will the the projection of the Segre variety to a subspace determined by the column space of M.

To begin, suppose that either $\sin\alpha_1$ or $\sin\alpha_2$ (but not both) vanish, in this case $\det(M) = 0$ is only possible if $d_1 = 0$ or $d_2 = 0$ respectively. This implies that joints 1 and 2 coincide or in the second case joints 2 and 3 are the same. Both cases are

degenerate and will not be considered further. Next suppose both $\sin\alpha_1$ and $\sin\alpha_2$ vanish. The equation $\det(M) = 0$ is satisfied and the linkage is planar as all joint are parallel.

In other cases we may assume that neither $\sin\alpha_1$ nor $\sin\alpha_2$ vanish and so we can divide Eq. (2) by $\sin^4\alpha_1 \sin^4\alpha_2$ to get,

$$h^4 + 2h^2\left(\frac{d_1^2}{\sin^2\alpha_1} + \frac{d_2^2}{\sin^2\alpha_2}\right) + \left(\frac{d_1^2}{\sin^2\alpha_1} - \frac{d_2^2}{\sin^2\alpha_2}\right)^2 = 0.$$

Treating this as a quadratic in h^2 gives two potential solutions; $h^2 = -((d_1/\sin\alpha_1) \pm (d_2/\sin\alpha_2))^2$. Both these solutions give imaginary values for h and hence can be rejected.

The final possibility is $h = 0$ and $d_1^2 \sin^2\alpha_2 = d_2^2 \sin^2\alpha_1$. This gives two possible cases, if $d_1 = d_2 = 0$ then the linkage is spherical. In the final case neither link-length d_1 nor d_2 are zero and this will be called the Bennett linkage. The above shows that for a non-degenerate $3R$ linkage, the subvariety of the Study quadric generated by the end-effector fails to be the Segre variety $\mathbb{P}^1 \times \mathbb{P}^1 \times \mathbb{P}^1$ in just three possible cases; the planar, spherical and Bennett linkages.[1]

To satisfy the conditions for the Bennett linkage we set $h = 0$, $d_1 = d\sin\alpha_1$ and $d_2 = d\sin\alpha_2$ in the following. Call the displacement variety generated by the end-effector of a Bennett linkage X. This variety is given by a linear projection of the Segre variety $\mathbb{P}^1 \times \mathbb{P}^1 \times \mathbb{P}^1$. The centre of this projection is the line of points satisfying $M\mathbf{p} = \mathbf{0}$. This line is the join of the pair of points,

$$\mathbf{p}_1^T = \big(\sin(\alpha_2 - \alpha_1), 0, 0, 0, 0, \sin\alpha_1 + \sin\alpha_2, \sin(\alpha_2 - \alpha_1), -\sin\alpha_1 - \sin\alpha_2\big),$$
$$\mathbf{p}_2^T = \big(0, \sin\alpha_1 - \sin\alpha_2, \sin(\alpha_1 + \alpha_2), \sin\alpha_2 - \sin\alpha_1, \sin(\alpha_1 + \alpha_2), 0, 0, 0\big).$$

Substituting this line into the nine quadrics from Sect. 2.1 above, reveals that the centre of projection meets the Segre variety at a pair of points. So the degree of X is $6 - 2 = 4$.

The image of the projection X lies in a 5-plane. This is given by the intersection of two hyperplanes of the form $\mathbf{q}^T\mathbf{g} = 0$, where $\mathbf{g} = (a_0, a_1, a_2, a_3, c_0, c_1, c_2, c_3)^T$ as in Sect. 2. Two linearly independant hyperplanes are given by,

$$\mathbf{q}_1^T = \big(0, 0, d(\sin\alpha_1 - \sin\alpha_2), 0, 0, 0, \sin(\alpha_2 - \alpha_1), 1 - \cos(\alpha_2 - \alpha_1)\big)$$
$$\mathbf{q}_2^T = \big(0, d(\sin\alpha_1 + \alpha_2), 0, 0, \cos(\alpha_1 + \alpha_2) - 1, -\sin(\alpha_1 + \alpha_2), 0, 0\big)$$

This has an immediate application to line-symmetric $6R$ mechanisms. As in Sect. 2.2 above, we can form a line-symmetric $6R$ mechanism by joining the

[1] This results is due to Josef Schicho.

end-effectors of a Bennett linkage and its line symmetric linkage. The motion of the coupler link with then be the intersection of the variety X with a 5-plane determined by the symmetry. The result is a quartic curve in the Study quadric in general, this curve lies in the 3-plane given by the intersection of the symmetry 5-plane with the 5-plane that the variety X lies in.

The homology class of X will have the form $m\sigma_A + n\sigma_B$ where $m + n = 4$. Take a general A-plane, this will meet the $\mathbb{P}^5$ containing X in a line. The intersection of the Study quadric with this $\mathbb{P}^5$ is a 4-dimensional quadric. Since the line and the 3-dimensional variety X both lie in this 4-dimensional quadric they will generally meet in n points. Take a 2-plane which is tangent to the quadric along the line. The interstion of this 2-plane with the quadric is just the line but with multiplicity two. The intersection of X with the 2-plane has degree 4 since X has degrre 4. But this intersection can consist of at most 2 points on the line, since each point must have multiplicity at least 2. The same argument applies to a general B-plane hence we see that we must have $m = n = 2$. The homology class of X is $2\sigma_A + 2\sigma_B$.

This result has several simple consequences for robots and mechanisms. A $6R$ serial robot consisting of such a Bennett linkage and a $3R$ wrist will, in general, have $2\sigma_A \cap (2\sigma_A + 2\sigma_B) = 4$ postures. If the robot consists of two Bennett linkages then we should expect $(2\sigma_A + 2\sigma_B) \cap (2\sigma_A + 2\sigma_B) = 8$ postures in general. If the robot consists of a general $3R$ linkage and a Bennett $3R$ we expect $(2\sigma_A + 4\sigma_B) \cap (2\sigma_A + 2\sigma_B) = 12$ postures in general.

An *RRRS* single loop mechanism, where the *RRR* linkage has the form of a Bennett linkage as described above will have $\sigma_A \cap (2\sigma_A + 2\sigma_B) = 2$ assembly configurations in general.

4 Conclusions

Homological methods are powerful tools for solving enumerative problems in kinematics. The results are, of course, subject to the usual cautions: Solutions must be properly counted, solutions can be complex and in exceptional cases the set of solutions will have a higher dimension than predicted.

Finally it is hoped that the methods outlined here can be used to study the many kinematic structures formed from several Bennett linkages or mechanism.

References

1. Harris, J.: Algebraic Geometry a First Course. Springer, New York (1992)
2. Husty, M.L., Pfurner, M., Schröcker, H.-P.: A new and efficient algorithm for the inverse kinematics of a general serial $6R$ manipulator. Mech. Mach. Theory **42**(1), 66–81 (2007)
3. Selig, J.M.: Geometric Fundamentals of Robotics. Springer, New York (2005)
4. Selig, M.: Quadratic constraints on rigid-body displacements. ASME J. Mech. Robot. **2**(4), 041009 (2010)

5. Selig, J.M., Husty, M.: Half-turns and line symmetric motions. Mech. Mach. Theory **46**(2), 156–167 (2011)
6. Semple, J.G., Roth, L.: Introduction to Algebraic Geometry. Clarendon Press, Oxford (1985)
7. Sturmfels, B.: Gröbner Bases and Convex Polytopes. American Mathematical Society, Providence (1996)
8. Telling, H.G.: The rational quartic curve. In: Space of Three and Four Dimensions: Being an Introduction to Rational Curves. Cambridge Tracts in Mathematics and Mathematical Physics, vol. 34. Cambridge University Press, Cambridge (1936)

Force Capability Polytope of a 4RRR Redundant Planar Parallel Manipulator

Leonardo Mejia, Henrique Simas and Daniel Martins

Abstract In this chapter the characteristic *force capability polytope* of a 4RRR parallel manipulator is obtained from the optimization of its static equations. In order to solve the problem regarding the global optimization, an evolutionary algorithm known as Differential Evolution (*DE*) is used. The objective function of the optimization problem of force capability is defined by employing the Screws Theory and Davies's method as a primary mathematical tool. Finally, some force capability polytopes are obtained for different kinematic positions of the manipulator.

Keywords Force capability polytope · Optimization · Screws theory · Davies's method

1 Introduction

The task space capabilities of a manipulator to perform motion and/or to exert forces and moments are of fundamental importance in robotics. Their evaluation can be useful to determine the structure and the size of a manipulator that best fit the designer's requirements or they can be used to find a better configuration or a better operation point for a manipulator to perform a given task [2].

In robotics, the ***force capability*** of a manipulator is defined as the maximum wrench that can be applied (or sustained) by a manipulator for a given pose based on the limits of the actuators. By considering all possible directions of the applied wrench

L. Mejia (✉) · H. Simas · D. Martins
Federal University of Santa Catarina, Florianópolis, Brazil
e-mail: leonardo.mejia.rincon@posgrad.ufsc.br

H. Simas
e-mail: hsimas@gmail.com

D. Martins
e-mail: daniel.martins@ufsc.br

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_10,

or by considering specific directions along spatial trajectories, a force capability plot can be generated for the given pose [5]. The force capability of a manipulator depends on its design, posture, actuation limits and ***redundancies*** [8].

Redundant actuation in parallel manipulators can be divided into three categories. The first category features actuating some of the passive joints within the branches of a parallel manipulator. *The second category of redundant manipulators are those that feature additional branches beyond the minimum necessary to actuate the device (as in the present study)*. Finally, the third category of redundantly-actuated parallel manipulators are devices that are a hybrid of the first two categories [5].

With redundant actuation, the solution to the inverse force problem (given the desired wrench to be applied by the platform, what are the required joint torques/forces) no longer has a unique solution. An infinity of possible solutions exists to the inverse force problem. This infinity of possible solutions allows the joint torques/forces to be optimized [5].

The main objective of this study is to develop a method to obtain the force capability polytope of a 4RRR parallel manipulator in static or quasi-static conditions.

2 Geometric Representation of a 4RRR Planar Parallel Manipulator

Parallel manipulators usually consist in a mobile platform connected to a fixed platform by several branches in order to transmit the movement. Generally, the number of branches of parallel manipulators is equals to their degree of freedom (DoF), and the motors are usually located near the fixed base [7].

In this chapter, a "*4RRR Redundant Planar Parallel Manipulators (RPPM)*" is studied. In this parallel manipulator, the fixed and mobile platforms are joined by using four branches. Each branch has three rotational joints whose axes are perpendicular to the $(x - y)$ plane, and the first of the three joints in each branch is actuated, as shown in Fig. 1.

Furthermore, the mobile and fixed platforms are formed by squares with sides l_m and l_f respectively. The branches are formed by two links with lengths l_1 and l_2 respectively and the angle ϕ represents the orientation of the mobile platform. For the studied manipulator, the link lengths and platform edge lengths are specified as $l_1 = l_2 = l_m = 0.2\,\text{m}$, $l_f = 0.5\,\text{m}$, the manipulator's end effector is located in $[0.25m, 0.25m]$, the mobile platform is oriented in $\phi = 0°$ and the maximum torque capability for each actuated joint of the manipulator is $\pm 4.2\,\text{Nm}$.

3 Statics of the Manipulator

In the static analysis of manipulators, the goal is to determine the force and moment requirements in the joints. It is possible to apply forces and moments in the mechanism joints to analyse the efforts obtained in the end actuator, or to apply external

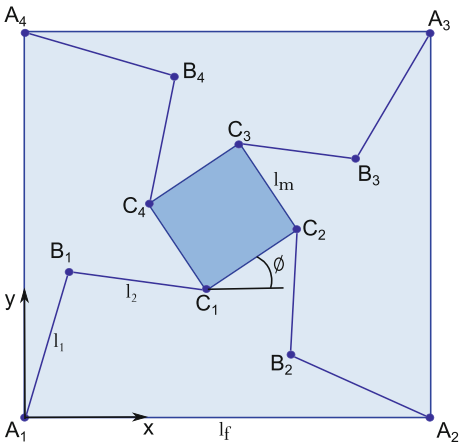


Fig. 1 Schematic representation of 4RRR RPPM

forces and calculate the necessary forces and moments in the joints to balance these external forces.

The wrenches (force and moments) applied (or sustained) *by the end effector* can be represented by the vector $F = [F_x, F_y, M_z]^T$, where F_x and F_y denotes the force in the directions "x" and "y" respectively, and M_z denotes the moment around the "z" axis. The actuator torques of joints A_1, A_2, A_3 and A_4 (denoted by the vector τ) are respectively τ_{A_1}, τ_{A_2}, τ_{A_3} and τ_{A_4}, in other words $\tau = [\tau_{A_1}, \tau_{A_2}, \tau_{A_3}, \tau_{A_4}]^T$.

Using the formalism presented in ***Davies*** [3], the primary variables are known, and the secondary variables are the unknown variables. Considering that the primary variables are the actuation torques, the direct force equation can be written in the form: $V_{27\times1} = [\widehat{A}_N]_{27\times4} \cdot \tau_{4\times1}$. Where $\widehat{A}_N$ is the unitary action matrix and V is a vector comprising all the unknown wrenches in the manipulator.

The $\widehat{A}_N$ matrix can be obtained using ***graph theory***, ***screw theory*** and ***Kirchhoff-Davies cutset law*** [1]. The complete manipulators action graph is shown in Fig. 2, where the edges are the joints and the vertices are the manipulator links. Existing wrenches in each joint are represented by the symbol \$.

The cutset law states that: when a manipulator is in static equilibrium, the sum of wrenches acting in a single cut must be zero. Each one of the k cuts divides the manipulator in subsets of links and joints, where in each subset, the static equilibrium must be preserved [8].

Since the manipulator is planar, the space dimension λ is three and only the F_x, F_y, and M_z wrench components are considered, while F_z, M_x, and M_y are always equal to zero [8] and will not be represented in this chapter.

For each ***revolute joint "i"***, the wrench can be written as shown in Eq. (1), where x_i and y_i are the location of the joint axis given in Cartesian coordinates, F_{x_i} and F_{y_i} are respectively the forces in directions x and y of the joint, and τ_i is the joint's actuation torque. If the joint is passive, τ_i is zero and the last term of equation Eq. (1) vanishes [8].

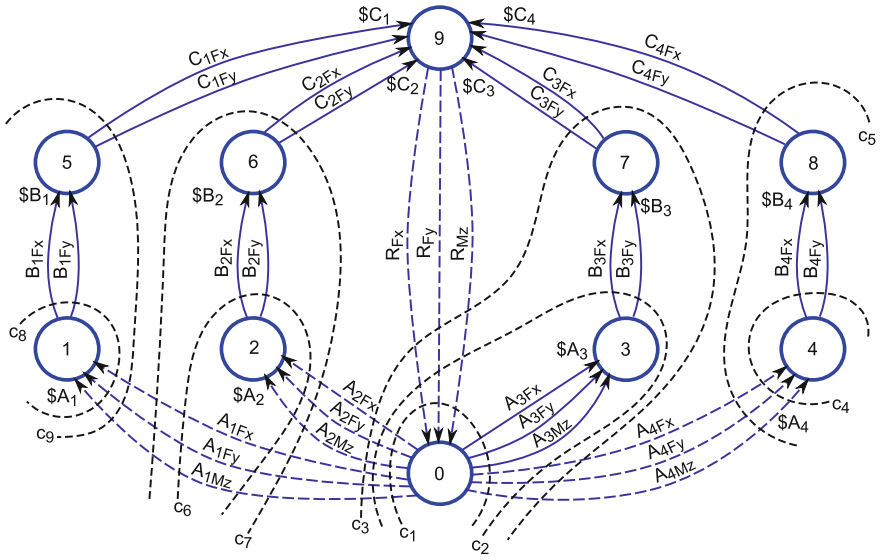


Fig. 2 Manipulator action graph of a 4RRR RPPM

$$\$_i = \begin{bmatrix} -y_i \\ 1 \\ 0 \end{bmatrix} F_{x_i} + \begin{bmatrix} x_i \\ 0 \\ 1 \end{bmatrix} F_{y_i} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tau_i \tag{1}$$

For each cutset there are three independent equations that can be written in the matrix as shown in Eq. (2), where A_D is the cut action matrix, $\widehat{A}_N$ is the unitary cut action matrix, Ψ is the vector comprising the wrenches magnitudes, λ is the space dimension, and C represent the manipulator's gross [1, 3, 8].

$$\Sigma\$ = [A_D]_{\lambda \times C} = [\widehat{A}_N]_{\lambda \times C}\{\Psi\}_{C \times 1} = \{0\}_{\lambda \times 1} \tag{2}$$

Algebraic manipulation of Eq. (2) allow us to obtain the general expressions for the forces and the moment in the manipulator's end effector as shown in Eq. (3), in these equations F_x and F_y denotes the force in the directions "x" and "y" of the manipulator's end effector, M_z denotes the moment around the "z" axis of the manipulator's end effector, the $k_1, \ldots, k_{12}$ terms represent the kinematic expressions as functions of the manipulator's joint's positions, and the $\tau_{A_1}, \tau_{A_2}, \tau_{A_3}, \tau_{A_4}$ terms represent the torques in the actuated joints.

$$\begin{bmatrix} F_x \\ F_y \\ M_z \end{bmatrix} = \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \\ k_5 & k_6 & k_7 & k_8 \\ k_9 & k_{10} & k_{11} & k_{12} \end{bmatrix} \cdot \begin{bmatrix} \tau_{A_1} \\ \tau_{A_2} \\ \tau_{A_3} \\ \tau_{A_4} \end{bmatrix} \tag{3}$$

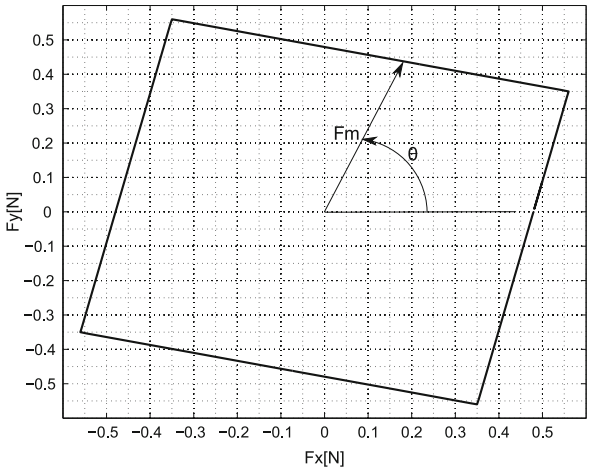


Fig. 3 Force capability map of a 4RRR RPPM

4 Optimization Problem

The aim of the optimization problem studied in this chapter, is to maximize the force F_m applied or sustained for the manipulator in a given direction by the angle θ, while the moment is imposed as a constant. A force capability map (Fig. 3) is obtained when all the possible directions of the maximum force F_m are considered. A force capability map similar to the one shown in Fig. 3 was first studied by Nokleby [4, 5] for a 3RRR Planar Parallel Manipulator.

The optimization problem can be described as the process in which the torque in the actuators τ_{A_1}, τ_{A_2}, τ_{A_3} and τ_{A_4} must be optimized in order to maximize the pure force and minimize the error in the imposed value for the moment in the manipulator's end effector. This optimization must be done in all possible directions given for the angle θ. In our simulations, were used 360 repetitions, one repetition per degree of the angle θ, and the imposed moment was increased positively and negatively until the force in any direction was zero.

4.1 Objective Function and Diferential Evolution (DE) Algorithm

The objective function used in the optimization process is shown in Eq. (4), where the terms F_x and F_y are the components of the force obtained in each iteration of the optimization, α_d is the desired angle of the application of the force, α_o is the obtained angle of the application of the force as a function of the F_x and F_y components, M_z is the moment obtained in the manipulator's end effector, M_k is the constant moment

imposed in the manipulator's end effector and finally the "P" term is the penalization of the objective function.

In Eq. (4), the $|\alpha_d - \alpha_o/\alpha_d|$ term minimizes the normalized error between the obtained and desired force direction, the $\left|3l_1\tau_{An_{\max}}/\sqrt{F_x^2 + F_y^2}\right|$ term maximizes the normalized force obtained, and the $|M_z - M_k/M_k|$ term minimizes the normalized error between the obtained moment and the desired moment in the manipulator's end effector.

$$F_{obj} = \left|\frac{\alpha_d - \alpha_o}{\alpha_d}\right| + \left|\frac{3l_1\tau_{An_{\max}}}{\sqrt{F_x^2 + F_y^2}}\right| + \left|\frac{M_z - M_k}{M_k}\right| + P \tag{4}$$

The penalization term "P" included in Eq. (4) is activated when the condition $[\tau_{An_{\min}} \leq \tau_{An} \leq \tau_{An_{\max}}]$ is not satisfied, this condition is imposed as the maximum admissible torque in the actuators τ_{An}. In the present chapter were used $\tau_{An_{\min}} = -4.2\,\text{Nm}$ and $\tau_{An_{\max}} = 4.2\,\text{Nm}$.

In order to solve the problem regarding the global optimization, an evolutionary algorithm known as Differential Evolution (*DE*) was used. *DE* is a very simple population based, stochastic function minimizer and very powerful at the same time. This algorithm is commonly accepted as one of the most successful algorithms for the global continuous optimization problem [9].

DE optimizes a problem by maintaining a population of candidate solutions, and creating new candidate solutions by combining existing ones, according to its simple formula, and then keeping whichever candidate solution has the best score or fitness on the optimization problem at hand. In this way, the optimization problem is treated as a black box that merely provides a measure of quality given a candidate solution, and therefore, the gradient is not needed [9].

The performance of the DE algorithm is sensitive to the mutation strategy and respective control parameters, such as the population size (NP), crossover rate (CR), and the mutation factor (MF). The best settings for control parameters can be different for different optimization problems, and the same functions with different requirements, for consumption time and accuracy [8].

In this study the parameters $NP = 30$, $CR = 0.8$, and $MF = 0.5$ were used, as suggested in [6], and the maximum iteration number was established in 4,000.

5 Results

From the optimization of the objective function shown in Eq. (4) and using the topology shown in Sect. 2, is possible to obtain the maximum force in a desired direction. By repeating the optimization process for each possible direction, the force capability map as shown in Fig. 3 is obtained. Finally, by varying the value

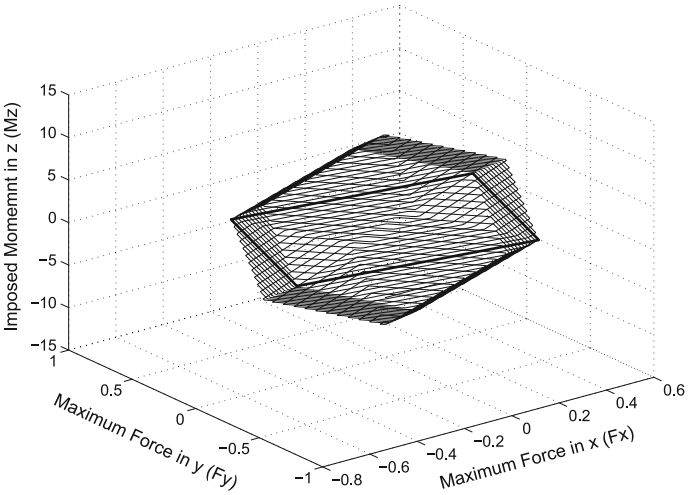


Fig. 4 Force capability polytope of a 4RRR RPPM (first studied case)

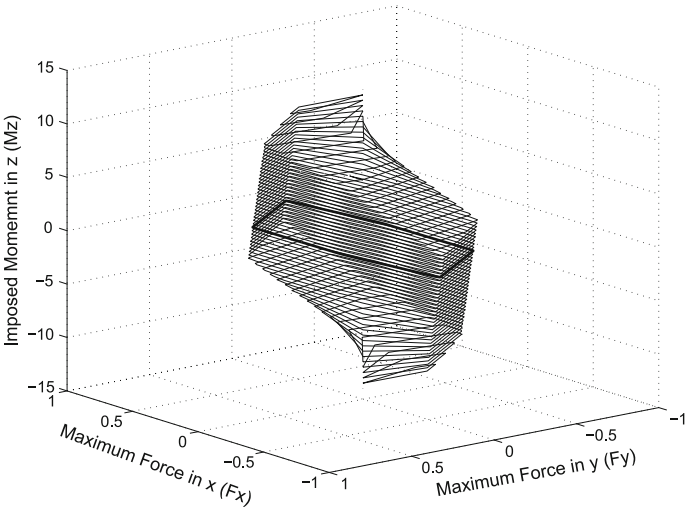


Fig. 5 Force capability polytope of a 4RRR RPPM (second studied case)

for the imposed moment in the manipulator's end effector, the manipulator's force capability polytope as shown in Fig. 4 is obtained.

Using the same manipulator shown in Fig. 1, but changing the end effector's position to $[0.35m,\ 0.35m]$, the orientation of the mobile platform to $\phi = 8°$, and inverting the working mode of the branch A_1, B_1, C_1, is obtained the manipulator's force capability polytope shown in Fig. 5.

Finally, using the same strategy, is possible to obtain the manipulator's force capability polytope for each position into the manipulator's workspace.

6 Conclusions

This chapter presents a method to obtain the force capability polytope in a 4RRR redundant parallel manipulator optimizing the torque in the actuators of the manipulator and imposing the moment in the manipulator's end effector. The optimization problems were solved using *DE* algorithms.

The force capability polytope is composed by the superposition of several force capability maps and depends on several parameters as the manipulator's kinematic position, orientation, working mode and redundancies, two study cases were shown in order to illustrate these dependencies.

The present study may be extended in various ways. Manipulators with different DOFs, kinematic chains and including dynamic behavior may be studied, and the minimization of the force and the maximization of the moment may be considered in future researches.

References

1. Cazangi, H.R.: Aplicação do método de davies para análise cinemática e estática de mecanismos de múltiplos graus de liberdade. Ph.D. thesis, Universidade Federal de Santa Catarina, Brazil (2008)
2. Chiacchio, P., Bouffard-VercelliO, Y., Pierroto, F.: Evaluation of force capabilities for redundant manipulators. In: IEEE International Conference on Robotics and Automation, pp. 3520–3525 (1996)
3. Davies, T.H.: Mechanical networks: wrenches on circuit screws. Mech. Mach. Theory **18**, 107–112 (1983)
4. Nokleby, S.B., Firmani, F., Zibil, A., Podhorodeski, R.P.: Force moment capabilities of redundantly actuated planar parallel architectures. In: 12th IFToMM 2007 World Congress, pp. 17–21 (2007)
5. Nokleby, S.B., Fisher, R., Podhorodeski, R.P., Firmani, F.: Force capabilities of redundantly-actuated parallel manipulators. Mech. Mach. Theory **40**, 578–599 (2004)
6. Storn, R., Price, K.: Differential evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012. International Computer Science Institute (2005)
7. Tsai, L.W.: Robot Analysis and Design: the Mechanics of Serial and Parallel Manipulators, 1st edn. Wiley, New York (1999)
8. Weihmann, L., Martins, D., Coelho, L.S.: Force capabilities of kinematically redundant planar parallel manipulators. In: 13th World Congress in Mechanism and Machine Science, pp. 483–483 (2011)
9. Weihmann, L., Martins, D., dos Santos Coelho, L.: Modified differential evolution approach for optimization of planar parallel manipulators force capabilities. In: Expert Systems with Applications, vol. 39. http://dx.doi.org/10.1016/j.eswa.2011.11.099 (2012)

Motion Planning of Non-holonomic Parallel Orienting Platform: A Jacobian Approach

Krzysztof Tchoń and Janusz Jakubiak

Abstract This chapter addressed the motion planning problem for a non-holonomic parallel orienting mechanism whose kinematics are represented by a driftless control system on $SO(3)$. A coordinate-free, Jacobian motion planning algorithm is proposed. A normal form of singularity locus is derived. Two example motion planning problems are solved numerically.

Keywords Non-holonomic mechanism · Motion planning · Jacobian algorithm · Singularities

1 Introduction

The parallel non-holonomic orienting platform is a robotic mechanism designed at the Institute of Robotics and Industrial Informatics, UPC. The design guidelines of this platform, defined in [1, 3], rely on replacing a prismatic actuated joint by a passive joint introducing a non-holonomic constraint. Thanks to this constraint the platform can be oriented by means of two actuators instead of three. However, like in other non-holonomic mechanisms, the reduction of actuation is usually made at the expense of more complicated control [5]. A view of the platform used in [2] as an experimental testbed for control algorithms is shown in Fig. 1 (left part). The right part of this figure shows schematically the design idea. The platform is represented by a ball. Two linear actuators attached to the ball at b_1 and b_2 change its orientation, while a rolling disk prevents the ball's rotation around the r axis coplanar with

K. Tchoń (✉) · J. Jakubiak
Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology, Wrocław, Poland
e-mail: krzysztof.tchon@pwr.wroc.pl

J. Jakubiak
e-mail: janusz.jakubiak@pwr.wroc.pl

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_11,

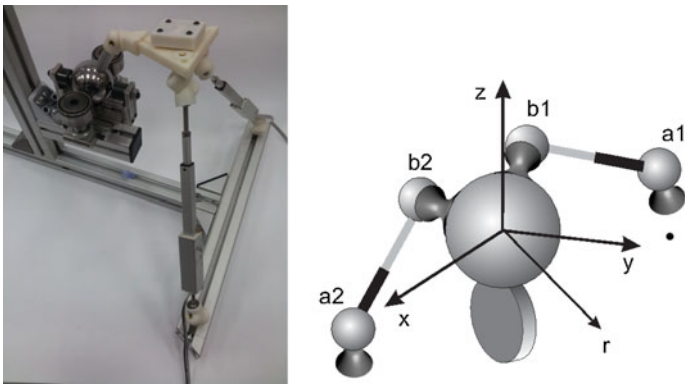


Fig. 1 Non-holonomic parallel orienting platform (courtesy of Prof. F. Thomas)

the disk, and passing through the center of the ball. In Grosch and Thomas [2] the motion planning problem of the non-holonomic parallel orienting platform has been studied using the quaternion representation of rotations, and the theory of bilinear control systems. Although geometrically insightful, that approach uses a specific arrangement of platform's actuators, and is not adopted to challenge the problem of mechanical singularities of the platform. In this chapter we propose a Jacobian motion planning algorithm devised for general velocity affine mechanical systems [4]. The algorithm is not restricted to any specific values of design parameters, and works in a coordinate-free way. An attempt at applying this kind of algorithm to a parallel non-holonomic robot has been made in [8]. In this chapter we assume that the platform's motions remain completely within the singularity-free region. However, by employing sufficiently rich controls and the multiple-task approach, our method should be able to solve the motion planning problem and keep the platform's trajectories away from singularities.

This chapter is composed in the following way. Section 2 presents the kinematics model of the platform and a local model of its singularities. Section 3 is devoted to the motion planning algorithm. Performance of the algorithm is illustrated with two motion planning problems solved in Sect. 4. Section 5 concludes the chapter. Mathematical details are collected in Appendix.

2 Modeling

Taking into account a specific position of the attachment points a_1, a_2, b_1, b_2, and of the vector r (see Fig. 1), the kinematic model of the platform takes the following form [2]

$$\dot{R} = [L(R)u]\, R, \tag{1}$$

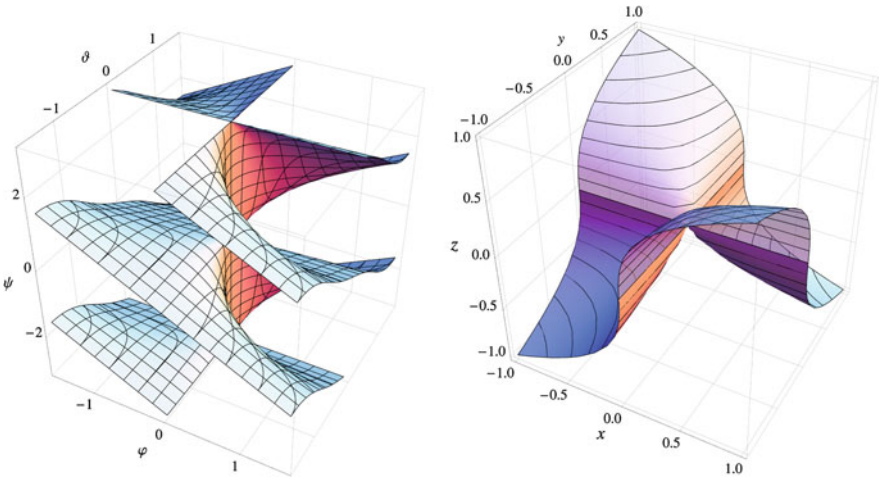


Fig. 2 Singularity locus: global and local, around $R = \mathbb{I}_3$

where $R = [r_{ij}] \in SO(3)$ is a rotation matrix, and the control vector $u \in \mathbb{R}^2$. The operator $[\cdot]$ denotes the isomorphism between $\mathbb{R}^3$ and the Lie algebra $so(3)$ of skew symmetric 3×3 matrices, such that for any $v, w \in \mathbb{R}^3$ $[v \times w] = [v][w] - [w][v]$. Admissible control functions entering (1) will be taken from the Hilbert space $L_2^2[0, T]$ of Lebesgue square integrable functions $u(\cdot) : [0, T] \to \mathbb{R}^2$. Given an initial condition $R_0 \in SO(3)$ and a control function $u(\cdot) \in L_2^2[0, T]$, the system (1) produces a trajectory $R_t = \phi_{R_0,t}(u(\cdot))$ evolving on $SO(3)$. In this chapter, following [2], we shall study the system (1) defined by

$$L(R) = \frac{1}{\Delta(R)} \begin{bmatrix} r_{21} & r_{12} \\ -r_{21} & -r_{12} \\ -r_{31} & r_{32} \end{bmatrix}, \tag{2}$$

with $\Delta(R) = r_{12}r_{31} + r_{21}r_{32}$. By design, the rotations satisfying the equation $\Delta(R) = 0$ are inaccessible configurations of the platform, called mechanical singularities. A global view of these singularities in the Roll-Pitch-Yaw coordinates is shown in Fig. 2 (left). A usual way of locally visualizing a singularity consists in using its normal form representation. It can be shown that in suitably chosen coordinates around the unit rotation the singularity can be described by a normal form $f(x, y, z) = z^3 + xy$. A derivation is presented in Appendix. A view of the singularity locus $f(x, y, z) = 0$ is displayed in Fig. 2 (right).

3 Motion Planning Algorithm

The following motion planning problem will be addressed in (1): For an initial orientation R_0 and a desired R_d, find a control function $u_d(\cdot)$, such that the corresponding trajectory reaches R_d at a prescribed time instant $T > 0$, i.e. $R_T = R_d$. A motion planning algorithm that will be applied to the platform has been obtained along the lines described in [4]. A detailed derivation is deferred to Appendix, so here we shall present only the final result. The algorithm computes a curve $u_\theta(\cdot)$ of control functions parametrized by $\theta \in \mathbb{R}$, that converges to a solution of the motion planning problem, $\lim_{\theta\to+\infty} u_\theta(t) = u_d(t)$. Let $R_t(\theta) = \phi_{R_0,t}(u_\theta(\cdot))$ denote a trajectory of (1) under the control function $u_\theta(\cdot)$, and let the planning error along this curve be $e(\theta) = \log(R_T(\theta)R_d^T)$. Then, the motion planning algorithm consists in solving simultaneously the following set of differential equations

$$\begin{cases} \frac{\partial R_t(\theta)}{\partial t} = [L(R_t(\theta))u_\theta(t)]\, R_t(\theta), \quad R_0(\theta) = R_0, \\ \frac{\partial \Phi_\theta(T,t)}{\partial t} = -\Phi_\theta(T,t)A_\theta(t), \quad \Phi_\theta(T,T) = \mathbb{I}_3, \\ \frac{\partial M_\theta(t)}{\partial t} = B_\theta(t)B_\theta^T(t) + A_\theta(t)M_\theta(t) + M_\theta(t)A_\theta^T(t), \quad M_\theta(0) = 0, \\ \frac{\partial u_\theta(t)}{\partial \theta} = -\gamma\, B_\theta^T(t)\Phi_\theta^T(T,t)M_\theta^{-1}(T)l_T(\theta), \quad u_0(t) = u_{\theta=0}(t), \\ [l_T(\theta)] = \log(R_T(\theta)R_d^T). \end{cases} \tag{3}$$

This algorithm should be initialized by the initial orientation R_0 and the initial control function $u_0(t)$. The matrices $A_\theta(t)$ and $B_\theta(t)$ have been defined in the Appendix. The computation of the control function in the line 4 of the system (3) involves the Moore-Penrose inverse of the motion planning problem Jacobian (see Appendix), so this algorithm belongs to the family of Jacobian motion planning algorithms. The system (3) can be solved using either parametric or non-parametric approach [6]. In this chapter the parametric approach will be employed, relying on representing the control functions by truncated Fourier series. The number of terms in the representation should not be less than the dimension of $SO(3)$.

4 Computer Simulations

In order to illustrate the performance of the motion planning algorithm, we shall solve two example motion planning problems. In the calculations the parametric representation $u(t) = P(t)\lambda$ of the control functions is chosen, where the matrix $P(t)$ collects the base functions, and $\lambda \in \mathbb{R}^s$ denotes parameters, $s \geq 3$. In the assumed representation the curve $u_\theta(t) = P(t)\lambda_\theta$ of control functions is equivalent to a curve $\lambda_\theta \in \mathbb{R}^s$. Specifically, a substitution of the parametrized control into the integral equation (10) from Appendix leads to the equation

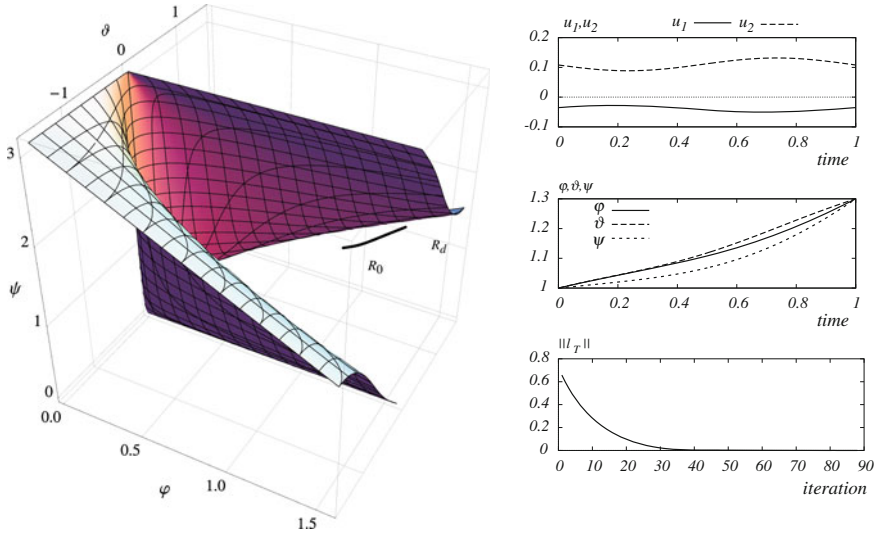


Fig. 3 Example 1: $R_d = RPY(1.3, 1.3, 1.3)$

$$J_{R_0,T}(\theta)\frac{\partial \lambda_\theta}{\partial \theta} = \int_0^T \Phi_\theta(T,t)B_\theta(t)P(t)dt\frac{\partial \lambda_\theta}{\partial \theta} = -\gamma l_T(\theta),$$

that can be solved for $\frac{d\lambda_\theta}{d\theta}$ using the Moore-Penrose inverse of the Jacobian $J_{R_0,T}(\theta)$, so that $\frac{d\lambda_\theta}{d\theta} = -\gamma J^{\#}_{R_0,T}(\theta)l_T(\theta)$. By integrating this equation by the fixed step Euler method we get the discrete motion planning algorithm

$$\lambda_{\theta+1} = \lambda_\theta - \gamma J^{\#}_{R_0,T}(\lambda_\theta)l_T(\theta), \quad \theta = 0, 1, \ldots.$$

Furthermore, it follows that the Jacobian $J_{R_0,T}(\theta)$ appears to be a solution of the differential equation $\frac{dJ_{R_0,t}(\theta)}{dt} = A(t)J_{R_0,t}(\theta) + B_\theta(t)P(t), \quad J_{R_0,0}(\theta) = 0$. In the examples a 6-dimensional representation of control functions is chosen, in the form $u = P(t)\lambda = \left(\lambda_1 + \lambda_2 \sin\frac{2\pi}{T}t + \lambda_3 \cos\frac{2\pi}{T}t,\ \lambda_4 + \lambda_5 \sin\frac{2\pi}{T}t + \lambda_6 \cos\frac{2\pi}{T}t\right)^T$. The initial orientation of the platform was set to $R_0 = RPY(1, 1, 1)$ (Roll-Pitch-Yaw angles $\varphi = \vartheta = \psi = 1$), which lies in the region free from singularities. The desired orientation was equal to $R_d = RPY(1.3, 1.3, 1.3)$ for the first example and $R_d = RPY(0.5, 0.5, 0.5)$ for the second one. The initial controls were chosen, respectively, as $u_0(t) = P(t)\lambda_0 = (0.1, 0.1)^T$ and $u_0(t) = P(t)\lambda_0 = (0.1, -0.1)^T$. In the both examples the motion planning horizon $T = 1$, the error decay rate $\gamma = 0.1$. The algorithm stops when the Euclidean norm of the planning error $||l_T||$ drops below 10^{-4}. The computations were made in Matlab, with `ode45` function used to compute the system trajectory and the Jacobian. The results are presented, respectively, in Figs. 3 and 4. The image on the left in each figure presents the

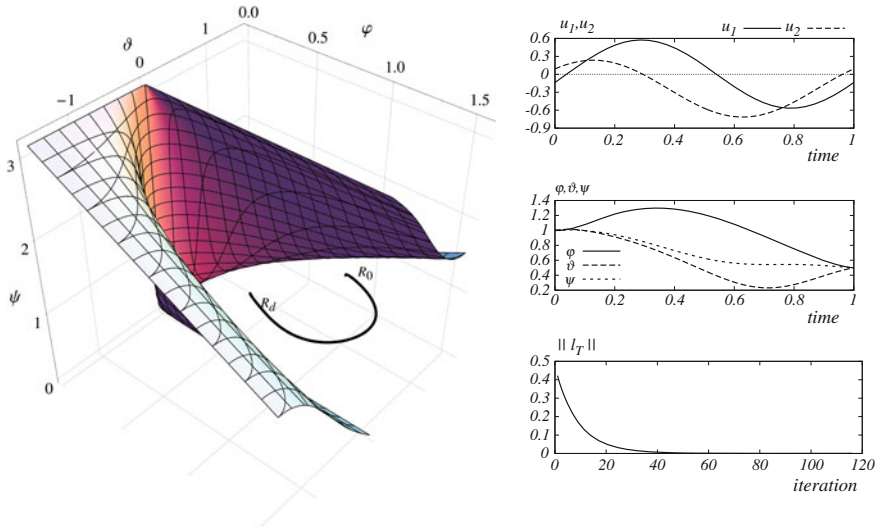


Fig. 4 Example 2: $R_d = RPY(0.5, 0.5, 0.5)$

resulting trajectory in RPY coordinates against the background of the singularity surface.

5 Conclusion

The examples demonstrate that the motion planning algorithm works efficiently on condition that the platform trajectory remains within the singularity-free region of the platform orientations. In order to prevent the trajectory from entering into these singularities, the primary motion planning task may be completed by a secondary task of avoiding singularities. For the specific arrangement of platform's actuators, a preliminary feedback can be applied to (1), and the motion planning problem solved by means of the imbalanced or singularity robust Jacobian methods. These issues will be addressed in a future chapter.

Acknowledgments This research was supported by the National Science Centre, Poland, under the grant decision No DEC-2013/09/B/ST7/02368. The authors are very much indebted to prof. Federico Thomas, CSIC-UPC, for having kindly provided them with a copy of the chapter [2] as well as with the photos presented in Fig. 1.

Appendix

Normal Form

Originally, the singularity locus of the non-holonomic orienting platform is given by the equation $\Delta(R) = r_{12}r_{31} + r_{21}r_{32} = 0$. Let's first use the Roll-Pitch-Yaw coordinates, $RPY(\varphi, \vartheta, \psi) = R(Z, \varphi)R(Y, \vartheta)R(X, \psi)$. Obviously, $RPY(0, 0, 0) = \mathbb{I}_3$. It is easily seen that in these coordinates $\Delta(\varphi, \vartheta, \psi) = \sin\varphi \sin\vartheta \cos\psi + \sin\psi(\sin\varphi\cos^2\vartheta - \cos\varphi\sin^2\vartheta)$. Next, we choose $\xi = \sin\vartheta\cos\psi$, $\eta = \sin\varphi \cdot \cos^2\vartheta - \cos\varphi\sin^2\vartheta$, and $\zeta = \sin\psi$. This is a well defined coordinate system around 0, so $\Delta(\xi, \eta, \zeta) = \xi\sin\varphi(\xi, \eta, \zeta) + \eta\zeta$. The function φ satisfies $\eta - \sin\varphi(\xi, \eta, \zeta)\cos^2\vartheta(\xi, \eta, \zeta) + \cos\varphi(\xi, \eta, \zeta)\sin^2\vartheta(\xi, \eta, \zeta) = 0$, that after suitable substitutions, converts into $-\sin\varphi(\xi, \eta, \zeta) + \eta + \frac{\xi^2(\cos\varphi(\xi,\eta,\zeta)+\sin\varphi(\xi,\eta,\zeta))}{1-\zeta^2} = 0$. It follows that $\varphi(0, 0, 0) = 0$, and in new coordinates, $\Delta(\xi, \eta, \zeta) = \frac{\xi^3(\sin\varphi(\xi,\eta,\zeta)+\cos\varphi(\xi,\eta,\zeta))}{1-\zeta^2} + \xi\eta + \eta\zeta$. Finally, a change of coordinates $x = \xi + \zeta$, $y = \eta$, $z = \xi\left(\frac{\sin\varphi(\xi,\eta,\zeta)+\cos\varphi(\xi,\eta,\zeta)}{1-\zeta^2}\right)^{1/3}$ provides the desired normal form $f(x, y, z) = z^3 + xy$.

Motion Planning Algorithm

We begin with any admissible control function $u_0(\cdot)$ of (1). If $R_T = \varphi_{R_0,T}(u_0(\cdot)) = R_d$, we are done. Otherwise, we choose a smooth curve $u_\theta(\cdot)$ of control functions, starting at $\theta = 0$ from $u_0(\cdot)$, and compute a logarithmic error $e(\theta) = \log(R_T(\theta)R_d^T) = [l_T(\theta)]$, where $R_t(\theta) = \phi_{R_0,t}(u_\theta(\cdot))$. We request that along the control curve the error decreases exponentially, i.e.

$$\frac{de(\theta)}{d\theta} = -\gamma e(\theta), \tag{4}$$

for a positive decay rate γ. It follows that the identity (4) is tantamount to

$$\frac{dl_T(\theta)}{d\theta} = -\gamma l_T(\theta). \tag{5}$$

Letting $S_t(\theta) = \frac{\partial R_t(\theta)}{\partial\theta}R_t^T(\theta) = [s_t(\theta)]$, we get

$$S_T(\theta) = \frac{dR_T(\theta)}{d\theta}R_T^T(\theta) = \frac{d\left(R_T(\theta)R_d^T\right)}{d\theta}\left(R_T(\theta)R_d^T\right)^T = \frac{d\exp([l_T(\theta)])}{d\theta}\exp(-[l_T(\theta)]).$$

Now, the Hausdorff formula [7] yields

$$\frac{d\exp([l_T(\theta)])}{d\theta}\exp(-[l_T(\theta)]) = \left[P(\theta)\frac{dl_T(\theta)}{d\theta}\right], \tag{6}$$

what means that

$$s_T(\theta) = P(\theta)\frac{dl_T(\theta)}{d\theta}, \tag{7}$$

where

$$P(\theta) = \exp([l_T(\theta)]) = \mathbb{I}_3 + \frac{1-\cos\alpha}{\alpha^2}[l_T(\theta)] + \frac{\alpha-\sin\alpha}{\alpha^3}[l_T(\theta)]^2,$$

and $\alpha(\theta)$ denotes the rotation angle of $R_T(\theta)R_d^T$, given by $\cos\alpha(\theta) = \frac{1}{2}(tr(R_T(\theta)\cdot R_d^T) - 1)$. Referring to the system (1), we compute

$$\begin{aligned}\frac{\partial}{\partial t}\frac{\partial R_t(\theta)}{\partial\theta} &= \frac{\partial}{\partial\theta}\frac{\partial R_t(\theta)}{\partial t}\\ &= \left[\left(\frac{\partial L_t(\theta)}{\partial\theta}\right)u_\theta(t) + L_t(\theta)\frac{\partial u_\theta(t)}{\partial\theta}\right]R_t(\theta) + [L_t(\theta)u_\theta(t)]\frac{\partial R_t(\theta)}{\partial\theta},\end{aligned}$$

and, after some mathematical developments, conclude that

$$\frac{\partial s_t(\theta)}{\partial t} = \frac{\partial L_t(\theta)}{\partial\theta}u_\theta(t) + L_t(\theta)\frac{\partial u_\theta(t)}{\partial\theta} + [L_t(\theta)u_\theta(t)]\,s_t(\theta). \tag{8}$$

A little more computation shows that $\frac{\partial L_t(\theta)}{\partial\theta}u_\theta(t) = Q_t(\theta)s_t(\theta)$, for a certain matrix $Q_t(\theta)$, that will be displayed below. Thus, (8) can be represented as a linear, time dependent control system

$$\frac{\partial s_t(\theta)}{\partial t} = A_\theta(t)s_t(\theta) + B_\theta(t)\frac{\partial u_\theta(t)}{\partial\theta}, \tag{9}$$

where $A_\theta(t) = Q_t(\theta) + [L_t(\theta)u_\theta(t)]$, $B_\theta(t) = L_t(\theta)$, and $\frac{\partial u_\theta(t)}{\partial\theta}$ playing the role of control. After the integration of (9) with initial condition $s_0(\theta) = 0$, we deduce from (7) a basic motion planning integral equation

$$J_{R_0,T}(u_\theta(\cdot))\frac{\partial u_\theta(t)}{\partial\theta} = \int_0^T \Phi_\theta(T,t)B_\theta(t)\frac{\partial u_\theta(t)}{\partial\theta}dt = -\gamma l_T(\theta), \tag{10}$$

where the transition $\Phi_\theta(t,s)$ of (9) satisfies the evolution equation $\frac{\partial\Phi_\theta(t,s)}{\partial t} = A_\theta(t)\Phi_\theta(t,s)$ along with $\Phi_\theta(t,t) = \mathbb{I}_3$. Specifically, $\frac{\partial\Phi_\theta(T,t)}{\partial t} = -\Phi_\theta(T,t)A_\theta(t)$.

The operator $J_{R_0,T}(u_\theta(\cdot))$ appearing in (10) can be regarded as a Jacobian operator associated with the motion planning problem in the system (1). Using the Moore-Penrose inverse of this Jacobian, $J^{\#}_{R_0,T}(u_\theta(\cdot))$, we derive the motion planning algorithm

$$\frac{\partial u_\theta(t)}{\partial \theta} = -\gamma \left(J^{\#}_{R_0,T}\left(u_\theta(\cdot)\right)\right)(t) l_T(\theta) = -\gamma B^T_\theta(t) \Phi^T_\theta(T,t) M^{-1}_\theta(T) l_T(\theta). \quad (11)$$

The matrix $M(T)_\theta = \int_0^T \Phi_\theta(T,t) B_\theta(t) B^T_\theta(t) \Phi^T_\theta(T,t) dt$ that can be called a mobility matrix of the system (1) is found as a solution of the Lyapunov matrix differential equation $\frac{\partial M_\theta(t)}{\partial t} = B_\theta(t) B^T_\theta(t) + A_\theta(t) M_\theta(t) + M_\theta(t) A^T_\theta(t), \quad M_\theta(0) = 0$. Finally, taking into account the identity (2), the matrix $Q_t(\theta)$ has been computed as

$$Q_t(\theta) = \frac{2}{\Delta^2} \begin{bmatrix} -r_{12}(m_1 e_2^T - m_2 e_3^T)[r_1] + r_{21}(m_1 e_1^T + m_2 e_3^T)[r_2] \\ r_{12}(m_1(e_2^T - m_2 e_3^T)[r_1] - r_{21}(m_1 e_1^T + m_2 e_3^T)[r_2] \\ -r_{32}(m_1(e_2^T - m_2 e_3^T)[r_1] - r_{31}(m_1 e_1^T + m_2 e_3^T)[r_2] \end{bmatrix}. \quad (12)$$

Here above $R_t(\theta) = [r_{ij}]$, $m_1 = r_{31} u_{\theta 1} - r_{32} u_{\theta 2}$, $m_2 = r_{21} u_{\theta 1} + r_{12} u_{\theta 2}$, r_1, r_2, r_3 denote columns of $R_t(\theta)$, e_1, e_2, e_3 are unit vectors in $\mathbb{R}^3$, and $\Delta = r_{12} r_{31} + r_{21} r_{32}$.

References

1. Di Gregorio, R.: Position analysis and path planning of the S-(nS) PU-S PU and S-(nS)PU-2S PU underactuated wrists. ASME J. Mech. Robotics 4(2) (2012)
2. Grosch, J., Thomas, F.: A bilinear formulation for the motion planning of non-holonomic parallel orienting platforms. In: Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan, pp. 953–958 (2013)
3. Grosch, P., Di Gregorio, R., Thomas, F.: Generation of under-actuated manipulators with non-holonomic joints from ordinary manipulators. ASME J. Mech. Robotics 2(1) (2010)
4. Jakubiak, J., Tchoń, K., Magiera, W.: Motion planning in velocity affine mechanical systems. Int. J. Control **83**(9), 1965–1974 (2010)
5. Nakamura, Y., Chung, W., Sordalen, O.J.: Design and control of the nonholonomic manipulator. IEEE Trans. Robot. Automat. **1**(17), 48–59 (2001)
6. Ratajczak, A., Tchoń, K.: Parametric and nonparametric jacobian motion planning for non-holonomic robotic systems. J. Intell. Rob. Syst. (2013)
7. Selig, J.M.: Geometric Fundamentals of Robotics. Springer Science+Business Media B. V, Berlin (2005)
8. Tchoń, K., Jakubiak, J., Grosch, P., Thomas, F.: Motion planning for parallel robots with non-holonomic joints. In: Latest Advances in Robot Kinematics, pp. 115–122. Springer (2012)

Nonsingular Change of Assembly Mode Without any Cusp

Michel Coste, Damien Chablat and Philippe Wenger

Abstract This chapter shows for the first time a parallel manipulator that can execute nonsingular changes of assembly modes while its joint space is free of cusp points and cuspidal edges. The manipulator at hand has two degrees of freedom and is derived from a 3-RPR manipulator; the shape of its joint space is a thickening of a figure-eight curve. A few explanations concerning the relationship between cusps and alpha curves are given.

Keywords Parallel robots · Singularities

1 Introduction

The nonsingular change of assembly mode in parallel manipulators was first observed by Innocenti and Parenti-Castelli [1]. The possibility to execute such a motion is most frequently associated with the presence of cusps, and the nonsingular change of assembly mode is realized by turning around a cusp point, or a cuspidal edge of the singularity surface (see for instance [2–5]). It has also been reported [6, 7] that nonsingular change of assembly modes can be realized by following an "alpha curve" (i.e. a fold curve intersecting itself transversally) but, in the examples shown, such an alpha curve is always associated with the presence of cusps. Figure 1 shows such an example. The figure depicts a slice of the joint space for a 3-RPR manipulator,

M. Coste (✉)
IRMAR, Université de Rennes 1—CNRS, Rennes, France
e-mail: michel.coste@univ-rennes1.fr

D. Chablat · P. Wenger
IRCCyN, CNRS—École Centrale de Nantes, Nantes, France
e-mail: damien.chablat@irccyn.ec-nantes.fr

P. Wenger
e-mail: philippe.wenger@irccyn.ec-nantes.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_12,

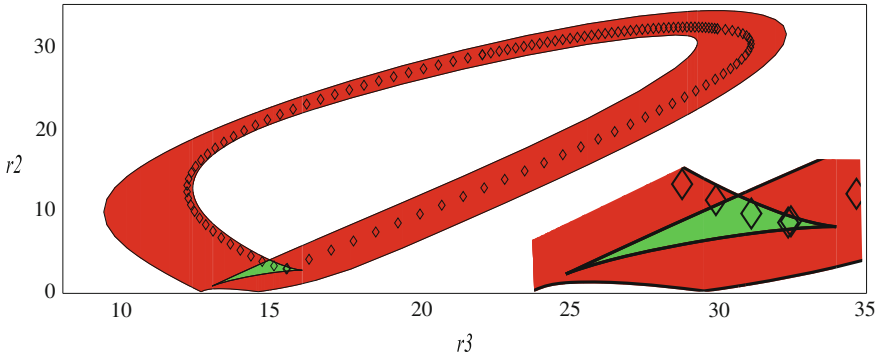


Fig. 1 Change of assembly mode looping around an alpha curve, with a zoom on the zone with four solutions to the direct kinematic problem

along with a nonsingular assembly mode changing trajectory that goes around the alpha-curve. The zone with four (resp. two) solutions to the direct kinematic problem is colored in green (resp. red). Going from the green zone to the green zone following the trajectory, one passes from one solution to the other solution in the same aspect. Since there are two cusps, note that a nonsingular change of assembly mode may also be executed by encircling one of them.

In the present chapter we describe a planar parallel manipulator with two degrees of freedom which has absolutely no cusp, but nevertheless allows nonsingular changes of assembly modes. This manipulator is derived from a 3-RPR manipulator with one prismatic articulation blocked, coupled with a mechanism for reversing an angle. We describe this manipulator in the next section. The idea behind the special design of this manipulator is to obtain a joint space which has the shape of a thickened figure-eight curve. This is first established with an asymptotic simplification of the inverse kinematic mapping in Sect. 3. Section 4 contains the actual example of a nonsingular change of assembly mode showing no cusp at all. Some remarks on stable singularities are given in Sect. 5; in particular, we comment about the relationship between cusps and alpha curves.

2 The Non Cuspidal Manipulator

The manipulator we describe here derives from a planar 3-RPR. It has a triangular moving platform $B_1B_2B_3$ which is linked to the base $A_1A_2A_3$ via three legs A_iB_i, with passive rotoidal joints at A_i and B_i (for $i = 1, 2, 3$). The length r_1 of the leg A_1B_1 is fixed. The lengths r_2 and r_3 of the other two legs A_2B_2 and A_3B_3 are controlled by actuated prismatic joints. The vertices A_1 and A_2 of the base are fixed. The length A_1A_3 is fixed. The peculiarity of this manipulator, which makes it different from a

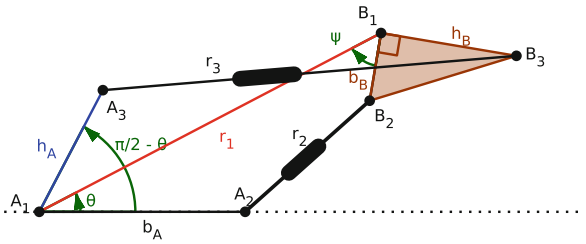


Fig. 2 The manipulator

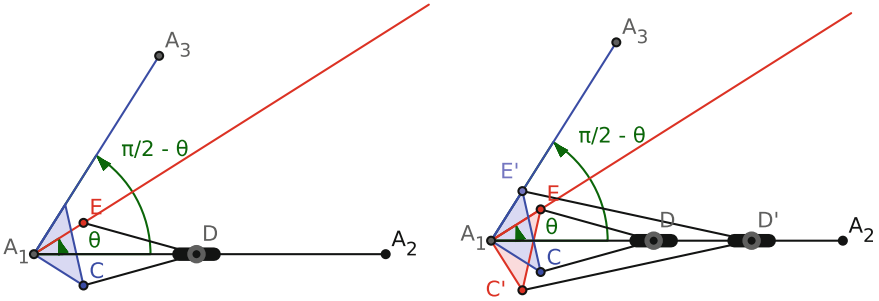


Fig. 3 The 4-bar kite and the two coupled kites

3-RPR with one blocked prismatic joint, is the fact that the angle $\widehat{A_2A_1A_3}$ varies. Precisely, if we denote by θ the angle $\widehat{A_2A_1B_1}$, then $\widehat{A_2A_1A_3} = \dfrac{\pi}{2} - \theta$ (see Fig. 2).

Let us complete the description of the manipulator. The platform $B_1B_2B_3$ is a right-angled triangle, with right angle at B_1. Its dimensions are given by $b_B = B_1B_2$ and $h_B = B_1B_3$. The dimensions of the base are given by $b_A = A_1A_2$ and $h_A = A_1A_3$. The position of the manipulator is completely described by the angles θ and $\psi = \widehat{B_2B_1A_1}$.

We explain now how one can constrain the angle $\widehat{A_2A_1A_3}$ to be equal to $\dfrac{\pi}{2} - \theta$. Consider a 4-bar kite A_1CDE with $A_1C = A_1E$ and $CD = ED$. The rotoidal joint A_1 is fixed and D is constrained to glide on A_1A_2. The rotoidal joint E is on the leg A_1B_1. Of course, we have $\widehat{A_2A_1C} = -\theta$. We constrain A_1A_3 to be orthogonal to A_1C so that $\widehat{A_2A_1A_3} = \dfrac{\pi}{2} - \theta$ (see Fig. 3, left).

When the 4-bar kite is flat (this happens when $\theta = 0$ or π), it can change its operating mode by moving in such a way that C and E remain coincident. In order to rule out this possibility, we can couple the first kite A_1CDE with a second one $A_1C'D'E'$, as shown on Fig. 3, right.

We could have used other mechanisms in order to reverse the angle θ: for instance Kempe's reverser consisting of two contra-parallelograms (see [8] p. 270), or belts, or gears.

3 The Inverse Kinematic Mapping and Its Asymptotic Simplification

The constraint equations of the manipulator can be written as follows. Recall that r_i is the length of the leg A_iB_i. The actuated joint variables are r_2 and r_3, while r_1 is fixed. The position of the manipulator is determined by the angles θ and ψ. We have:

$$\begin{aligned} r_2^2 &= (r_1 - b_A\cos\theta - b_B\cos\psi)^2 + (b_A\sin\theta + b_B\sin\psi)^2, \\ r_3^2 &= (r_1 - h_A\sin 2\theta - h_B\sin\psi)^2 + (h_A\cos 2\theta + h_B\cos\psi)^2. \end{aligned} \tag{1}$$

Equation (1) describe the inverse kinematic mapping from the workspace, which is a torus parametrized by the angles θ and ψ, to the actuated joint space, which is the positive quadrant parametrized by the lengths r_2 and r_3.

We shall now consider the asymptotic version of the inverse kinematic mapping as r_1 tends to infinity, as was done in [9] for the usual 3-RPR manipulator. In order to do this, we replace r_2 and r_3 with $r_2 - r_1$ and $r_3 - r_2$ and take their limit $\delta_i = \lim_{r_1\to\infty}(r_i - r_1)$ for $i = 2, 3$. We obtain:

$$\begin{aligned} \delta_2 &= -b_A\cos\theta - b_B\cos\psi, \\ \delta_3 &= -h_A\sin 2\theta - h_B\sin\psi. \end{aligned} \tag{2}$$

Equation (2) describe the asymptotic simplification of the inverse kinematic mapping, from the torus (θ, ψ) to the plane (δ_2, δ_3). The singularities of the asymptotic inverse kinematic mapping are easily understood. Remark that $(b_A\cos\theta,\ h_A\sin 2\theta)$ is a parametrization of a figure-eight curve, or lemniscate of Gerono (a particular case of a Lissajous curve), while $(b_B\cos\psi,\ h_B\sin\psi)$ is, of course, the parametrization of an ellipse. So the curve of critical values of the asymptotic inverse kinematic mapping is the envelope of a family of translated ellipses with centres on a figure-eight curve. If the ellipse is small compared to the figure-eight curve, we obtain just a thickening of the figure-eight curve.

Specifically, take $b_A = 10$, $h_A = 5$, $b_B = 1$, $h_B = 2$. The Jacobian determinant of the asymptotic inverse kinematic mapping is then

$$J = -20\,\sin\theta\,\cos\psi + 10\,\cos 2\theta\,\sin\psi. \tag{3}$$

Solving $J = 0$ for ψ, we obtain that the critical points of the asymptotic inverse kinematic mapping are those such that

$$\cos\psi = \frac{\pm\cos 2\theta}{\sqrt{1 + 4\,\sin^4\theta}}, \quad \sin\psi = \frac{\pm 2\,\sin\theta}{\sqrt{1 + 4\,\sin^4\theta}}. \tag{4}$$

The two branches of the curve of critical points in the (θ, ψ) torus are represented in thick black line in Fig. 4, left. These two branches delimit two aspects.

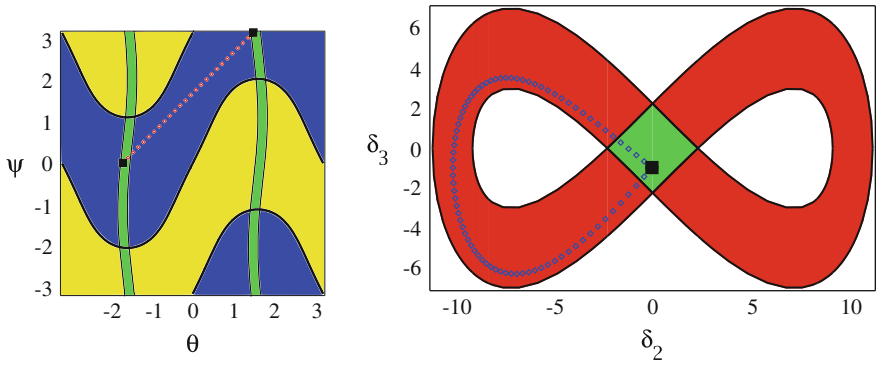


Fig. 4 Work space: the (θ, ψ) torus (*left*) and joint space: the (δ_2, δ_3) plane (*right*) for the asymptotic model

Carrying these values in the Eq. (2) for the asymptotic inverse kinematic mapping, we get the two branches of the curve of critical values in the (δ_2, δ_3) plane. They are represented in Fig. 4, right. These curves delimit zones where the direct kinematic problem has two solutions (red zone), and four solutions (green zone).

There is no cusp, but a nonsingular change of assembly mode is possible since there are two aspects and a zone where the direct kinematic problem has four solutions. A trajectory for such a nonsingular change of assembly mode is represented in Fig. 4. The nonsingular change of assembly mode can be interpreted using a comparison with a figure-eight race track where the crossroads is realized with a bridge, i.e. there are two levels at the crossroads. Each level is associated with an assembly mode. The track represents an aspect (the joint space is made of two superimposed similar aspects). Starting from the crossroads on the bridge (level 1), the car can drive until it reaches the crossroads at the same horizontal position but under the bridge (level 0).

The characteristic curves have been plotted in thin black line in Fig. 4, left. These curves, together with the critical curves, define the uniqueness domains: any two points in the same uniqueness domain of the (θ, ψ) torus have different images in the (δ_1, δ_2) plane (joint space). For a complete definition of the characteristic surfaces and uniqueness domains, see [10]. Any point in the green region of the joint space (Fig. 4, right) is the image of four points in the workspace (four assembly modes). These four points are distributed in the four green regions shown in the workspace (Fig. 4, left), two in each aspect.

The preceding analysis concerns, of course, only an asymptotic case. However, since the asymptotic inverse kinematic mapping has only stable singularities (folds and transverse intersection of folds, see Sect. 5) which are not altered by small perturbations, the conclusion obtained will remain valid for all sufficiently large values of r_1. We shall check this in the next section.

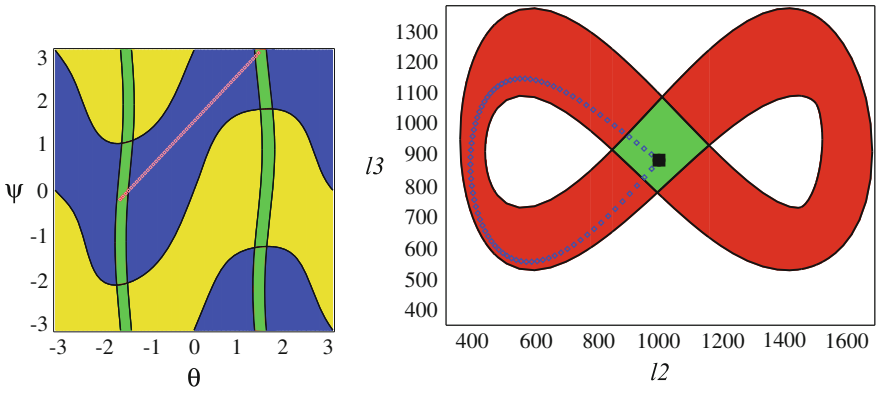


Fig. 5 Workspace and joint space for $r_1 = 30$, with a nonsingular change of assembly mode

4 An Actual, Non Asymptotic Example

The asymptotic analysis of the preceding section corresponds to $r_1 = \infty$. We show here that the conclusion remains valid for $r_1 = 30$ (keeping the same b_A, h_A, b_B, h_B). We have now to deal with Eq. (1) which are more complicated than their asymptotic simplification. The computation is done using the SIROPA library. In order to simplify calculations, we use coordinates $l_2 = r_2^2,\ l_3 = r_3^2$ for the actuated joint space, which is an unessential change.

We can see on Fig. 5 that the picture of the singularities is the same as the one for the asymptotic simplification (Fig. 4). In particular, there is no cusp. We have also represented a nonsingular trajectory between two configurations corresponding to the same values $l_2 = 1000, l_3 = 880$ for the actuated joints.

5 Stable Singularities and Nonsingular Change of Assembly Modes

The only stable singularities of a mapping from a surface to a surface (see [11]) are the folds (codimension one singularity, giving a fold curve), the cusps and the transverse intersections of fold curves (codimension two singularities, i.e. isolated points). Any other singularity will be decomposed into a combination of these stable singularities by a small perturbation, whereas a stable singularity is persistent under small perturbations.

Figure 6 represents these three stable singularities for a critical value P. Generically, the picture of parallel singularities in a two-dimensional surface around a critical value of the inverse kinematic mapping in the joint space will fit into one of these three cases. The direct kinematic problem has a double solution for a fold, a triple solution for a cusp and two double solutions for a transverse intersection of fold curves.

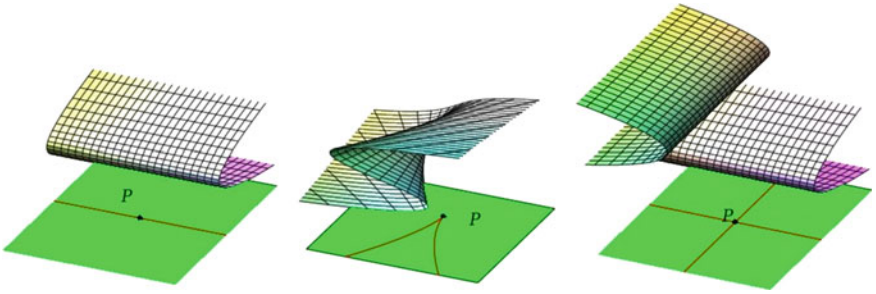


Fig. 6 Stable singularities: fold, cusp and transverse intersection of fold curves

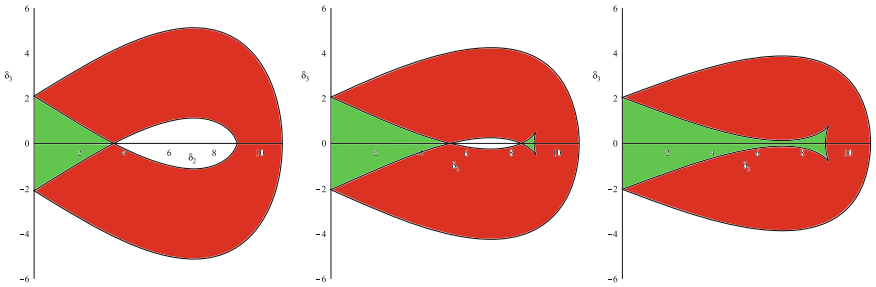


Fig. 7 Degeneration of an alpha curve

The only stable singularity which allows *local* nonsingular change of assembly mode is the cusp. By "local", we mean that the change of assembly mode is performed by following a loop which can be shrunk to become arbitrarily small. This is certainly the case for a loop encircling a cusp, but not for a loop following an "alpha curve", i.e. a fold curve intersecting itself transversally.

In examples, an alpha curve is frequently associated with cusps because a transversal intersection of folds often appears after a "swallowtail bifurcation" (see [12] p. 34), together with a pair of cusps. One might think that a cusp is obtained by shrinking the loop of an alpha curve to a point. This is not the case, since a cusp is a stable singularity which cannot be obtained by degenerating another singularity. If the loop of an alpha curve is shrunk to a point, then this point corresponds to a quadruple, not triple, solution to the direct kinematic problem.

In order to illustrate an example of degeneration of an alpha curve, and also the swallowtail bifurcation, we show the transformation of the (δ_2, δ_3) joint space of Sect. 3, as we decrease h_A keeping the other dimensions fixed. Only a half of the joint space is shown in Fig. 7, since it is symmetric.

Figure 7 shows that, at first, two cusps and a transverse intersection of fold curves appear as a result of a swallowtail bifurcation. Then, the two transverse intersections of fold curves disappear through a tangent crossing of these curves.

6 Conclusion

While it is clear that the existence of cusps or cuspidal edges is a sufficient condition for allowing a nonsingular change of assembly mode, the question of whether this condition is necessary or not remained to be fixed because previous examples of nonsingular assembly mode changing manipulators always exhibited cusps, even in the presence of an alpha-curve. This chapter showed that the aforementioned condition is indeed not necessary, through the presentation of a 2-DOF parallel manipulator that can execute a nonsingular change of assembly mode while its joint space is free of cusps. This manipulator is derived from a 3-RPR and its joint space is a thickening of a figure-eight curve. The kinematic properties of this manipulator were first established using the asymptotic simplification of its inverse kinematic mapping.

It was recalled that a cusp is a necessary and sufficient condition for *local* nonsingular change of assembly mode in generic situations; it was also explained that a cusp cannot be interpreted as a degeneration of an alpha-curve.

References

1. Innocenti, C., Parenti-Castelli, V.: Singularity-free evolution from one configuration to another in serial and fully-parallel manipulators? J. Mech. Des. **120**, 73–79 (1998)
2. McAree, P.R., Daniel, R.W.: An explanation of never-special assembly changing motions for 3–3 parallel manipulators. Int. J. Robot. Res. **18**(6), 556–574 (1999)
3. Zein, M., Wenger, P., Chablat, D.: Singular curves in the joint space and cusp points of 3-RPR parallel manipulators. Robotica **25**(6), 717–724 (2007)
4. Caro, S., Wenger, P., Chablat, D.: Non-singular assembly mode changing trajectories of a 6-DOF parallel robot. In: Proceedings of the ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE 2012, pp. 1245–1254, (2012)
5. Husty, M., Schadlbauer, J., Caro, S., Wenger, P.: The 3-RPS manipulator can have non-singular assembly-mode changes. Comput. Kinematics Mech. Mach. Sci. **15**, 339–348 (2014)
6. Bamberger, H., Wolf, A., Shoham, M.: Assembly mode changing in parallel mechanisms. IEEE Trans. Robot. **24**(4), 765–772 (2008)
7. Macho, E., Altuzarra, O., Pinto, C., Hernandez, A.: Transitions between multiple solutions of the direct kinematic problem. In: Advances in Robot Kinematics: Analysis and Design, pp 301–310. Springer (2008)
8. Koenigs, G.: Leçons de cinématique. Hermann, Paris (1897)
9. Coste, M.: Asymptotic singularities of planar parallel 3-R PR manipulators. In: Latest Advances in Robot Kinematics, pp 35–42. Springer (2012)
10. Wenger, P., Chablat, D.: Uniqueness domains in the workspace of parallel manipulators. In: I.F.A.C-Symposium on Robot, Control (SYROCO'97), pp 431–436 (1997)
11. Whitney, H.: On singularities of mappings of Euclidean spaces, I. Mapping of the plane into the plane. Ann. Mathe. **62**(3), 374–410 (1955)
12. Arnol'd, V.I.: Catastrophe Theory. Springer, Berlin (1992)

The Influence of Discrete-Time Control on the Kinematico-Static Behavior of Cable-Driven Parallel Robot with Elastic Cables

Jean-Pierre Merlet

Abstract Robots are controlled with a discrete-time controller that includes a high-level loop for motion control and a faster internal loop that controls the actuators. We intend to simulate the behavior of the whole chain for a cable-driven parallel robot (CDPR) with linear elastic cables and we will show that such a simulation cannot be performed using classical simulation tools. We exhibit a simulation algorithm which computes exactly the pose and cable tensions on a given trajectory. As an example we consider a redundantly actuated robot with 8 cables. We show that the discrete-time control has a moderate influence on the accuracy of the positioning but a very large influence on the cable tensions.

Keywords Cable-driven parallel robot · Elastic cable · Discrete-time control

1 Introduction

The study of CDPR has started about 30 years ago with the pioneering work of Albus [2] and Landsberger [13] but there has been recently a renewed interest in such a robot, both from a theoretical and application viewpoint. For example kinematics analysis of CDPR is much more complex than the one of parallel robot with rigid legs as static equilibrium has to be taken into account [5, 11, 21] and is still an open issue especially as not all cables of a robot with m cables may be under tension [1, 3, 6, 8, 16] and that only stable solutions have to be determined [7]. This analysis is even more complex if we consider that the cables may be elastic and/or deformable [9, 10, 12, 18].

J.-P. Merlet (✉)
INRIA Sophia-Antipolis, Sophia-Antipolis, France
e-mail: Jean-Pierre.Merlet@inria.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_13,

Numerous applications of CDPRs have been mentioned e.g. large scale maintenance studied in the European project Cablebot [17], rescue robot [15, 19] and transfer robot for elderly people [14] to name a few.

However a problem has never been addressed when simulating CDPRs: the influence of the use of discrete-time control on the kinematic and static behavior of the robot. Only Borgstrom [4] has presented a motion planning algorithm for CDPRs that takes into account the discrete-time nature of the controller. The purpose of this paper is to study the controller influence on the kinematic and static behavior of a CDPR with linear elastic cables when performing a given trajectory.

2 Discrete-Time Control

A robot controller is basically constituted of two discrete-time control loops:

- a high-level loop with a sampling frequency Δt_1: at time $k\Delta t_1$, k being an integer, this loop get sensory information from the robot, process it and send a new command for the actuators at time $(k+1)\Delta t_1$
- an inner loop with a sampling frequency $\Delta t_2 < \Delta t_1$: at time $l\Delta t_2$, l being an integer, this loop get sensory information from the actuators and process it for sending new voltages or currents to the actuators at time $(l+1)\Delta t_2$. For the sake of simplicity we will assume that Δt_1 is a multiple of Δt_2.

A consequence of this scheme is that during the time interval $[l\Delta t_2, (l+1)\Delta t_2]$ the actuators are submitted to a constant voltage/current V and the CDPR state evolves according to only the actuator state. We will assume that a time-model of the actuator is available i.e. the output θ of the motor at any time T in the range $[t, t+\Delta t_2]$ may be obtained as $\theta(T) = H(T, \theta(t), V)$.

3 Cable Configurations and Kinematico-Static Equations

We consider a CDPR with m elastic cables, numbered from 1 to m, whose extremities are located on the robot base at point A_i and attached to the platform at point B_i. If ρ is the cable real length and l_0 its length at rest, then the tension τ in the cable is $\tau = K(\rho - l_0)$, where K is the stiffness constant, provided that l_0 is larger than the distance $||\mathbf{A_iB_i}||$, otherwise the cable is slack and $\tau = 0$. As a cable may become slack we introduce the concept of *cable configuration* (CC): a cable configuration M_i at a pose is a set of $i \leq m$ integers which are the numbers of the cables that are currently under tension, all other cables being slack. Note that at a given pose several CC may be possible, the current one depending on the history of the system.

We may now investigate the equations that are involved in the forward kinematics (FK) problem for a given CC M_j. The unknowns are the 6 parameters $\mathbf{X}$ that describe the pose of the platform and the i tensions τ or, equivalently, the i cable lengths ρ,

for a total of $6 + i$ unknowns. The equations are the i inverse kinematic equations and the 6 equation from the mechanical equilibrium:

$$\rho = G(\mathbf{X}) \qquad \mathbf{F} = \mathbf{J}^{-\mathbf{T}}\tau \tag{1}$$

where $\mathbf{F}$ is the external force applied on the platform. In this paper we will assume that the only external force is gravity that is applied at the center of mass C of the platform. The matrix $\mathbf{J}^{-\mathbf{T}}$ is the $6 \times i$ transpose of the inverse kinematic jacobian. The j-th row J_j of $\mathbf{J}^{-\mathbf{T}}$ is given by

$$J_j = ((\frac{\mathbf{A_jB_j}}{\rho_j} \quad \mathbf{CB_j} \times \frac{\mathbf{A_jB_j}}{\rho_j}))$$

Hence we end up with a square system of $6 + i$ equations that may be written as

$$\mathbf{F}(\mathbf{X}, \rho, l_0) = 0. \tag{2}$$

4 Kinematico-Static Simulation on a Trajectory

We are interested in determining the kinematic and static behavior of a CDPR under a discrete-time controller when it has to move along a given trajectory. We will assume that when the CDPR starts its task the current CC M_l is known. We will also assume that the pose at the start point is known.

4.1 Finding a Certified End-Pose

We consider a time interval $\mathscr{T} = [l\Delta t_2, l\Delta t_2 + \Delta t]$, with $\Delta t \leq \Delta t_2$, and we assume that at time t the CC M_j is known, together with the pose $\mathbf{X_t}$. Our objective is to determine what is the pose at time $l\Delta t_2 + \Delta t$, under the assumption that the CC does not change on the whole time interval. If such result can be obtained the time interval $\mathscr{T}$ will be called *valid*. For checking the validity of $\mathscr{T}$ we consider the system of Eq. (1) at time T, i.e. for fixed values of the l_0. This system may admit several solutions i.e. the pose $\mathbf{X}$ may lie on different *kinematic branches* $\mathscr{S}_i$ and our objectives are to show (1) that for any time T the pose lies on the same branch $\mathscr{S}_t$ than $\mathbf{X_t}$ and (2) to calculate the pose at time $l\Delta t_2 + \Delta t$. Assume that we are able to show that for any time T system (1) admits a single solution in a ball centered at $\mathbf{X_t}$: this implies that during the time interval the kinematic branch on which lies $\mathbf{X_t}$ does not cross any other branch and that the pose always lies on the branch $\mathscr{S}_t$, hence fulfilling 1). For showing the unicity of the solution in a ball centered at $\mathbf{X_t}$ we will use *Kantorovitch theorem* [20], that is presented now. Let an arbitrary system of n

equations in n unknowns $\mathbf{F} = \{F_i(x_1, \ldots, x_n) = 0, i \in [1, n]\}$ and $\mathbf{x_0}$ be a point and U a ball centered at $\mathbf{x_0}$ with radius B_0. Assume that $\mathbf{x_0}$ is such that:

1. the Jacobian matrix $\mathbf{J_0}$ of the system has an inverse Γ_0 at $\mathbf{x_0}$ such that $||\Gamma_0|| \leq A_0$
2. $||\Gamma_0 \mathbf{F}(\mathbf{x_0})|| \leq 2B_0$
3. $\sum_{k=1}^{n} |\frac{\partial^2 F_i(\mathbf{x})}{\partial x_j \partial x_k}| \leq C$ for $i, j = 1, \ldots, n$ and $\mathbf{x} \in U$
4. the constants A_0, B_0, C satisfy $2nA_0B_0C \leq 1$ (A)

Then there is an unique solution of $\mathbf{F} = 0$ in U and Newton iterative scheme used with $\mathbf{x_0}$ as estimate of the solution will converge toward this solution.

In our case however as we consider any time in the time interval, Eq. (1) is not a single system but a family of systems because l_0 vary over time. We may however assume that the time model of the actuator allow us to determine an interval $\mathscr{I}^i = [l^i_{min}, l^i_{max}]$ such that for all cables we have $l^i_0 \in \mathscr{I}^i$. Note that the width of $\mathscr{I}^i$ will decrease with Δt. Assuming that the reader is familiar with interval analysis (IA) we may now apply Kantorovitch theorem to the system (2) using $\mathbf{X_t}$ as $\mathbf{x_0}$ with the following modifications:

- $\mathbf{F}(\mathbf{x_0})$ has now an interval value
- the matrix $\mathbf{J_0}$ is an interval matrix. Classical method allows to obtain its inverse but may fail if the width of the intervals in $\mathbf{J_0}$ is too large
- the Hessian matrix appearing in item 3 of the theorem is also an interval matrix but its norm can be calculated with IA methods

We start by setting $\Delta t = \Delta t_2$. If the interval matrix $\mathbf{J_0}$ cannot be inverted or condition (A) of the theorem is not satisfied, then we set $\Delta t = \Delta t/2$, update the ranges $\mathscr{I}^i$ and starts again until a valid Δt is determined. This approach may fail only in two cases: (a) system (2) is close to a singularity (in which case we cannot predict the behavior of the robot) or (b) in case of insufficient computer accuracy (this issue will be addressed in a later section). If a valid Δt is found we are able to calculate the pose at time $l\Delta t_2 + \Delta t$ unless a CC change occurs in the time interval.

4.2 Finding Cable Configuration Changes

Assume that a valid interval $[l\Delta t_2, l\Delta t_2 + \Delta t]$ has been determined in the previous step. If no CC change occur in this time interval, then we are able to calculate the pose at time $l\Delta t_2 + \Delta t$. Necessary conditions for a CC change are

1. there is a time T in the time interval at which the tension of a cable $i \in M_j$ is exactly equal to 0, i.e. $\rho_i = l^i_0 = ||\mathbf{A_i B_i}||$.
2. there is a time T in the time interval at which the length l^i_0 a cable $i \notin M_j$ is such that $l^i_0 = ||\mathbf{A_i B_i}||$.

In both cases Eq. (2) for the CC M_j are still valid but we have an additional unknown, T, while one of the unknown, ρ_i, has now a known value. Hence this new version

of (2), denoted $\mathbf{F_{mod}}^i$, is still a square system. As we have to consider that all the m cable may possibly satisfy in turn $\rho_i = l_0^i$, we have therefore m system $\mathbf{F_{mod}}^i$. Note that we have bounds on all unknowns of the new system: $\mathbf{X}$ and the ρ_i have to lie in the ball provided by the Kantorovitvh theorem. Hence it is quite natural to use IA to determine **all** possible solutions of all m systems $\mathbf{F_{mod}}$. All the n solutions are then ordered by increasing value $T_1^{k_1}, T_2^{k_2}, \ldots, T_n^{k_n}$ where the superscript k_i denotes the cable number for which $\rho_{k_i} = l_0^{k_i}$. In the time interval $[l\Delta t_2, l\Delta t_2 + T_1^{k_1}]$ we are sure that the platform lie on the kinematic branch $\mathscr{S}_t$. A possible CC change may occur at time $T_1^{k_1}$, where we have $\rho_{k_1} = l_0^{k_1}$, but is not certain. Indeed if $k_1 \in M_j$ the tension may decrease before $T_1^{k_1}$, cancel at $T_1^{k_1}$ but may then increase. In the same manner if $k_1 \notin M_j$ the distance $||\mathbf{A_{k_1}B_{k_1}}||$ may increase before $T_1^{k_1}$, reach $l_0^{k_1}$ at $T_1^{k_1}$ but may then decrease so that cable k_1 remains slack. To determine if such case occurs we consider a new CC M_{j+1} obtained by adding k_1 to the CC M_j. We then apply the Kantorovitch theorem on the Eq. (2) valid for this new CC, using as $\mathbf{x_0}$ the pose obtained for the time $T_1^{k_1}$. We then calculate times $T_1', T_2', \ldots, T_u'$ at which a CC change may occur. We then solve the system obtained for time $(T_1^{k_1} + T_1')/2$. If at this time we have both $k_1 \in M_j$ and $||\mathbf{A_{k_1}B_{k_1}}|| < l_0^{k_1}$ or both $k_1 \notin M_j$ and $\rho_{k_1} > l_0^{k_1}$, then a CC change occurs at $T_1^{k_1}$, cable k_1 becoming slack in the first case and under tension in the second one. If this not the case we repeat the procedure for time $T_2^{k_2}$, …, $T_n^{k_n}$ until either a CC change occur or there is no CC change at $T_n^{k_n}$, which implies that at time $l\Delta t_2 + \Delta t$ the platform still lies on the branch $\mathscr{S}_t$.

4.3 Trajectory Checking

The two previous sections allow us to determine the pose, cable configuration and cable tensions at any time when the CDPR performs a trajectory. As soon as the high level loop has sent an order to the inner one we will determine the pose and tension during the time intervals $[k\Delta t_1 + l\Delta t_2, k\Delta t_1 + (l+1)\Delta t_2]$ until $k\Delta t_1 + (l+1)\Delta t_2 = (k+1)\Delta t_1$. We store the CDPR status at times $k\Delta t_1 + l\Delta t_2, k\Delta t_1 + (l+1)\Delta t_2$ and possible at intermediate times. Note that uncertainties may be taken into account: for example we may use in the simulation a different value of the cable stiffness K_i than the one used in the high level loop or we may introduce arbitrary random errors in the measurements of the l_0 that is used by the inner loop.

5 Implementation and Example

The previous algorithm has been implemented assuming a first order time model for the velocity of the actuator. Let V_c be the desired velocity of the actuator and $V(t_0)$ the known velocity at time t_0. Then the actuator velocity $V(t)$ at time $t \geq t_0$ is

$$V(t) = V_c + (V(t_0) - V_c)e^{-\frac{t-t_0}{U}}$$

where U is a known constant. The inner loop is a simple P controller that send to the actuator at time $(l+1)\Delta t_2$ a velocity order $V^l = V_c + k_p(V_c - V_m)$, where V_m is the measured velocity at time $l\Delta t_2$, V_c the velocity sent by the high level loop and k_p a constant gain.

The IA part of the algorithm has been implemented using our IA library `ALIAS`, that takes into account round-off errors. But even with this library we have encountered numerical problems, especially with the convergence of the Newton scheme. The satisfaction of Kantorovitch theorem requires that the pose computed for a given time T is accurate enough. Indeed a minimum condition for the theorem to provide a positive answer is that it is satisfied at time T. Roughly this means that the absolute value of the components of $\mathbf{F}$ at this time have a value less than $1/(2kA_0^2C_0)$, where k is the number of equations in $\mathbf{F}$. In some cases we have noticed that this value is very low and well below the accuracy of floating-point calculation. Hence the floating-point version of the Newton scheme oscillates around the solution without ever producing a value of $\mathbf{F}$ which is small enough. Fortunately we have a mean to solve this issue: `ALIAS` has a Maple interface that includes a multi-precision Newton scheme, allowing to calculate a solution with an arbitrary accuracy. Consequently when Kantorovitch theorem is satisfied but the floating-point Newton does not converge, then we use the Maple version.

As a complex example we consider the 8-cables large scale robot developed by LIRMM and Tecnalia as part of the ANR project Cogiro and the FP7 project CABLEBOT. This robot is a suspended CDPR whose dimensions have been given in several papers [10]. The platform is assumed to have a mass of 1/9.81 kg. We consider a planar circular trajectory centered at (0,0,2) with radius 1. The sampling times were fixed to $\Delta t_1 = 0.005$ s, $\Delta t_2 = 0.001$ s and the motor constant U to 0.1 s. The high level loop computes at time $k\Delta t_1$ what should be the pose at time $(k+2)\Delta t_1$ and calculates the l_0^c for this pose that minimize $\sum \tau_j^2$. The inner loop generates a velocity order for the actuators as $K_p(l_0^c - l_0^m)$ where K_p is a constant gain and l_0^m the l_0 measured at time $k\Delta t_1$. We have considered two simulation cases. In the first one there is no error on the measurement of the l_0 and the stiffness of the cables was set to 1,000 N/m (which correspond roughly to the stiffness of nylon). In the second case we add a random error on the l_0^m in the range $[-0.01, 0.1]$ (the average value of the l_0 on this trajectory is about 800), the high level loop assume a cable stiffness of 1,000 but the real cable stiffness was set to 1, 050, 900, 950, 1, 020, 1, 010, 1, 000, 1040, 980.

In the first case the maximal positioning error on the trajectory is 0.00002275 with a mean value of 0.36610^{-5}. In the second case the maximal error is 0.00575 with a mean value of 0.00104 (Fig. 1). Hence it may be seen that the uncertainties on the stiffness and length measurement has a relatively low influence on the positioning accuracy,

The situation is quite different for the tensions in the cables. Without uncertainties the maximal difference between the cable tensions and the optimal one over all cables is 0.000221 N with a mean value of 0.0001 N. With uncertainties the maximal

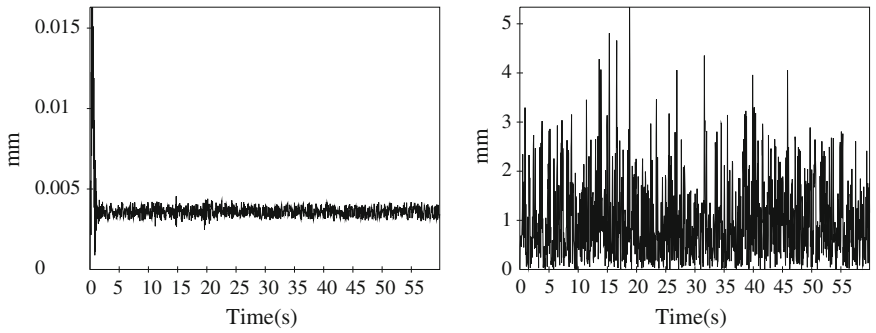


Fig. 1 Positioning error without and with uncertainties (mm)

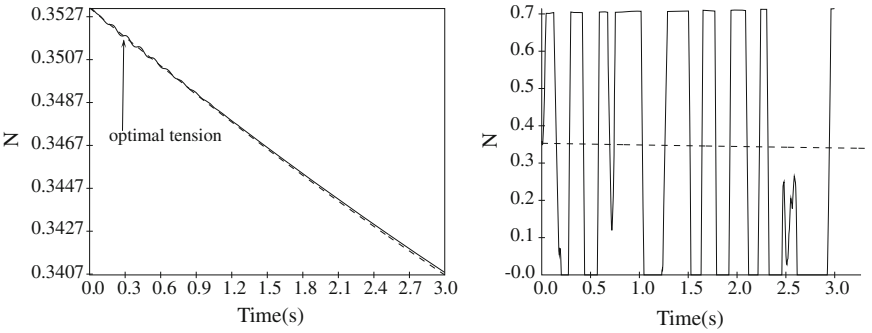


Fig. 2 Tension of cable 1 without uncertainty and with uncertainty (optimal tension is the *dashed line*)

difference is 0.4844 N with a mean value of 0.28097 N. In percentage of the optimal tension the maximal difference is 140.13 % and the mean value is 72.85 %. Figure 2 presents tension of cable 1 together with its optimal tension during the first 3 seconds of the trajectory. It may be seen that a perfect knowledge of the cable stiffness allows to follow accurately the optimal tension. But as soon that as the real stiffness differ by a small amount from the assumed one the cable tension oscillates between slack state and under tension.

This analysis confirms that the use of a discrete-time controller prohibits tension control in CDPRs.

6 Conclusion

To the best of our knowledge this paper has presented for the first time a simulation of CDPRs that takes into account the discrete-time nature of current controller. Implementing this simulation is a complex task because it involves solving the FK

but also because the necessary accuracy for obtaining this simulation may be lower than the one obtained with floating point calculation. Results on an example shows that positioning accuracy is not that much influenced by the controller but that on the other hand cable tensions are drastically influenced. Our next objective will be to take into account the dynamics of the robots in this simulation.

This research has received partial funding from the EC's Seventh Framework Program under grant agreement NMP2-SL-2011-285404 (CABLEBOT).

References

1. Abbasnejad, G., Carricato, M.: Real solutions of the direct geometrico-static problem of underconstrained cable-driven parallel robot with 3 cables: a numerical investigation. Meccanica 473(7), 1761Ã¢â?¬â??1773 (2012)
2. Albus, J., Bostelman, R., Dagalakis, N.: The NIST SPIDER, a robot crane. J. Res. Natl. Inst. Stand. Technol. **97**(3), 373–385 (1992)
3. Berti, A., Merlet, J.P., Carricato, M.: Solving the direct geometrico-static problem of the 3–3 cable-driven parallel robots by interval analysis: preliminary results. In: 1st International Conference on Cable-Driven Parallel Robots, pp. 251–268. Stuttgart 3–4 Sept 2012
4. Borgstrom, P., et al.: Discrete trajectory control algorithms for NIMS3D, an autonomous underconstrained three-dimensional cabled robot. In: IEEE International Conference on Intelligent Robots and Systems (IROS), pp. 253–240. San Diego, 22–26 Sept 2007
5. Bruckman, T., et al.: Parallel Manipulators, New Developments, Chap. Wire Robot Part I, Kinematics, Analysis and Design, pp. 109–132. ITECH (2008)
6. Carricato, M., Abbasnejad, G.: Direct geometrico-static analysis of under-constrained cable-driven parallel robots with 4 cables. In: 1st International Conference on Cable-Driven Parallel Robots, pp. 269–286. Stuttgart, 3–4 Sept 2012
7. Carricato, M., Merlet, J.P.: Stability analysis of underconstrained cable-driven parallel robots. IEEE Trans. Robotics **29**(1), 288–296 (2013)
8. Carricato, M., Merlet, J.P.: Direct geometrico-static problem of under-constrained cable-driven parallel robots with three cables. In: IEEE International Conference on Robotics and Automation, pp. 3011–3017. Shangai, 9–13 May 2011
9. Nguyen, D.Q., et al.: On the simplification of cable model in static analysis of large dimension cable-driven parallel robots. In: IEEE International Conference on Intelligent Robots and Systems (IROS), pp. 928–934. Tokyo, 3–7 Nov 2013
10. Gouttefarde, M., et al.: Simplified static analysis of large-dimension parallel cable-driven robots. In: IEEE International Conference on Robotics and Automation, pp. 2299–2305. Saint Paul, 14–18 May 2012
11. Jiang, Q., Kumar, V.: The inverse kinematics of 3-d towing. In: ARK, pp. 321–328. Piran, 28 June–1 July 2010
12. Kozak, K., et al.: Static analysis of cable-driven manipulators with non-negligible cable mass. IEEE Trans. Robotics **22**(3), 425–433 (2006)
13. Landsberger, S., Sheridan, T.: A new design for parallel link manipulator. In: Proceedings of the Systems, Man and Cybernetics Conference pp. 812–814. Tucson (1985)
14. Merlet, J.P.: MARIONET, a family of modular wire-driven parallel robots. In: ARK, pp. 53–62. Piran, 28 June–1 July 2010
15. Merlet, J.P., Daney, D.: A portable, modular parallel wire crane for rescue operations. In: IEEE International Conference on Robotics and Automation, pp. 2834–2839. Anchorage, 3–8 May 2010
16. Pott, A.: An algorithm for real-time forward kinematics of cable-driven parallel robots. In: ARK, pp. 529–538. Piran, 28 June–1 July 2010

17. Pott, A., et al.: IPAnema: a family of cable-driven parallel robots for industrial applications. In: 1st International Conference on Cable-Driven Parallel Robots, pp. 119–134. Stuttgart, 3–4 Sept 2012
18. Riehl, N., et al.: Effects of non-negligible cable mass on the static behavior of large workspace cable-driven parallel mechanisms. In: IEEE International Conference on Robotics and Automation, pp. 2193–2198. Kobe, 14–16 May 2009
19. 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 International Conference on Intelligent Robots and Systems (IROS), pp. 1222–1227. Kyongju, 17–21 Oct 1999
20. Tapia, R.: The Kantorovitch theorem for Newton's method. Am. Math. Monthly 78(1.ea), 389–392 (1971)
21. Verhoeven, R.: Analysis of the workspace of tendon-based Stewart platforms. Ph.D. thesis, University of Duisburg-Essen, Duisburg (2004)

Derivatives of Screw Systems in Body-Fixed Representation

Andreas Müller

Abstract The configuration of a kinematic chain can be uniquely expressed in terms of the joint screws via the product of exponentials. Twists on the other hand can be represented in various forms. The particular representation is determined by the reference frame in which the velocity is measured and the reference frame in which this velocity is expressed. For kinematic analyses the spatial twists are commonly used. Analytical mechanism dynamics, on the other hand, uses body-fixed twists. The body-fixed twist of a moving body is the velocity of a body-attached frame relative to the spatial frame expressed in the body-attached moving frame. Accordingly the spatial and body-fixed twists are expressed in terms of spatial and body-fixed instantaneous joint screw coordinates, respectively. Crucial for analytical kinematics and dynamics are the derivatives of twists, and thus of the mechanism's screw system. Whereas higher-order derivatives of screw systems in spatial representation have been a subject of intensive research, the body-fixed representation has not yet been addressed systematically. In this chapter a closed form expression for higher-order partial derivatives of the screw system of a kinematic chain w.r.t. the joint variables is reported. The final expression is a nested Lie bracket of the body-fixed instantaneous joint screws. It resembles the previously presented results for the spatial representation.

Keywords Kinematics · Screw systems · Lie groups · Higher-order analysis

A. Müller (✉)
University of Michigan-Shanghai Jiao Tong University Joint Institute,
Shanghai, People's Republic of China
e-mail: andreas.muller@ieee.org

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_14,

1 Introduction

Velocities of frames (representing rigid bodies or nodal elements of flexible bodies) in a multibody system (MBS) can be expressed recursively in terms of the instantaneous joint screws. Kinematic and dynamic analysis of MBS require higher-order derivative of frame twists, and thus of the joint screws. The twists and joint screws can be expressed in an arbitrary reference, and accordingly their derivatives depend on this reference frame. Mechanism theory and robot kinematics traditionally employ spatial representations of twists, and many publications deal with the issue of higher-order derivatives of screws to aid the analysis of mechanisms. Dynamics formulations on the other hand use body-fixed reference frames and body-fixed twists.

The derivatives of screw systems have been addressed for instance in [3, 8, 14]. It is well-know that for a kinematic chain the partial derivatives of the spatial screw representation is given in terms of Lie brackets, and explicit expressions have been reported [6, 7, 9, 11]. Partial derivatives of the body-fixed representation are also of great importance for MBS kinematics and dynamics, as well sensitivity analysis and optimization [5, 13]. However, explicit closed form expressions have not yet been reported. In this chapter such closed form expressions are presented complementing these for the known spatial representation [9].

2 Kinematics of Open Chains

Consider a kinematic chain comprising 1-DOF screw joints, and let the constant screw coordinates of joint i in the reference configuration be $\mathbf{Y}_i = (\mathbf{e}_i, \mathbf{s}_i \times \mathbf{e}_i + h_i \mathbf{e}_i)^T$. The configuration of the rigid body r in a kinematic chain is represented by the configuration of a body-fixed reference frame (RFR), denoted $\mathbf{C}_r \in SE(3)$. This can be determined in terms of the joint screws by the kinematic mapping $f : \mathbb{V}^n \to SE(3)$ that is formulated by the product of exponential (POE) formula [1, 12]

$$f_r(\mathbf{q}) = \exp(\mathbf{Y}_1 q^1) \cdot \ldots \cdot \exp\left(\mathbf{Y}_r q^r\right) \mathbf{m}_r \tag{1}$$

where $\mathbf{m}_r \in SE(3)$ is the reference configuration of body r for $\mathbf{q} = \mathbf{0}$. Therefore (1) is also referred to as zero reference formulation [4].

The spatial twist $\mathbf{V}_r^s = \left(\omega_r^s, \mathbf{v}_r^s\right)^T$ of body r, i.e. the velocity of RFR at body r relative to the space-fixed frame measured in the space-fixed frame, is introduced as $\widehat{\mathbf{V}}_r^s := \dot{\mathbf{C}}_r \dot{\mathbf{C}}_r^{-1} \in se(3)$. Using (1) the twists vector is expressible as

$$\mathbf{V}_r^s = \mathbf{S}_1(\mathbf{q})\, \dot{q}^1 + \cdots + \mathbf{S}_r(\mathbf{q})\, \dot{q}^r \tag{2}$$

where $\mathbf{S}_i(\mathbf{q}) = (\mathbf{e}_i(\mathbf{q}), \mathbf{s}_i(\mathbf{q}) \times \mathbf{e}_i(\mathbf{q}) + h_i \mathbf{e}_i(\mathbf{q}))^T$ are the instantaneous joint screws, in the configuration $\mathbf{q}$, expressed in the spatial frame. That is, these are the reference screw coordinates transformed to the current configuration

$$\mathbf{S}_i = \mathbf{Ad}_{f_i \mathbf{m}_i^{-1}} \mathbf{Y}_i, \quad i \leq r \tag{3}$$

which follows immediately from (1).

The body-fixed twist $\mathbf{V}_r^b = (\omega_r^b, \mathbf{v}_r^b)^T$ of body r, i.e. the velocity of body-fixed RFR at body r relative to the space-fixed frame measured in the body-fixed RFR, is introduced as $\widehat{\mathbf{V}}_r^b := \dot{\mathbf{C}}_r^{-1} \dot{\mathbf{C}}_r \in se(3)$. The body-fixed twist vector is given by

$$\mathbf{V}_r^b = \underset{r}{\mathbf{B}}_1 (\mathbf{q}) \dot{q}^1 + \cdots + \underset{r}{\mathbf{B}}_r (\mathbf{q}) \dot{q}^r \tag{4}$$

where $\underset{r}{\mathbf{B}}_i(\mathbf{q}) = (\mathbf{e}_i^r(\mathbf{q}), \mathbf{b}_i^r(\mathbf{q}) \times \mathbf{e}_i^r(\mathbf{q}) + h_i \mathbf{e}_i^r(\mathbf{q}))^T$ is the instantaneous screw coordinate vector of joint i expressed in RFR on body r. I.e. $\mathbf{e}_i^r$ is a unite vector along the joint axis i and $\mathbf{b}_i^r(\mathbf{q})$ is the position vector of a point on the axis measured from the RFR on body r and expressed in that RFR. With (1) this is

$$\underset{r}{\mathbf{B}}_i = \mathbf{Ad}^{-1}_{\mathbf{m}_i \mathbf{R}_{r,i}} \mathbf{Y}_i, \quad i \leq r \tag{5}$$

with $\mathbf{R}_{r,i}(\mathbf{q}) := f_i^{-1}(\mathbf{q}) f_r(\mathbf{q}) = \mathbf{m}_i^{-1} \exp\left(\mathbf{Y}_{i+1} q^{i+1}\right) \cdot \ldots \cdot \exp(\mathbf{Y}_r q^r) \mathbf{m}_r$. The latter is the relative configuration of body r w.r.t. body i.

3 Partial Derivatives of Spatial Instantaneous Joint Screws

It is known that the partial derivatives of the spatial screws (3) are given by the Lie brackets

$$\frac{\partial}{\partial q^j} \mathbf{S}_i = [\mathbf{S}_j, \mathbf{S}_i], \ j < i, \quad \frac{\partial}{\partial q^k \partial q^j} \mathbf{S}_i = \begin{cases} [\mathbf{S}_k, [\mathbf{S}_j, \mathbf{S}_i]], & k < j < i \\ [\mathbf{S}_j, [\mathbf{S}_k, \mathbf{S}_i]], & j \leq k < i. \end{cases} \tag{6}$$

This follows directly from the POE (1). It is straightforward to derive explicit expressions for higher order derivatives noting the bilinearity of the Lie product and inserting (6). This has been pursued in [2, 6, 7, 10, 11], and the closed form for arbitrary orders was presented recently [9]. It remains to derive closed form relations for the body-fixed twists. This is presented in the next section.

4 Partial Derivatives of Body-Fixed Joint Screws

As the interest in higher-order relations for body-fixed representations is mainly limited to robot dynamics, and MBS dynamics in particular, there is little research aiming at a systematic and conclusive treatment of this topic. In the following a closed form expression is presented that can be considered as conclusive statement.

The central relations for the following derivations are the bilinearity and skew symmetry of the Lie bracket, the Jacobian identity $[\mathbf{Y},[\mathbf{Z},\mathbf{X}]]+[\mathbf{Z},[\mathbf{X},\mathbf{Y}]]+[\mathbf{X},[\mathbf{Y},\mathbf{Z}]]=\mathbf{0}$, and the fact that for constant $\mathbf{Y}_i$

$$\frac{\partial}{\partial q^i}\exp(\widehat{\mathbf{Y}}_i q^i)=\widehat{\mathbf{Y}}_i\exp(\widehat{\mathbf{Y}}_i q^i). \tag{7}$$

For sake of brevity introduce the notations $\underset{r}{\mathbf{B}}{}^1_{ij}:=\frac{\partial}{\partial q^j}\underset{r}{\mathbf{B}}_i$, $\underset{r}{\mathbf{B}}{}^2_{ijk}:=\frac{\partial}{\partial q^k}\underset{r}{\mathbf{B}}{}^1_{ij}$, and so forth for $\underset{r}{\mathbf{B}}{}^\nu_{ij\ldots kl}:=\frac{\partial}{\partial q^l}\underset{r}{\mathbf{B}}{}^{\nu-1}_{ij\ldots k}$.

The relation (5) leads to the expression for the partial derivatives of $\underset{r}{\mathbf{B}}_i$

$$\frac{\partial}{\partial q^j}\widehat{\underset{r}{\mathbf{B}}}_i=\frac{\partial}{\partial q^j}(f_r^{-1}f_i)\mathbf{m}_i^{-1}\widehat{\mathbf{Y}}_i\mathbf{m}_i f_i^{-1}f_r+f_r^{-1}f_i\mathbf{m}_i^{-1}\widehat{\mathbf{Y}}_i\mathbf{m}_i\frac{\partial}{\partial q^j}(f_i^{-1}f_r). \tag{8}$$

The relation (7) applied to the first term in (8) yields

$$\begin{aligned}\frac{\partial}{\partial q^j}(f_r^{-1}f_i)&=\frac{\partial}{\partial q^j}(\mathbf{m}_r^{-1}\exp(-\mathbf{Y}_r q^r)\ldots\exp(-\mathbf{Y}_{i+1}q^{i+1})\mathbf{m}_i))\\&=-\mathbf{m}_r^{-1}\exp(-\mathbf{Y}_r q^r)\ldots\exp(-\mathbf{Y}_{j+1}q^{j+1})\widehat{\mathbf{Y}}_j\exp(-\mathbf{Y}_j q^j)\ldots\\&\quad\exp(-\mathbf{Y}_{i+1}q^{i+1})\mathbf{m}_i\\&=-f_r^{-1}f_j\mathbf{m}_j^{-1}\widehat{\mathbf{Y}}_j\mathbf{m}_j f_j^{-1}f_i=-f_r^{-1}f_j\mathbf{m}_j^{-1}\widehat{\mathbf{Y}}_j\mathbf{m}_j f_j^{-1}f_r f_r^{-1}f_i\\&=-\widehat{\underset{r}{\mathbf{B}}}_j f_r^{-1}f_i,\ i\le j\le r.\end{aligned}$$

In the same way it follows that

$$\frac{\partial}{\partial q^j}(f_i^{-1}f_r)=f_i^{-1}f_r\widehat{\underset{r}{\mathbf{B}}}_j,\ i\le j\le r$$

so that in summary

$$\begin{aligned}\frac{\partial}{\partial q^j}\widehat{\underset{r}{\mathbf{B}}}_i&=-\widehat{\underset{r}{\mathbf{B}}}_j f_r^{-1}f_i\mathbf{m}_i^{-1}\widehat{\mathbf{Y}}_i\mathbf{m}_i f_i^{-1}+f_r^{-1}f_i\mathbf{m}_i^{-1}\widehat{\mathbf{Y}}_i\mathbf{m}_i f_i^{-1}f_r\widehat{\underset{r}{\mathbf{B}}}_j\\&=\widehat{\underset{r}{\mathbf{B}}}_i\widehat{\underset{r}{\mathbf{B}}}_j-\widehat{\underset{r}{\mathbf{B}}}_j\widehat{\underset{r}{\mathbf{B}}}_i,\ i\le j\le r.\end{aligned}$$

The matrix commutator in the last term is the Lie bracket on $se(3)$, and hence the explicit expression for the partial derivative of the instantaneous joint screws is

$$\underset{r}{\mathbf{B}}{}^1_{ij}=[\underset{r}{\mathbf{B}}_i,\underset{r}{\mathbf{B}}_j],\quad i<j\le r. \tag{9}$$

It is important to notice the index range in (9). As it is clear from the kinematics that the contribution of the ith joint twist to the body twist only depends on the relative

configuration of body i and body r, and thus on the joint variables $q^{i+1}, \ldots, q^r$. Moreover, comparing (9) with (6) shows that the derivative of the spatial twist is identically zero, exactly when the partial derivative of the body-fixed twist is not.

The second partial derivative follows immediately from as (9) and the bilinearity of the Lie bracket as

$$\begin{aligned} {}^2_r\mathbf{B}_{ijk} &= [\frac{\partial}{\partial q^k}{}_r\mathbf{B}_i, {}_r\mathbf{B}_j] + [{}_r\mathbf{B}_i, \frac{\partial}{\partial q^k}{}_r\mathbf{B}_j], \quad i < j \leq r \\ &= [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_k], {}_r\mathbf{B}_j] \quad (\text{for } i < k,\ j \leq r) \\ &\quad + [{}_r\mathbf{B}_i, [{}_r\mathbf{B}_j, {}_r\mathbf{B}_k]] \quad (\text{for } i < j < k \leq r). \end{aligned} \tag{10}$$

The last two terms have an overlapping index range, and are thus partially redundant. This redundancy can be eliminated by application of the Jacobi identity $[[{}_r\mathbf{B}_i, {}_r\mathbf{B}_k], {}_r\mathbf{B}_j] + [[{}_r\mathbf{B}_k, {}_r\mathbf{B}_j], {}_r\mathbf{B}_i] + [[{}_r\mathbf{B}_j, {}_r\mathbf{B}_i], {}_r\mathbf{B}_k] = \mathbf{0}$ for $i < j \leq k$. Solving for $[[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], {}_r\mathbf{B}_k]$ leads to

$${}^2_r\mathbf{B}_{ijk} = \begin{cases} [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], {}_r\mathbf{B}_k], & i < j \leq k \leq r \\ [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_k], {}_r\mathbf{B}_j], & i < k < j \leq r. \end{cases} \tag{11}$$

The third partial derivative of the body-fixed instantaneous screw vector is obtained in the same way noticing the bilinearity of the Lie bracket and the linearity of the derivative operator. Making use of (11) leads to

$${}^3_r\mathbf{B}_{ijkl} = \begin{cases} [\frac{\partial}{\partial q^l}[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], {}_r\mathbf{B}_k], & i \leq j \leq k \leq r \\ [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], \frac{\partial}{\partial q^l}{}_r\mathbf{B}_k], & i \leq j \leq k \leq r \\ [\frac{\partial}{\partial q^l}[{}_r\mathbf{B}_i, {}_r\mathbf{B}_k], {}_r\mathbf{B}_j], & i \leq k \leq j \leq r \\ [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_k], \frac{\partial}{\partial q^l}{}_r\mathbf{B}_j], & i \leq k \leq j \leq r \end{cases}$$

$$= \begin{cases} [[[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], {}_r\mathbf{B}_l], {}_r\mathbf{B}_k], & i \leq j \leq l \leq r,\ i \leq j \leq k \leq r \quad * \\ [[[{}_r\mathbf{B}_i, {}_r\mathbf{B}_l], {}_r\mathbf{B}_j], {}_r\mathbf{B}_k], & i \leq l \leq j \leq r,\ i \leq j \leq k \leq r \\ [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], [{}_r\mathbf{B}_k, {}_r\mathbf{B}_l]], & i \leq j \leq k \leq l \leq r \quad ** \\ [[[{}_r\mathbf{B}_i, {}_r\mathbf{B}_k], {}_r\mathbf{B}_l], {}_r\mathbf{B}_j], & i \leq j \leq l \leq r,\ i \leq k \leq j \leq r \quad *** \\ [[[{}_r\mathbf{B}_i, {}_r\mathbf{B}_l], {}_r\mathbf{B}_k], {}_r\mathbf{B}_j], & i \leq l \leq k \leq r,\ i \leq k \leq j \leq r \\ [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_k], [{}_r\mathbf{B}_j, {}_r\mathbf{B}_l]], & i \leq k \leq j \leq l \leq r\,. \quad **** \end{cases}$$

Again the terms * and **, as well as *** and **** have respectively overlapping index ranges. These can again be eliminated with help of the Jacobi identity. To this end, the identity $[{}_r\mathbf{B}_k, [{}_r\mathbf{B}_l, [{}_r\mathbf{B}_i, {}_r\mathbf{B}_j]]] + [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], [{}_r\mathbf{B}_k, {}_r\mathbf{B}_l]] + [{}_r\mathbf{B}_l, [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], {}_r\mathbf{B}_k]] = 0$ is solved for $[[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], [{}_r\mathbf{B}_k, {}_r\mathbf{B}_l]]$ so that $* + ** = -[{}_r\mathbf{B}_l, [[{}_r\mathbf{B}_i, {}_r\mathbf{B}_j], {}_r\mathbf{B}_k]]$ in the range $i \leq j \leq k \leq l$. In order to harmonize the overlapping index ranges of *** and ****

the identity $[\underset{r}{\mathbf{B}}_j, [\underset{r}{\mathbf{B}}_l, [\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_k]]] + [[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_k], [\underset{r}{\mathbf{B}}_j, \underset{r}{\mathbf{B}}_l]] + [\underset{r}{\mathbf{B}}_l, [[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_k], \underset{r}{\mathbf{B}}_j]] = 0$ is solved for $[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_k], [\underset{r}{\mathbf{B}}_j, \underset{r}{\mathbf{B}}_l]]$ so that $*\,*\,*\,+\,*\,*\,*\,** = -[\underset{r}{\mathbf{B}}_l, [[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_j], \underset{r}{\mathbf{B}}_j]]$ for $i \le k \le j \le l$. This leads to

$$\underset{r}{\mathbf{B}}{}^{3}_{ijkl} = \begin{cases} -[\underset{r}{\mathbf{B}}_l, [[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_j], \underset{r}{\mathbf{B}}_k]], & i < j \le k \le l \le r \\ [\underset{r}{\mathbf{B}}_k, [\underset{r}{\mathbf{B}}_l, [\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_j]]], & i < j \le l \le k \le r \\ [[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_l], \underset{r}{\mathbf{B}}_j], \underset{r}{\mathbf{B}}_k], & i < l < j \le k \le r \\ -[\underset{r}{\mathbf{B}}_l, [[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_k], \underset{r}{\mathbf{B}}_j]], & i < k \le j \le l \le r \\ [\underset{r}{\mathbf{B}}_j, [\underset{r}{\mathbf{B}}_l, [\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_k]]], & i < k < l \le j \le r \\ [[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_l], \underset{r}{\mathbf{B}}_k], \underset{r}{\mathbf{B}}_j], & i < l < k \le j \le r \end{cases} \tag{12}$$

$$= \begin{cases} [[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_j], \underset{r}{\mathbf{B}}_k], \underset{r}{\mathbf{B}}_l], & i < j \le k \le l \le r \\ [[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_j], \underset{r}{\mathbf{B}}_l], \underset{r}{\mathbf{B}}_k], & i < j \le l \le k \le r \\ [[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_k], \underset{r}{\mathbf{B}}_j], \underset{r}{\mathbf{B}}_l], & i < k < j \le l \le r \\ [[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_k], \underset{r}{\mathbf{B}}_l], \underset{r}{\mathbf{B}}_j], & i < k \le l \le j \le r \\ [[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_l], \underset{r}{\mathbf{B}}_j], \underset{r}{\mathbf{B}}_k], & i < l < j \le k \le r \\ [[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_l], \underset{r}{\mathbf{B}}_k], \underset{r}{\mathbf{B}}_j], & i < l < k \le j \le r \end{cases} \tag{13}$$

where the last form is obtained noticing the skew symmetry of Lie bracket.

Proceeding for higher derivatives, the closed form expressions are obtained in the same manner. In particular for each term of order ν the index range is split leading respectively to two overlapping index ranges. These overlaps are eliminated invoking the Jacobi identity in an appropriate subrange, and the skew symmetry gives rise to the systematic ordering so that the Lie brackets are nested from left to right. Inspection of the resulting expressions shows an apparent systematics, induced by the separation into disjoined index ranges, giving rise to a general rule. This general expression for the νth order derivative is summarized as follows.

Corollary 1 The non-zero terms in the νth partial derivative of the instantaneous joint screws $\underset{r}{\mathbf{B}}_i$ of a kinematic chain, with (forward) kinematic mapping (1), is given by

$$\underset{r}{\mathbf{B}}{}^{\nu}_{i\alpha_1\alpha_2\ldots\alpha_\nu} = [\ldots[[[\underset{r}{\mathbf{B}}_i, \underset{r}{\mathbf{B}}_{\beta_1}], \underset{r}{\mathbf{B}}_{\beta_2}], \underset{r}{\mathbf{B}}_{\beta_3}]\ldots, \underset{r}{\mathbf{B}}_{\beta_\nu}], \; i < \beta_1 \le \beta_2 \le \cdots \le \beta_\nu \le r \tag{14}$$

with

$$\{\beta_1, \beta_2, \ldots, \beta_\nu\} \in \Pi_\nu\,(\alpha_1, \ldots, \alpha_\nu)$$

where Π_ν is the permutation group of the indices $\alpha_1, \ldots, \alpha_\nu$.

This is the explicit closed form expression for the partial derivative of arbitrary order ν. The corollary says that the repeated derivative w.r.t. $q^{\alpha_1}, \ldots, q^{\alpha_\nu}$ is non-zero if and only if the indexes $\alpha_1, \ldots, \alpha_\nu$ can be permuted so to satisfy the inequality (14), and it is given by the nested Lie brackets in (14). Each permutation gives rise to an index range where the derivative is non-zero. Inspection of (9), (11), and (12) indicates a sharp increase of the number of these index ranges. In fact the number of index ranges with non-zero partial derivative of order ν is $\nu!$.

Comparing the corresponding result for the spatial screw system, reported in [9], the closed form for the spatial and body-fixed version apparently only differs in the index range and in the way in which the brackets are arranged.

5 Conclusions

The body-fixed twists are essential for mechanism analysis and dynamics in particular since the kinetic energy is naturally expressed in terms of body-fixed quantities. Hence the derivatives of the body-fixed representation of instantaneous joint screws is vital. Kinematic and static sensitivity measures require first partial derivatives of a mechanism's screw system (i.e. its Jacobian). Also rigid body (and also flexible) body dynamics requires partial derivatives of first order in order to derive the motion equations. Dynamic sensitivity considerations need second order derivatives (recursive approaches were reported in [5]). Moreover, nonlinear dynamics and control demands higher-order derivatives of body-fixed screws. Even though no closed form expressions have been presented in the literature so far. Such expressions are derived in this chapter that have not appeared previously in the literature.

It should be mentioned that, noticing the generality of (1), the presented expressions can be adopted straightforwardly for partial derivatives w.r.t. to geometric design parameters. This has certainly some significance for optimal design of robots and mechanisms.

It should finally be mentioned that the νth-order derivative can be recursively evaluated from the lower-order derivatives. To this end denote $\mathscr{B}_r^\nu := \sum_{j\le r} \mathbf{B}_{r\,j} \frac{d^\nu}{dt^\nu} q^j$. Then it follows that

$$\begin{aligned}\dot{\mathscr{B}}_r^\nu &= \sum_{i<j\le r} [\mathbf{B}_{r\,i} \frac{d^\nu}{dt^\nu} q^i, \mathbf{B}_{r\,j} \dot{q}^j] + \mathscr{B}_r^{\nu+1} \\ &= \sum_{i\le r}\sum_{j\le r} [\mathbf{B}_{r\,i} \frac{d^\nu}{dt^\nu} q^i, \mathbf{B}_{r\,j} \dot{q}^j] - \sum_{i\le r}\sum_{j<i} [\mathbf{B}_{r\,i} \frac{d^\nu}{dt^\nu} q^i, \mathbf{B}_{r\,j} \dot{q}^j] + \mathscr{B}_r^{\nu+1}\end{aligned}$$

$$= \sum_{i \le r} \sum_{j<i} [\mathbf{B}_{r\,j} \dot{q}^j, \mathbf{B}_{r\,i} \frac{d^\nu}{dt^\nu} q^i] + [\mathscr{B}_r^\nu, \mathscr{B}_r^1] + \mathscr{B}_r^{\nu+1}.$$

This relation might be useful for recursive implementations.

References

1. Brockett, R.: Robotic manipulators and the product exponentials formulae. Math. Theor. Netw. Syst. Lect. Notes Control Inf. Sci. **58**, 120–129 (1984)
2. Cervantes-Sánchez, J.J., et al.: The differential calculus of screws: theory, geometrical interpretation, and applications. Proc. Inst. Mech. Eng. C. J. Mech. Eng. Sci. **223**, 1449–1468 (2009)
3. Gallardo-Alvarado, J., Orozco-Mendoza, H., Rodrí guez-Castro, R.: Finding the jerk properties of multi-body systems using helicoidal vector fields. In: Proc. Inst. Mech. Eng. C. J. Mech. Eng. Sci. **222**, 2217–2229 (2008)
4. Gupta, K.C.: Kinematic analysis of manipulators using the zero reference position description. Int. J. Rob. Res. **5**(2), 5–13 (1986)
5. Hsu, Y., Anderson, K.S.: Recursive sensitivity analysis for constraintee multi-rigid-body dynamic systems design optimization. Struct. Multidisc Optim. **24**, 312–324 (2002)
6. Karger, A.: Singularity analysis of serial robot-manipulators. ASME J. Mech. Des. **118**(4), 520–525 (1996)
7. Lerbet, J.: Analytic Geometry and Singularities of Mechanisms. ZAMM, Z. angew. Math. Mech. **78**(10b), 687–694 (1999)
8. Lipkin, H.: Time derivatives of screws with applications to dynamics and stiffness. Mech. Mach. Theor. **40**(3), 259–273 (2005)
9. Müller, A.: Closed form higher derivatives of screw systems. In: 37th Mechanisms and Robotics Conference (MECH), ASME 2013 International Design Engineering Technical Conferences, Portland, OR, August 4–7 (2013)
10. Perez-Soto, G.I., et al.: On the computer solutions of kinematics analysis of linkages. In. Eng. Comput. (2013). doi:10.1007/s00366-013-0321-2
11. Rico, J.M., Gallardo, J., Duffy, J.: Screw theory and higher order kinematic analysis of open serial and closed chains. Mech. Mach. Theor. **34**(4), 559–586 (1999)
12. Selig, J.M.: Geometric fundamentals of robotics. In: Monographs in Computer Science, 2nd edn. Springer, New York, (2005)
13. Sohl, J.A., Bobrow, J.E.: A recursive multibody dynamics and sensitivity algorithm for branched kinematic chains. ASME J. Dyn. Syst. Meas. Contr. **123**, 391–399 (2001)
14. Wohlhart, K.: Time derivatives of the velocity motor. In: Proceedings of the SYROM, 2001, vol. 2, pp. 371–376. Bucharest (2001)

Sharp Linkages

Zijia Li

Abstract In this chapter, we consider a special kind of overconstrained 6R closed linkages which we call sharp linkages. These are linkages with the property that their bond diagram looks like a ♯ sign. We give a construction of this linkage using the bond theory and motion polynomial factorization methods. These two methods are introduced recently in [6, 7]. Another type of 6R linkages is also introduced. To my knowledge, both types of linkages are new.

Keywords Dual quaternions · Motion polynomials · Factorization · Bond theory · Overconstrained 6R linkages

1 Introduction

In kinematics, a closed 6R linkages with mobility one have been considered by many authors (see [1, 3, 4, 6, 11–13]).

In this chapter, we mainly focus on closed 6R linkages. More precisely, we consider a very special type of 6R linkages, which we call sharp linkages. Their bond diagrams look like a ♯ sign. Namely, this bond diagram has two Bennett conditons as the bond diagram of Waldrons double Bennett hybrid, Dietmaier 6R linkages and Bricard plane symmetric 6R linkages [3, Sect. 4.8.3]. But it is not a special case of those 6R linkages. We also get another 6R linkage which has quasi-symmetric bond diagram. One can find a special angle symmetric 6R linkage in [8] with the same bond diagram. But this new linkage is not angle symmetric, because there is one pair of opposite angles which are not equal.

Z. Li (✉)
Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences (RICAM), Altenbergerstrasse 69, 4040 Linz, Austria
e-mail: zijia.li@oeaw.ac.at

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_15,

Our main tools are the bond theory and the factorization of a motion polynomial. These two are based on dual quaternions. In the chapter [6], the authors found a new 6R linkage by using the factorization of a cubic motion polynomial. We find a quartic motion polynomial in this chapter. This quartic motion polynomial has two factorizations which generate two 3R open chains. Using these two chains we can construct the sharp linkage. The main difficulty is to find such quartic motion polynomials. In the future, we want to find all quartic motion polynomial that lead to closed 6R linkages.

The remaining part of the chapter is set up as follows. In Sect. 2, we give the preliminaries we need i.e. dual quaternions, the factorization of a motion polynomial, the bond theory. Section 3 introduces our motivation. Section 4 contains the main result and examples.

2 Preliminaries

In this chapter, we mainly use two tools (Bond theory and motion polynomial factorization). Before introducing each of them, let us recall the dual quaternions.

2.1 Dual Quaternions

The algebra $\mathbb{DH}$ of dual quaternions is the 8-dimensional real vector space generated by $1, \varepsilon, \mathbf{i}, \mathbf{j}, \mathbf{k}, \varepsilon\mathbf{i}, \varepsilon\mathbf{j}, \varepsilon\mathbf{k}$ (see [6]). Following [6], we can represent a rotation by a dual quaternion of the form $\left(\cot\left(\frac{\phi}{2}\right) - h\right)$, where ϕ is the rotation angle and h is a dual quaternion such that $h^2 = -1$ depending only on the rotation axis. We use projective representations, which means that two dual quaternions represent the same Euclidean displacement if only if one is a real scalar multiple of the other.

The set of all possible motions of a closed 6R linkage is determined by the position of the six rotation axes in some fixed initial configuration. Let L be a 6R linkage given by six lines, represented by dual quaternions $h_1, \ldots, h_6$ such that $h_i^2 = -1$ for $i = 1, \ldots, 6$. A configuration (see [6]) is a 6-tuple $(t_1, \ldots, t_6)$, such that the closure condition

$$(t_1 - h_1)(t_2 - h_2)(t_3 - h_3)(t_4 - h_4)(t_5 - h_5)(t_6 - h_6) \in \mathbb{R}\backslash\{0\} \tag{1}$$

holds. The configuration parameters t_i—the cotangents of the rotation angles—may be real numbers or ∞, and in the second case we evaluate the expression $(t_i - h_i)$ to 1, the rotation with angle 0. The set of all configurations of L is denoted by K_L. We say L is movable when K_L is a one-dimensional set. Mostly, we will assume, slightly stronger, that there exists an irreducible one-dimensional set for which none of the t_i is fixed. Such a component is called a non-degenerate component. We also exclude the case $\dim_{\mathbb{C}} K_L \geq 2$. Linkages with mobility ≥ 2 do exist, for instance linkages with all axes parallel have mobility 3.

2.2 The Factorization of a Motion Polynomial

In the chapter [6], the authors introduced the motion polynomial P which is a monic polynomial in Study quadric of degree n with $P\bar{P} \in \mathbb{R}[t]$. Let $h_1, h_2, \ldots, h_n$ be rotations; using their algorithm, in general, one can compute a factorization

$$P = (t - h_1)(t - h_2) \cdots (t - h_n).$$

One application of the factorization of a motion polynomial is to construct closed linkages by combining the different factorization (which corresponding to different open chains). The difficulty is to find a quartic (or higher degree) motion polynomial which has two factorizations. Furthermore, each of these two factorizations should be corresponding to an open 3R chains. Our main contribution is that we construct such two special quartic motion polynomials. Our construction is based on the bond theory [7].

2.3 The Bond Theory

Let $L = (h_1, \ldots, h_6)$ be a closed 6R linkage with mobility 1. We assume, for simplicity, that the configuration curve $K_L \subset (\mathbb{P}^1_{\mathbb{R}})^n$ has only one component of dimension 1. Let $K_{\mathbb{C}} \subset (\mathbb{P}^1_{\mathbb{C}})^n$ be the Zariski closure of K_L. We set

$$B := \{(t_1, \ldots, t_n) \in K_{\mathbb{C}} \mid (t_1 - h_1)(t_2 - h_2) \cdots (t_n - h_n) = 0\}. \tag{2}$$

The set B is a finite set of conjugate complex points on the configuration curve's Zariski closure.

Let β be a bond with coordinates $(t_1, \ldots, t_n)$. By Theorem 2 in [7], there exist indices $i, j \in [n]$, $i < j$, such that $t_i^2 + 1 = t_j^2 + 1 = 0$. If there are exactly two coordinates of β with values $\pm$i, then we say that β *connects* joints i and j. In general, the situation, is more complicated.

We visualize bonds and their connection numbers by *bond diagrams.* We start with the link diagram, where vertices correspond to links and edges correspond to joints. Then we draw a connecting line between the edges h_i and h_j for each set $\{\beta, \overline{\beta}\}$ of conjugate complex bonds. Multiple connections are possible.

Let us recall [7, Corollary 12] for explaining the connection.

Corollary 1 *For a bond β with $t_i^2 + 1 = t_j^2 + 1 = 0$ and $i < j$, the equality*

$$(t_i - h_i)(t_{i+1} - h_{i+1}) \cdots (t_j - h_j) = 0 \tag{3}$$

holds.

In Fig. 1, we show some known examples and our new examples with bond diagrams.

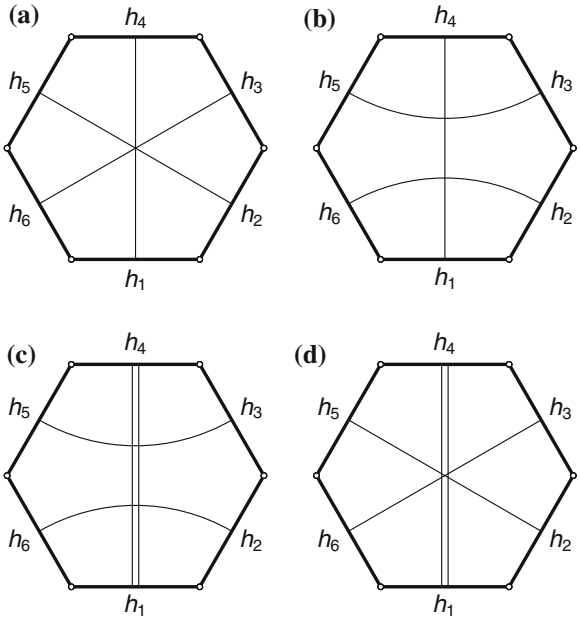


Fig. 1 Bond diagrams for the cube linkage (**a**), the Waldrons double Bennett hybrid (**b**), the sharp linkage type one (**c**), the sharp linkage type two (**d**)

3 Motivation

In the chapter [6], the authors constructed a new 6R linkage by using the factorization of a cubic motion polynomial. It has bond diagram of Fig. 1a which is one of simplest bond diagrams. The other one of simplest bond diagram is Fig. 1b which is known as the Waldrons double Bennett hybrid (see [3], Sect. 4.2.5). There is no other 6R linkages with bond diagrams of only three bond connections. Using [7, Theorem 23], one can find the reason as an excise. We consider diagrams with four bond connections, e.g. Fig. 1c, d. There are some other types of bond diagrams with four bond connections. We only consider these two types in this chapter.

4 The Main Results

First, let us make our purpose clear. We want to construct a monic quartic polynomial Q in $\mathbb{DH}[t]$ such that $Q\bar{Q} \in \mathbb{R}[t]$. Furthermore, we can factor Q in two different ways (at least) which both constitute a 3R open chain. Then we can construct a 6R linkage by combining these two factorizations.

Remark 1 Such 6R linkage exists (angle symmetric 6R linkage [8, 9]). Up to now, it was not known whether or not there exist such 6R linkages that are not angle symmetric. This chapter gives a positive answer.

Now we introduce our procedure for finding such examples.

I. We choose four lines with two different bond connections (3) as following

$$\begin{aligned}(\mathrm{i}-h_1)(\alpha-h_2)(\beta-h_3)(\mathrm{i}-h_4)&=0,\\ (\mathrm{i}-h_1)(\alpha'-h_2)(\beta'-h_3)(\mathrm{i}+h_4)&=0,\end{aligned}$$

where i is the imaginary unit, complex numbers α and β have the same linear relation as α' and β' i.e.

$$\beta=a\alpha+b,\quad \beta'=a\alpha'+b.$$

II. Use these two bond conditions to calculate quartic motion polynomials.
III. Use the factorization algorithm to compute another factorization of the first three factors. This procedure contribute the two lines h_5 and h_6 which we want.
IV. Return the 6R linkage $[h_1, h_2, h_3, h_4, h_5, h_6]$.

Remark 2 There are two options in procedure III (either change the order of second and third or not), which contribute two kinds of 6R linkage with bond diagrams 1(c) and (d).

As the first step is the most important step, we show the details in the following subroutine.

I.a Choose h_2 and h_3 as two random lines with $h_2^2=h_3^2=-1$.
I.b Choose two complex number α and α' where $\alpha\neq\pm\mathrm{i}$ and $\alpha'\neq\pm\mathrm{i}$.
I.c Choose two random real numbers a, b with $a\neq 0$.
I.d Assume that the other two lines have the following formula

$$\begin{aligned}h_1&=(x_1\mathbf{i}+x_2\mathbf{j}+x_3\mathbf{k})+(y_1\mathbf{i}+y_2\mathbf{j}+y_3\mathbf{k})\varepsilon,\\ h_4&=(u_1\mathbf{i}+u_2\mathbf{j}+u_3\mathbf{k})+(v_1\mathbf{i}+v_2\mathbf{j}+v_3\mathbf{k})\varepsilon.\end{aligned}$$

I.e Solve the following system for unknowns $x_1, x_2, x_3, y_1, y_2, y_3, u_1, u_2, u_3, v_1, v_2, v_3$

$$\begin{cases}(\mathrm{i}-h_1)(\alpha-h_2)(\beta-h_3)(\mathrm{i}-h_4)=0,\\ (\mathrm{i}-h_1)(\alpha'-h_2)(\beta'-h_3)(\mathrm{i}+h_4)=0,\\ h_1^2=-1,\qquad h_4^2=-1.\end{cases}$$

I.f Choose one solution (all variables are in real) for the next steps.

We add one example to support our procedure. This is a particularly easy example which we found by our procedure.

Input: I.a, I.b, I.c

$$h_2 = \left(-\frac{3}{5}\mathbf{i} - \frac{4}{5}\mathbf{j}\right) - \frac{6}{5}\mathbf{k}\varepsilon,$$
$$h_3 = \left(\frac{3}{7}\mathbf{i} - \frac{2}{7}\mathbf{j} + \frac{6}{7}\mathbf{k}\right) + \left(\frac{76}{49}\mathbf{i} + \frac{24}{49}\mathbf{j} - \frac{30}{49}\mathbf{k}\right)\varepsilon,$$
$$\alpha = -\frac{1}{5} - \frac{4}{3}\mathrm{i}, \quad \alpha' = \frac{4}{5} - \frac{1}{2}\mathrm{i},$$
$$a = \frac{5}{2}, \qquad b = -\frac{3}{4}.$$

Output: Then one can get a numerical solution with ten digits as following

$$x_1 = 0.4058453976, \quad x_2 = -0.9139192147, \quad x_3 = -0.0064173294,$$
$$y_1 = 1.244931364, \quad y_2 = 0.5535129673, \quad y_3 = -0.09606363509,$$
$$u_1 = -0.6219669897, \quad u_2 = -0.3316117352, \quad u_3 = 0.7093593733,$$
$$v_1 = -0.5417103337, \quad v_2 = -1.024569908, \quad v_3 = -0.9539386886.$$

Then the next two steps are for calculating the factorization. We assume that $t_1(t)$ and $t_4(t)$ are quadric rational functions of t, and we also assume that

$$t_1(\alpha) = \mathrm{i}, \quad t_1(\alpha') = \mathrm{i}, \quad t_4(\alpha) = \mathrm{i}, \quad t_4(\alpha') = -\mathrm{i}. \tag{4}$$

The quartic motion polynomial is $(t_1(t) - h_1)(t - h_2)(at + b - h_3)$. The other factorization is obtained by multiplying $(t_4(t) - h_4)$ from the right. Then $(t_1(t) - h_1)(t - h_2)(at + b - h_3)(t_4(t) - h_4)$ is a quadric motion polynomial when we remove the real denominators and factors. The next step is to factor this quadric motion polynomial. We show all these details in the following:

Assumption:

$$t_1(t) = \frac{t^2 + p_2 t + p_3}{p_4 t + p_5}, \quad t_4(t) = \frac{t^2 + p_2' t + p_3'}{p_4' t + p_5'},$$
$$\alpha = -\frac{1}{5} - \frac{4}{3}\mathrm{i}, \quad \alpha' = \frac{4}{5} - \frac{1}{2}\mathrm{i}.$$

Do: Solve the linear system (4) for unknowns p_2, p_3, p_4, p_5, p_2', p_3', p_4', p_5'.

Output: Then one can get a solution of $t_1(t)$ and $t_4(t)$ as following

$$t_1(t) = \frac{t^2 - \frac{3}{5}t - \frac{62}{75}}{-\frac{11}{6}t + \frac{29}{30}}, \qquad t_4(t) = \frac{t^2 - \frac{3}{5}t + \frac{38}{75}}{-\frac{5}{6}t + \frac{7}{6}}.$$

After substituting $t_1(t)$ and $t_4(t)$ into

$$(t_1(t) - h_1)(t - h_2)(at + b - h_3)(t_4(t) - h_4),$$

we have a numeric quadric motion polynomial in 10 digits (replacing the real denominators and factors)

$$\begin{aligned} t^2 &+ (-0.3000000000 + 0.6543154994\mathbf{i} - 1.037575959\mathbf{j} + 0.2365105645\mathbf{k} \\ &+ 1.210540727\mathbf{i}\varepsilon - 0.0349528507\mathbf{j}\varepsilon + 0.4738323880\varepsilon\mathbf{k})t \\ &- 70.2003149450 - 0.0160185109\mathbf{i} + 0.3911798525\mathbf{j} + .2378984092\mathbf{k} \\ &- 0.9404081633\varepsilon - 1.436504834\mathbf{i}\varepsilon - 0.5526215606\mathbf{j}\varepsilon + 0.0201175896\varepsilon\mathbf{k}. \end{aligned}$$

As the norm of this quadric motion polynomial is $(t^2 + 1)(t^2 - \frac{3}{5}t + \frac{1}{4})$, we can construct two 6R linkages $L_c = [h_1^c, h_2^c, h_3^c, h_4^c, h_5^c, h_6^c]$ and $L_d = [h_1^d, h_2^d, h_3^d, h_4^d, h_5^d, h_6^d]$ (with bond diagram 1c, d) basing on these two factorization as following (numerically in ten digits).

$$\begin{aligned} h_1^c &= (0.4058453976\mathbf{i} - 0.9139192147\mathbf{j} - 0.0064173294\mathbf{k}) \\ &\quad + (1.244931364\mathbf{i} + 0.5535129673\mathbf{j} - 0.09606363509\mathbf{k})\,\varepsilon, \\ h_2^c &= \left(-\frac{3}{5}\mathbf{i} - \frac{4}{5}\mathbf{j}\right) - \frac{6}{5}\mathbf{k}\varepsilon, \\ h_3^c &= \left(\frac{3}{7}\mathbf{i} - \frac{2}{7}\mathbf{j} + \frac{6}{7}\mathbf{k}\right) + \left(\frac{76}{49}\mathbf{i} + \frac{24}{49}\mathbf{j} - \frac{30}{49}\mathbf{k}\right)\varepsilon, \\ h_4^c &= (-0.6219669897\mathbf{i} - 0.3316117352\mathbf{j} + 0.7093593733\mathbf{k}) \\ &\quad + (-0.5417103337\mathbf{i} - 1.024569908\mathbf{j} - 0.9539386883\mathbf{k})\,\varepsilon, \\ h_5^c &= (0.9529670102)\mathbf{i} - 0.2884245020\mathbf{j} - 0.0930869702\mathbf{k}) \\ &\quad + (0.145998817\mathbf{i} - 0.4419436106\mathbf{j} + 2.863982166\epsilon\mathbf{k})\,\varepsilon, \\ h_6^c &= (0.2731286954)\mathbf{i} - 0.9222061578\mathbf{j} + 0.2737453525\mathbf{k}) \\ &\quad + (1.152141200\mathbf{i} + 0.1418245937\mathbf{j} - 0.6717604788\mathbf{k})\,\varepsilon. \end{aligned}$$

$$\begin{aligned} h_1^d &= h_1^c, \quad h_2^d = h_2^c, \quad h_3^d = h_3^c, \quad h_4^d = h_4^c, \\ h_5^d &= (0.6843121346\mathbf{i} - 0.7290081982\mathbf{j} - 0.0162465108\mathbf{k}) \\ &\quad + (0.7852041130\mathbf{i} + 0.7074301081\mathbf{j} + 1.329661169\mathbf{k})\,\varepsilon, \\ h_6^d &= (-0.0749915882\mathbf{i} - 0.7714194013\mathbf{j} + 0.6318926880\mathbf{k}) \\ &\quad + (1.063341534\mathbf{i} - 1.855957397\mathbf{j} - 2.139571953\mathbf{k})\,\varepsilon. \end{aligned}$$

Remark 3 At several places, we used the computer algebra system Maple for more elaborate computations: examples, animations. Because of the length of these computations, it is not reasonable to reproduce them in this chapter, but they can be found

at our webpage.[1] They can be read with any text editor and verified using Maple 16. One can use a new technique, namely, quad polynomials [10],[2] to check mobility from their symbolic Denavit/Hartenberg parameters[2, 5][3] which have complicate square roots.

Acknowledgments We would like to thank Gábor Hegedüs, Hans-Peter Schröcker and Josef Schicho for discussion and helpful remarks. The research was supported by the Austrian Science Fund (FWF): W1214-N15, project DK9.

References

1. Baker, J.E.: An analysis of the Bricard linkages. Mech. Mach. Theory **15**(4), 267–286 (1980)
2. Denavit, J., Hartenberg, R.S.: A kinematic notation for lower-pair mechanisms based on matrices. Transactions of the ASME. J. Appl. Mech. **22**, 215–221 (1955)
3. Dietmaier, P.: Einfach übergeschlossene Mechanismen mit Drehgelenken. Habilitation thesis, Graz University of Technology (1995)
4. Goldberg, M.: New five-bar and six-bar linkages in three dimensions. Trans. ASME **65**, 649–656 (1943)
5. Hegedüs, G., Li, Z., Schicho, J., Schröcker, H.P.: The theory of bonds ii: Closed 6R linkages with maximal genus. ArXiv e-prints (2013)
6. Hegedüs, G., Schicho, J., Schröcker, H.P.: Factorization of rational curves in the study quadric. Mech. Mach. Theory **69**, 142–152 (2013)
7. Hegedüs, G., Schicho, J., Schröcker, H.P.: The theory of bonds: a new method for the analysis of linkages. Mech. Mach. Theory **70**, 407–424 (2013)
8. Li, Z., Schicho, J.: Classification of angle-symmetric 6R linkages. Mech. Mach. Theory **70**, 372–379 (2013)
9. Li, Z., Schicho, J.: Three types of parallel 6R linkages. In: Thomas, F., Perez Gracia, A. (eds.) Computational Kinematics, *Mechanisms and Machine Science*, vol. 15, pp. 111–119. Springer, Netherlands (2014)
10. Li, Z., Schicho, J.: A new technique for analyzing 6R linkages: Quad Polynomials. Tech. rep. (2014)
11. Sarrus, P.: Note sur la transformation des mouvements rectilignes alternatifs, en mouvements circulaires: et rèciproquement. Comptes Rendus des Séances de l'Académie des Sciences de Paris **36**, 1036–1038 (1853)
12. Waldron, K.J.: Overconstrained linkages. Environ. Plann. B-plann. Des. **6**, 393–402 (1979)
13. Wohlhart, K.: Merging two general Goldberg 5R linkages to obtain a new 6R space mechanism. Mech. Mach. Theory **26**, 659–668 (1991)

[1] http://people.ricam.oeaw.ac.at/z.li/softwares/sharplinkages.html

[2] http://people.ricam.oeaw.ac.at/z.li/softwares/quadpolynomials.html

[3] One can check that L_c does not fulfill the necessary conditions of Waldrons double Bennett hybrid, Dietmaier 6R linkages or Bricard plane symmetric 6R linkages as an exercise.

Solvable Multi-Fingered Hands for Exact Kinematic Synthesis

Abhijit Makhal and Alba Perez-Gracia

Abstract Multi-fingered hands are kinematic chains with a tree topology, that is, with a set of common joints that span several branches and end-effectors. When performing dimensional kinematic synthesis with simultaneous tasks for all the end-effectors, a new solvability criterion needs to be applied that includes checking the solvability of sub-chains. This criterion yields as a result that not all possible topologies are solvable for a common number of positions for all end-effectors. This article shows and proves the solvability criterion and derives some properties of the kinematic chains with tree topology for a single branching and identical fingers.

Keywords Knematic synthesis · Multi-fingered hands

1 Introduction

Kinematic chains with a tree topology consist of several common joints that branch to a number of serial chains, each of them corresponding to a different end-effector. A typical example of a kinematic chain with a tree topology is a wristed, multi-fingered hand.

Compared to other topologies, the tree topologies have not been so widely studied. Kinematic analysis for applications in modular robots and robotic hands can be found in [8, 9], and [1], and dynamic analysis is found in [3] and [2]. Structural synthesis for multiple fingers with no wrist, considering grasping and manipulation requirements, are found in [4]. The first reference to kinematic design of tree topologies is found in [5].

A. Makhal · A. Perez-Gracia (✉)
Department of Mechanical Engineering, Idaho State University, Pocatello, USA
e-mail: makhabhi@isu.edu

A. Perez-Gracia
e-mail: perealba@isu.edu

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_16,

The kinematic synthesis of these topologies presents particular challenges that are different of those that appear in single serial chains or in closed-loop systems. In particular, the kinematic synthesis of multi-fingered hands has been explored also in [7] and more extensively in [6].

When dealing with exact kinematic synthesis, one of the first steps is to calculate the maximum number of positions that can be used, which define the workspace of the chain. In the case of tree topologies, consider a task having the same number of positions for each of the multiple end-effectors; this means that we are dealing with a coordinated action of all those end-effectors, denoted as a *simultaneous task*.

In this chapter we focus on the particular issues that appear in tree topologies when dealing with exact synthesis for simultaneous positions of all end-effectors. The solvability for the simultaneous task case presented in [6] is proved and developed in further detail, and a basic classification, together with some results regarding solvable multi-fingered chains with identical fingers, are included.

2 Tree Topologies

We denote a tree topology for a kinematic chain as that of a chain having a set of common joints spanning several chains and ending in multiple end-effectors. The tree topology is modeled using graph theory; for this we follow the approach of Tsai [10]. The kinematic chain is represented as a rooted graph, with the root vertex being fixed with respect to a reference system.

A tree topology is denoted as $SerialChain - (Branch_1, Branch_2, \ldots, Branch_b)$, where $SerialChain$ are the common joints and the dash indicates a branching, with the branches contained in the parenthesis, each branch $Branch_i$ characterized by its type and number of joints. In the case of using just revolute joints, the joint type is dropped and only the number of joints is indicated. Figure 1 presents the compacted graph for a $3R - (2R, R - (R, R, R))$, or $3 - (2, 1 - (1, 1, 1))$ chain, with two branches, one of them branching again on three additional branches, for a total of four end-effectors. Branches are ordered according to their branching order (branch 1 is the first one to branch). The root vertex is indicated with a double circle.

In tree topologies, a vertex can be connected to several edges defining several branches; a tree topology will always have links that are ternary or above, which are identified in the graph as a vertex spanning several edges. For the purpose of this work we consider serial chains in each branch, with only revolute joints. For the reduction of closed-loop chains to serial chains for synthesis purposes, see [6].

The contracted graph does not have binary vertices, and the primary vertices are either the root node or end-effectors. Having several end-effectors allows us to change the root node to any of them: consider displacements $\mathbf{P}_i$ to each end-effector i, where $i = 1$ is the root node. The inversion of the root from $i = 1$ to $i = j$ is

$$\mathbf{P}_i^* = \mathbf{P}_j^{-1}\mathbf{P}_i, \tag{1}$$

where $\mathbf{P}_i^*$ would be the i^{th} end-effector's position with respect to the new root node j.

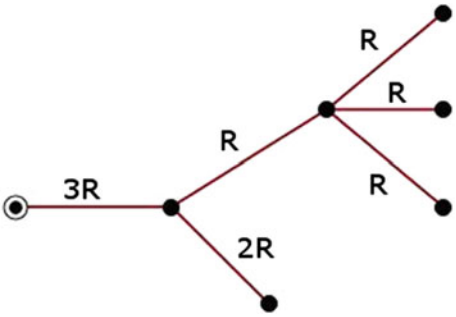


Fig. 1 A $3-(2,1-(1,1,1))$ tree topology

3 Dimensional Kinematic Synthesis for Tree Topologies

Dimensional kinematic synthesis seeks to find the position of the joint axes for a given topology, in order for each of the end-effectors to perform a given set of displacements. In this section we present a summary of the design methodology when the chosen topology is that of multi-fingered hands; for details, see [6].

Given a set of m task positions $\hat{P}^i_k$, $k = 1 \ldots m$, for each end-effector (denoted by superscript i), we compute the relative displacements from a selected reference position (let us say, position 1), and equate the relative forward kinematics to those relative positions, for all branches simultaneously,

$$\hat{P}^i_{1k} = \underbrace{\prod_{j=1}^{k_i} e^{\frac{\Delta\hat{\theta}^k_j}{2}\mathsf{S}_j}}_{\text{common}} \underbrace{\prod_{j=k_i+1}^{n_i} e^{\frac{\Delta\hat{\theta}^k_{i,j}}{2}\mathsf{S}_{i,j}}}_{\text{branch}} \quad \begin{matrix} i = 1, \ldots, b \\ k = 2, \ldots, m, \end{matrix} \tag{2}$$

where the number of common joints is indicated by k_i and the number of end-effectors, or branches, is indicated by b. Using this notation, each branch i has a total of n_i joints, with k_i common joints. The joint axes at the reference configuration are S_j for the common joints and $\mathsf{S}_{i,j}$ for the joints of branch i.

This yields a total of $6(m-1)b$ independent equations to be simultaneously solved.

4 Solvability of Tree Topologies for Exact Synthesis

We define a kinematic chain as solvable if we can find a positive rational number of positions for which the exact dimensional synthesis yields a finite number of solutions. In the case of serial chains the solvability problem is trivial, and we can always find the maximum number of positions for exact synthesis by equating the

number of independent unknowns to the number of independent equations, for a chain that does not fully define its group of motion. In the most general case, a serial chain with less than six degrees of freedom is solvable.

When dealing with tree topologies, the task sizing must be done so that the system of equations can be solved simultaneously while not overconstraining any of the branches. The tree topology is solvable if we can find a rational number of positions so that we obtain a finite number of solutions for all branches. In order to do so, some conditions need to be defined. Here we present the theory for the most general case; see [6] for cases restricted to subgroups of the group of rigid motion.

The maximum number of positions for the overall system is computed as follows: let $\mathbf{D}_j^e$ be an $e \times 1$ vector containing the joint degrees-of-freedom for each edge of the contracted graph, and $\mathbf{D}_s^e$ be the $e \times 1$ vector containing the number of structural parameters (four per joint in the general case) for each edge of the contracted graph. Denote as $\mathbf{D}_{ee}^n$ the $b \times 1$ vector containing the degrees-of-freedom of the space of each end-effector, and $\mathbf{D}_c^n$ the $b \times 1$ vector with the number of additionally imposed constraints (if any) for each branch. Define the vectors $\mathbf{B}$ as a $b \times 1$ vector of ones corresponding to branches, or end-effectors, and $\mathbf{E}$ as an $e \times 1$ vector of ones for the edges in the graph considered. The maximum number of positions for the overall graph is given by

$$m = \frac{\mathbf{D}_s^e \cdot \mathbf{E} - \mathbf{D}_c^n \cdot \mathbf{B}}{\mathbf{D}_{ee}^n \cdot \mathbf{B} - \mathbf{D}_j^e \cdot \mathbf{E}} + 1. \tag{3}$$

It is necessary that $m \in \mathbb{Q}^+$ for the system to be solvable, but this is not a sufficient condition. In addition, no subgraph starting at the root node and ending at one or more end-effectors can be overdetermined. This phenomenon happens in some topologies with heterogenous branches, such as $2-(1,1,5)$ or $2-(1,5,5)$, to cite a couple of them.

In order to calculate the solvability of each of these subgraphs, use the end-effector path matrix $[\tilde{T}]$ and incidence matrix $[\tilde{B}]$ of the graph [6] to find the vectors $\mathbf{E}_i$ and $\mathbf{B}_i$ containing the edges and branches for a given subgraph i. There are $2^b - 2$ possible subgraphs for any given rooted tree graph, excluding both the full graph and the null graph; only the non-isomorphic graphs need to be considered.

For each subgraph i, calculate the number of positions needed for exact synthesis of the subgraph,

$$m_i = \frac{\mathbf{D}_s^e \cdot \mathbf{E}_i - \mathbf{D}_c^n \cdot \mathbf{B}_i}{\mathbf{D}_{ee}^n \cdot \mathbf{B}_i - \mathbf{D}_j^e \cdot \mathbf{E}_i} + 1. \tag{4}$$

In addition to this, all different and non-isomorphic subgraphs that appear when exchanging the root node with each of the end-effectors need to be considered. This can be proved using the system of design equations.

Let a tree topology have b branches, the first branching happening after k_1 joints, according to the notation in Eq. (2). For each task position k we can isolate the k_1

common joints by post-multiplying by the inverse forward kinematics corresponding to the rest of joints,

$$\left.\begin{aligned} \hat{P}^1_{1k}\left(\prod_{j=k_1+1}^{n_1} e^{\frac{\Delta\hat{\theta}^k_{1,j}}{2}\mathsf{S}_{1,j}}\right)^{-1} &= \prod_{j=1}^{k_1} e^{\frac{\Delta\hat{\theta}^k_{j}}{2}\mathsf{S}_{j}} \\ \vdots & \\ \hat{P}^b_{1k}\left(\prod_{j=k_1+1}^{k_b} e^{\frac{\Delta\hat{\theta}^k_{j}}{2}\mathsf{S}_{j}} \prod_{j=k_b+1}^{n_b} e^{\frac{\Delta\hat{\theta}^k_{b,j}}{2}\mathsf{S}_{b,j}}\right)^{-1} &= \prod_{j=1}^{k_1} e^{\frac{\Delta\hat{\theta}^k_{j}}{2}\mathsf{S}_{j}} \end{aligned}\right\},$$

and subtract the first equation from the rest, to obtain the new system of $6(m-1)(b-1)$ equations,

$$(\hat{P}^1_{1k})^{-1}(\hat{P}^i_{1k}) = \left(\prod_{j=k_1+1}^{n_1} e^{\frac{\Delta\hat{\theta}^k_{1,j}}{2}\mathsf{S}_{1,j}}\right)^{-1}\left(\prod_{j=k_1+1}^{k_i} e^{\frac{\Delta\hat{\theta}^k_{j}}{2}\mathsf{S}_{j}} \prod_{j=k_i+1}^{n_i} e^{\frac{\Delta\hat{\theta}^k_{i,j}}{2}\mathsf{S}_{i,j}}\right), \quad (5)$$
$$i = 2\ldots b, k = 2\ldots m,$$

in which all the unknowns corresponding to the common joints up to the first branching have been eliminated. Also notice that the new task positions correspond to considering the first end-effector as the root node and calculating displacements with respect to a reference frame attached to the new root node. These are the equations for the maximal subgraph not including the previous root node and expressed in this new root node.

We can again repeat the process starting with this new system and eliminating the common joints up to the next branching. At the end of the process, we have explored all non-isomorphic maximal subgraphs created by changing the root node to each of the end-effectors and discarding the previous root node.

As a summary, an overall solution can be imposed only when considering the solvability of all subgraphs that start at the root node and end at end-effectors, including all subgraphs obtained when exchanging the root node with one of the end effectors as described above. In this case, considering m_i as the number of positions for exact synthesis for a subgraph i, with $i \in S$ the set of all possible different end-effector subgraphs up to isomorphism, the topology is solvable if

1. $m \in \mathbb{Q}^+$
2. $m \leq m_i, \quad \forall m_i \in \mathbb{Q}^+,\ i \in \mathrm{S}$

In the case of a subgraph containing c branches and being solvable for $m_i = m$ positions, that subgraph can be solved separately, which eliminates exactly $6c(m-1)$ equations and the same number of unknowns, so that the rest of the graph can be solved a posteriori.

Table 1 Topologies with 1 common joint and 1-jointed branches

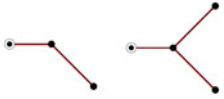
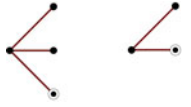
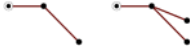
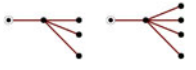
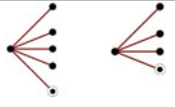
Topology	Graph	Subgraph	Root-change subgraph	Graph solvability
$1-(1,1)$	 $m = 2.33$	 $m = 3$	 $m = 3$	Solvable, $m = 2.33$
$1-(1,1,1)$	 $m = 2.14$	 $m = 3$ $m = 2.33$	 $m = 2.33$ $m = 3$	Solvable, $m = 2.14$
$1-(1,1,1,1)$	 $m = 2.05$	 $m = 3$ $m = 2.33$ $m = 2.14$	 $m = 2.14$ $m = 2.33$ $m = 3$	Solvable, $m = 2.05$
$1-(1,1,1,1,1)$	 $m = 2$	 $m = 3$ $m = 2.33$  $m = 2.14$ $m = 2.05$	 $m = 2.05$ $m = 2.14$  $m = 2.33$ $m = 3$	Solvable, $m = 2$

Table 2 Maximum and minimum number of branches and solvability for topologies with identical branches consisting of revolute joints $p - (q, \ldots, q)$

Number of joints in wrist (p)	Number of joints in each branch (q)	Number of branches (b)		Solvability
		Minimum	Maximum	(S, NS)
1	1	1	5	S
	2/3/4	1	∞	S/S/S
	5	2	∞	S
2	1	1	10	S for $b = 1, 2$
	2/3	1	∞	S/S
	4	2	∞	S
	5	3	∞	S
3	1	1	15	S for $b = 1$
	2	1	∞	S for $b = 1, 2, 3$
	3/4	2	∞	S/S
	5	4	∞	S
4	1	1	20	S for $b = 1$
	2/3	2	∞	S for $b = 2$/S for $b = 2, 3, 4$
	4	3	∞	S
	5	5	∞	S
5	1	2	25	NS
	2/3	2	∞	NS/S for $b = 2$
	4	3	∞	S for $b = 3, 4, 5$
	5	6	∞	S

5 Solvable Tree Topologies with Identical Branches

Two conditions are established for a topology to be a good candidate for exact synthesis with simultaneous tasks: the topology must be solvable, and it must be solvable for $m \geq 2$ positions.

Considering these criteria, the topologies can be classified according to their solvability. As an example of the classification process, Table 1 presents all possible useful topologies for a single branching with 1-jointed branches consisting of revolute joints, together with the detailed analysis of their solvability.

Similar analysis can be performed for increasingly complex topologies, however the complete analysis of useful topologies is not possible, due to the fact that many topologies can have as many branches as desired and still get a useful simultaneous task (with 2 or more task positions per finger). Table 2 shows the candidates for dimensional synthesis for the simplest case of branching, a single branching in which all branches have the same number of joints. The minimum and maximum number of branches have been calculated for having an overall task with a finite number of positions, and greater or equal to 2.

Regarding the solvability of the tree topologies with identical branches, notice that the higher the number of branches, the smaller number of positions m_i obtained:

compare the value of Eq. (4) for the overall graph and the graph obtained after eliminating a single branch, and impose non-solvability. Let the tree topology have p common joints and b branches with q joints each, that is, a $p-(q,\ldots,q)$ topology with b branches, then the graph is not solvable if

$$\frac{4p+4bq}{6b-p-bq} > \frac{4p+4(b-1)q}{6(b-1)-p-(b-1)q} \quad \Longrightarrow \quad \frac{p}{6-q} < b < \frac{p+6-q}{6-q}. \quad (6)$$

It can be exhaustively checked for $0 < p, q < 6$ that there is no positive integer solution for b in this inequality. As a conclusion, all subgraphs starting at the original root node have $m_i \geq m$ when $m_i \in \mathbb{Q}^+$. In order to check for subgraphs when changing the root node, notice that the maximal subgraph obtained is $q-(q,\ldots,q)$ with $b-1$ branches. Following the same reasoning, only this maximal subgraph must be checked and compared to m in each case. The solvability of all useful tree topologies with identical branches presented in the Table 2 has been calculated using this strategy.

6 Conclusions

Dimensional synthesis applied to tree topologies can be used for the design of multi-fingered hands for simultaneous tasks of all fingers. This work focuses on the solvability of tree topologies, that is, on computing which topologies can be synthesized for simultaneous tasks that are meaningful, in which each finger has to reach at least two positions. Previously-stated solvability criteria are proved here and systematically applied to tree topologies with different number of fingers and joints. It turns out that the possible number of fingers is not limited for many topologies, leading to the impossibility of creating a chart of solvable topologies for the general case. It is possible however to classify the topologies with identical fingers, and this classification is presented here. For the general case, a search algorithm is to be developed in future research that can explore solvable topologies for a given task size and for specific requirements on the branches. The results of this research are to be implemented in a general design tool for multi-fingered hands.

Acknowledgments This work is supported by the National Science Foundation under Grant No. 1208385. The content is solely the author's responsibility.

References

1. Chen, I., Yang, G., Kang, I.: Numerical inverse kinematics for modular reconfigurable robtos. J. Robot. Syst. **16**(4), 213–225 (1999)
2. de Jalon, G.J., Bayo, E.: Kinematic and Dynamic Simulation of Multibody Systems: the Real-Time Challenge. Springer, New York (1994)

3. Jain, A.: Graph-theory roots of spatial operators for kinematics and dynamics. In: Proceedings of the 2010 International Conference on Robotics and Automation, pp. 2745–2750. Anchorage, Alaska, USA (2010)
4. Lee, J.J., Tsai, L.: Structural synthesis of multi-fingered hands. ASME J. Mech. Des. **124**, 272–276 (2002)
5. Simo-Serra, E., Moreno-Noguer, F., Perez-Gracia, A.: Design of non-anthropomorphic robotic hands for anthropomorphic tasks. In: Proceedings of the 2011 ASME International Design Engineering Technical Conferences, Washington D.C., USA (2011)
6. Simo-Serra, E., Perez-Gracia, A.: Kinematic synthesis using tree topologies. Mech. Mach. Theory **72**, 94–113 (2014)
7. Simo-Serra, E., Perez-Gracia, A., Moon, H., Robson, N.: Design of multi-fingered robotic hands for finite and infinitesimal tasks using kinematic synthesis. In: Lenarcic, J., Husty, M. (eds.) Recent Advances in Robot Kinematics. Springer, Innsbruck (2012)
8. Stramigioli, S.: Modeling and IPC Control of Interactive Mechanical Systems: a Coordinate-Free Approach. LNCIS, vol. 266. Springer, London (2001)
9. Tischler, C., Samuel, A., Hunt, K.: Kinematic chains for robot hands—1. Orderly number synthesis. Mech. Mach. Theory **30**(8), 1193–1215 (1995)
10. Tsai, L.W.: Mechanism Design: Enumeration of Kinematic Structures According to Function. CRC Press, Boca Raton (2001)

Non-singular Assembly Mode Changing Trajectories in the Workspace for the 3-RPS Parallel Robot

Damien Chablat, Ranjan Jha, Fabrice Rouillier and Guillaume Moroz

Abstract Having non-singular assembly modes changing trajectories for the 3-RPS parallel robot is a well-known feature. The only known solution for defining such trajectory is to encircle a cusp point in the joint space. In this chapter, the aspects and the characteristic surfaces are computed for each operation mode to define the uniqueness of the domains. Thus, we can easily see in the workspace that at least three assembly modes can be reached for each operation mode. To validate this property, the mathematical analysis of the determinant of the Jacobian is done. The image of these trajectories in the joint space is depicted with the curves associated with the cusp points.

Keywords Parallel robot · 3-RPS · Singularity · Operation mode · Aspect · Cylindrical algebraic decomposition

1 Introduction

When designing a robot, the last step is the trajectory planning. The task of the robot is generally defined in the workspace whereas the control loop depends on the joint space parameters. While defining the home pose of the robot, the Cartesian pose

D. Chablat (✉) · R. Jha
Institut de Recherche en Communications et Cybernétique de Nantes (UMR 6597), Nantes, France
e-mail: Damien.Chablat@irccyn.ec-nantes.fr

R. Jha
e-mail: Ranjan.Jha@irccyn.ec-nantes.fr

F. Rouillier
INRIA Paris-Rocquencourt Institut de Mathématiques de Jussieu (UMR 7586), Paris, France
e-mail: Fabrice.Rouillier@inria.fr

G. Moroz
INRIA Nancy-Grand Est, Nancy, France
e-mail: Guillaume.Moroz@inria.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_17,

and the joint values of the actuators are known. If the trajectory planning is done in the workspace by analyzing only the determinant of the Jacobian, we can reach a Cartesian pose different from the home pose but with the same joint value. This feature is called a non-singular assembly mode changing trajectory and stands only for the parallel robot.

For such robots, the inverse and direct kinematic problem (DKP) can have several solutions. To cope up with this problem, the notion of aspects was introduced for the serial robot in [1] and for the parallel robot in [2, 3]. For the serial robots, the aspects are defined as the maximal singularity-free sets in the joint space whereas in case of parallel robots, the aspects are defined as the maximal singularity-free sets in the workspace or the cross-product of the joint space by the workspace. However, there exists robots, referred as cuspidal robots, which are able to change the inverse kinematic solution without passing through a singularity for serial robots or direct kinematic solution without passing through a singularity for parallel robots [4–8]. The uniqueness domains are the connected subsets of the aspects induced by the *characteristic surface*. These notions are defined more precisely in Sects. 2.3 and 2.4.

The chapter elucidates the non-singular assembly mode changing trajectories in the workspace for the 3-RPS parallel robot. In Sect. 2.1 we describe the 3-RPS parallel robot, in Sect. 2.2 we set the related kinematic equations while in Sect. 2.3 we define the aspects for an operation mode. In Sect. 2.4 we analyze the characteristic surfaces for an operation mode, and in Sect. 2.5 we report the non-singular assembly modes changing trajectory between the two basic regions.

2 Kinematics

2.1 Mechanism Under Study

The robot under study is the 3-RPS parallel robot with three degrees of freedom. It been studied by many researchers [8, 9]. It is the assembly of two equilateral triangles (the base and the moving platform) by three identical RPS legs where R is a revolute passive joint, P an prismatic joint and S a passive spherical joint. Thus, the revolute joint is connected to the fixed base and the spherical joint to the mobile platform.

Considering the 3-RPS parallel manipulator, as shown in Fig. 1, the fixed base consists of an equilateral triangle with vertices A_1, A_2 and A_3, and circumradius g. The moving platform is another equilateral triangle with vertices B_1, B_2 and B_3, circumradius h and circumcenter P. The two design parameters g and h are positive numbers. Connecting each of the pairs of vertices of A_i, B_i ($i = 1, 2, 3$) by a limb, a rotational joint lies at A_i and a spherical joint lies at B_i. ρ_i denotes the length of each limb and their adjustment is done through an actuated prismatic joint. Thus we get five parameters, namely g, h, ρ_1, ρ_2 and ρ_3. g and h are the two design parameters

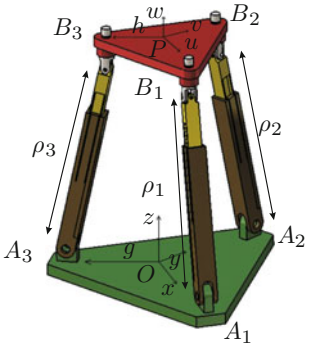


Fig. 1 3-RPS parallel robot

determine the design of the manipulator whereas the joint parameters ρ_1, ρ_2 and ρ_3 determine the motion of the robot. To simplify the equations, we will study a unit robot with $g = h = 1$.

2.2 Kinematic Equations

The transformation from the moving frame to the fixed frame can be described by a position vector $\mathbf{p} = OP$ and a 3×3 rotation matrix $\mathbf{R}$. Let $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{w}$ be the three unit vectors defined along the axes of the moving frame, then the rotation matrix can be expressed in terms of the coordinates of $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{w}$ as:

$$\mathbf{R} = \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} \tag{1}$$

The vertices of the base triangle and mobile platform triangle are

$$\mathbf{A}_1 = \begin{bmatrix} g \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{A}_2 = \begin{bmatrix} -g/2 \\ g\sqrt{3}/2 \\ 0 \end{bmatrix} \quad \mathbf{A}_3 = \begin{bmatrix} -g/2 \\ -g\sqrt{3}/2 \\ 0 \end{bmatrix} \tag{2}$$

$$\mathbf{b}_1 = \begin{bmatrix} h \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{b}_2 = \begin{bmatrix} -h/2 \\ h\sqrt{3}/2 \\ 0 \end{bmatrix} \quad \mathbf{b}_3 = \begin{bmatrix} -h/2 \\ -h\sqrt{3}/2 \\ 0 \end{bmatrix} \tag{3}$$

The coordinates of $\mathbf{b}_i$ with respect to fixed frame reference are obtained by $\mathbf{B}_i = \mathbf{P} + \mathbf{R}\mathbf{b}_i$ for $i = 1, 2, 3$. Also the coordinates of the centre of the mobile platform in the fixed reference is $\mathbf{P} = [x\ y\ z]^T$. The distance constraints yields:

$$||\mathbf{A}_i - \mathbf{B}_i|| = \rho_i^2 \quad \text{with} \quad i = 1, 2, 3 \tag{4}$$

As A_i are revolute joints, the motion of the B_i are constrained in planes. This leads to the three constraint equations:

$$u_y h + y = 0 \tag{5}$$

$$y - u_y h/2 + \sqrt{3} v_y h/2 + \sqrt{3} x - \sqrt{3} u_x h/2 + 3 v_x h/2 = 0 \tag{6}$$

$$y - u_y h/2 - \sqrt{3} v_y h/2 - \sqrt{3} x + \sqrt{3} u_x h/2 + 3 v_x h/2 = 0 \tag{7}$$

Solving with respect to x and y we get:

$$y = -h u_y \tag{8}$$

$$x = h \left(\sqrt{3} u_x - \sqrt{3} v_y - 3 u_y + 3 v_x \right) \sqrt{3}/6 \tag{9}$$

The coefficients of the rotation matrix can be represented by quaternions. The quaternion representation is used for modeling the orientation as quaternions do not suffer from singularities as Euler angles do. The quaternion rotation matrix for the parallel robot is then

$$\mathbf{R} = \begin{bmatrix} 2{q_1}^2 + 2{q_2}^2 - 1 & -2q_1q_4 + 2q_2q_3 & 2q_1q_3 + 2q_2q_4 \\ 2q_1q_4 + 2q_2q_3 & 2{q_1}^2 + 2{q_3}^2 - 1 & -2q_1q_2 + 2q_3q_4 \\ -2q_1q_3 + 2q_2q_4 & 2q_1q_2 + 2q_3q_4 & 2{q_1}^2 + 2{q_4}^2 - 1 \end{bmatrix} \tag{10}$$

with $q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1$. In Eqs. (4), (6), (7), we substitute x, y using relations (8) and (9), and $\mathbf{u}$, $\mathbf{v}$, $\mathbf{w}$ by quaternion expressions using (10). Then (6) and (7) become $q_1 q_4 = 0$. Thus, we have either $q_1 = 0$ or $q_4 = 0$. This property is associated with the notion of operation mode [10].

The notion of operation mode (OM) was introduced in [11] to explain the behavior of the DYMO robot. An operation mode is associated with a specific type of motion. For the DYMO, we have 5 operation modes: translational, rotational, planar (2 types) and mixed motions. In the workspace W, for each motion type, the W_{OM_j} is defined such that

- $W_{OM_j} \subset W$
- $\forall X \in W_{OM_j}$, OM is constant

For a parallel robot with several operating modes, the pose can be defined by fixing the control parameters. For an operation mode OM_j, if we have a single inverse kinematic solution, we can then define an application that maps $\mathbf{X}$ onto $\mathbf{q}$:

$$g_j(\mathbf{X}) = \mathbf{q} \tag{11}$$

Then, the images in W_{OM_j} of a posture $\mathbf{q}$ in the joint space Q are defined by:

$$g_j^{-1}(\mathbf{q}) = \mathbf{X} \mid (\mathbf{X}, \mathbf{q}) \in OM_j \tag{12}$$

where g_j^{-1} is the direct kinematic problem restricted to the operation mode j. Differentiating with respect to time the constraint equations leads to the velocity model:

$$\mathbf{A}\dot{\mathbf{t}} + \mathbf{B}\dot{\mathbf{q}} = 0 \tag{13}$$

where $\mathbf{A}$ and $\mathbf{B}$ are the parallel and serial Jacobian matrices respectively, $\dot{\mathbf{t}}$ is the velocity of P and $\dot{\mathbf{q}}$ is the joints velocity. The parallel singularities occur whenever $\det(\mathbf{A}) = 0$. Let OM_1 (reps. OM_2) be the operation mode where $q_1 = 0$ (reps. $q_4 = 0$). Then $\mathscr{S}_{OM_1}$ and $\mathscr{S}_{OM_2}$ are the loci of the parallel singularities and are characterized by:

$$\begin{aligned}\mathscr{S}_{OM_1} : q_4(&8q_2q_3^2q_4^6 + 2q_2q_4^8 - 64zq_3^6q_4 - 96zq_3^4q_4^3 - 36zq_3^2q_4^5 - 6zq_4^7 \\ &- 24z^2q_2q_3^2q_4^2 - 6z^2q_2q_4^4 - 32q_2q_3^2q_4^4 - 10q_2q_4^6 + 2z^3q_4^3 + 96zq_3^4q_4 \\ &+ 72zq_3^2q_4^3 + 23zq_4^5 + 16z^2q_2q_3^2 + 10z^2q_2q_4^2 + 8q_2q_4^4 - z^3q_4 - 36zq_3^2q_4 \\ &- 21zq_4^3 - 4z^2q_2 + 4zq_4) = 0\end{aligned} \tag{14}$$

$$\begin{aligned}\mathscr{S}_{OM_2} : q_1^2(&6q_1^7q_3 + 8q_1^5q_3^3 - 2zq_1^6 + 36zq_1^4q_3^2 + 96zq_1^2q_3^4 + 64zq_3^6 \\ &- 18z^2q_1^3q_3 - 24z^2q_1q_3^3 - 18q_1^5q_3 - 16q_1^3q_3^3 + 2z^3q_1^2 + 3zq_1{}^4 - 72zq_1^2q_3^2 \\ &- 96zq_3^4 + 18z^2q_1q_3 + 12q_1^3q_3 - z^3 + 3zq_1^2 + 36zq_3^2 - 4z) = 0\end{aligned} \tag{15}$$

The serial singularities occur whenever $\rho_1\rho_2\rho_3 = 0$. The common coordinates for both operation modes are z, q_2 and q_3. Due to the redundancy of the quaternion representation, there exists two triplets defined by these three coordinates that represent the same pose in the same operation mode. To overcome this problem, we set $q_1 > 0$ and $q_4 > 0$. We can then depict a slice of this hypersurface by fixing one parameter as shown in Fig. 2.

2.3 Aspect for an Operation Mode

In [2], the notion of aspect is defined for parallel robots with only one inverse kinematic solution. An aspect WA_i is a maximal singularity free set defined such that:

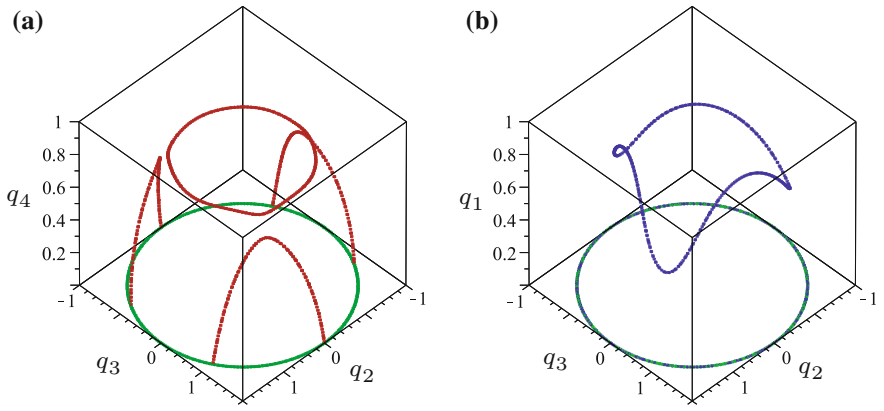


Fig. 2 Singularity curves for $z = 3$, $q_1 = 0$ (**a**) and $q_4 = 0$ (**b**)

- $WA_i \subset W$
- WA_i is connected
- $\forall X \in WA_i$, $\det(\mathbf{A}) \neq 0$ and $\det(\mathbf{B}) \neq 0$

This notion is now extended for a parallel robot with several operation modes such that:

- $WA_{ij} \subset W_{OM_j}$
- WA_{ij} is connected
- $\forall X \in WA_{ij}$, $\det(\mathbf{A}) \neq 0$ and $\det(\mathbf{B}) \neq 0$

In other words, an aspect WA_{ij} is the maximum connected region without any singularity of the OM_j. The analysis of the workspace is done in the projection space (z, q_2, q_3), and shows the existence of four aspects as shown in Fig. 3. However, no further analysis is done to prove this feature in the four dimension space. As there are several solutions for the DKP in the same aspect, non-singular assembly mode trajectories are possible. The cylindrical algebraic decomposition (CAD) implemented in the SIROPA library has been used to decompose an aspect into a set of cells where algebraic equations define its boundaries [12]. The CAD provides a formal decomposition of the parameter space in cells where the polynomials $\det(\mathbf{A})$ and $\det(\mathbf{B})$ have a constant sign [13] and the number of solutions for the DKP is constant.

2.4 Characteristic Surfaces for an Operation Mode

The notion of *characteristic surface* was introduced in [14] to define the uniqueness domains for serial Cuspidal robots. This definition was extended to parallel robots with one inverse kinematic solution in [2] and with several inverse kinematic solutions in [7]. In this chapter, we introduce this notion for a parallel robot with several operating modes.

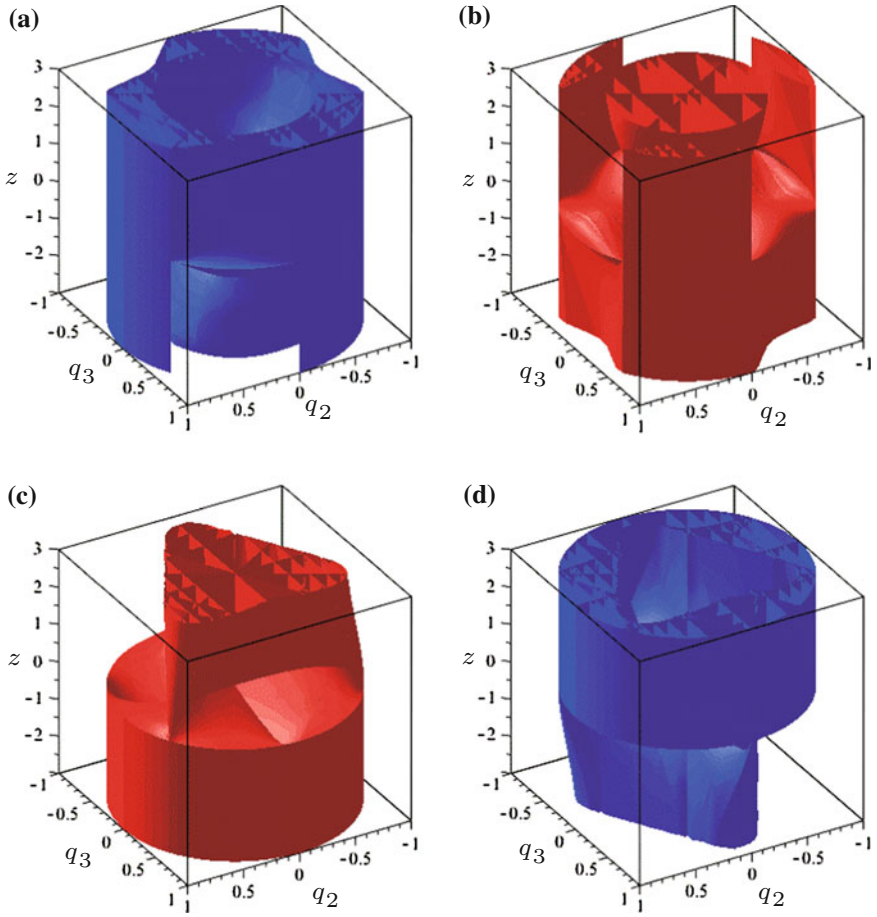


Fig. 3 Aspects for OM_1 with $\det(\mathbf{A}) < 0$ (**a**) and $\det(\mathbf{A}) > 0$ (**b**) and aspects for OM_2 with $\det(\mathbf{A}) < 0$ (**c**) and $\det(\mathbf{A}) > 0$ (**d**)

Let WA_{ij} be one aspect for the operation mode j. The characteristic surfaces, denoted by $\mathscr{S}_C(WA_{ij})$, are defined as the preimage in WA_{ij} of the boundary $\overline{WA}_{ij}$ of WA_{ij}.

$$\mathscr{S}_C(WA_{ij}) = g_j^{-1}\left(g(\overline{WA}_{ij})\right) \cap WA_{ij} \tag{16}$$

These characteristic surfaces are the images in the workspace of the singularity surfaces. By using the singularity and characteristic surfaces, we can compute the *basic regions* as defined in [2]. The joint space is divided by the singularity surfaces in regions where the number of solutions for the DKP is constant. We also name these regions the *basic components* as in [2]. For each operation mode, we find regions

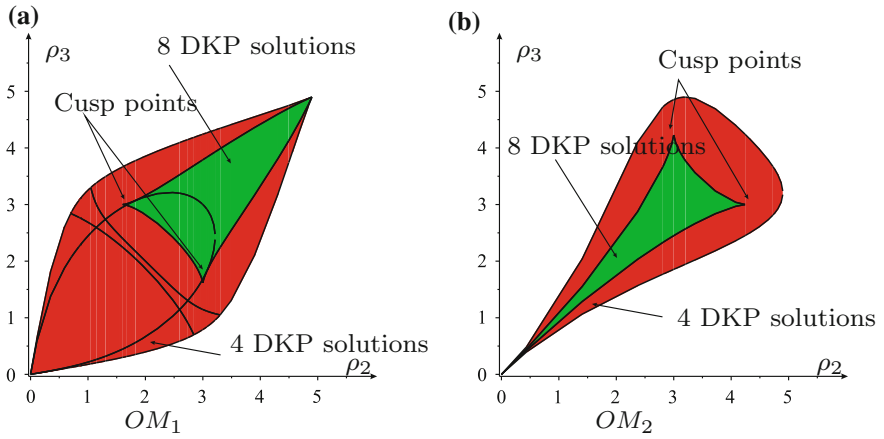


Fig. 4 Slice of the joint space for $\rho_1 = 3$ for OM_1 (**a**) and OM_2 (**b**)

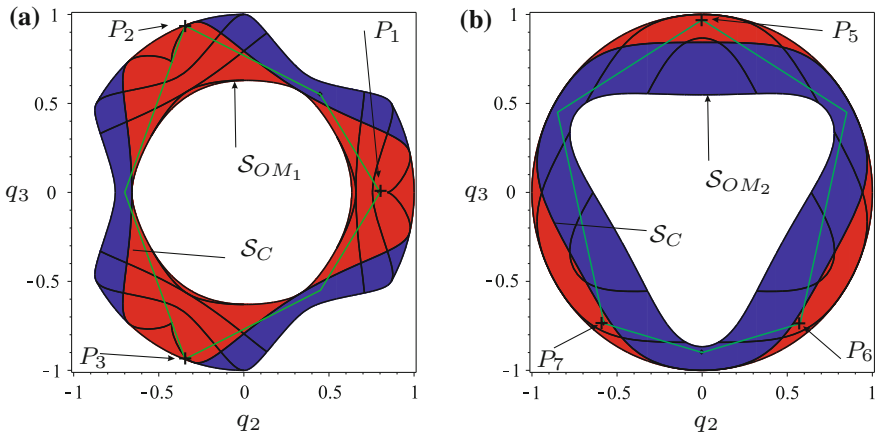


Fig. 5 Slice of an aspect for $z = 3$ and $\det(\mathbf{A}) > 0$ for OM_1 (**a**) and OM_2 (**b**) with in *blue* (resp. in *red*) a basic region coming from a basic component with four DKP (resp. eight)

where the DKP admits four (in red) or eight (in green) solutions, as it is depicted in Fig. 4. We can also notice in Fig. 4 the existence of cusp points.

2.5 Non-singular Assembly Mode Changing Trajectories

Due to the lack of space and for pedagogical purpose, we only report a slice of the workspace. Letting $z = 3$, the basic regions are computed by using the cylindrical algebraic decomposition for a given aspect. Figure 5 shows the three basic regions'

Table 1 Solutions of the DKP for $det(\mathbf{A}) > \mathbf{0}$

OM_1					OM_2				
$\rho_1 = 3.90, \rho_2 = 3.24, \rho_3 = 3.24$					$\rho_1 = 3.79, \rho_2 = 3.24, \rho_3 = 3.24$				
P	z	q_2	q_3	q_4	P	z	q_1	q_2	q_3
P_1	3.01	−0.34	−0.94	0.06	P_5	3.04	0.35	−0.58	−0.74
P_2	3.01	−0.34	0.94	0.06	P_6	3.04	0.35	0.586	−0.74
P_3	3	0.85	0.0	0.53	P_7	3	0.24	0.0	0.97
P_4	−2.88	−0.35	0.0	0.93	P_8	−3.42	0.98	0.0	0.19

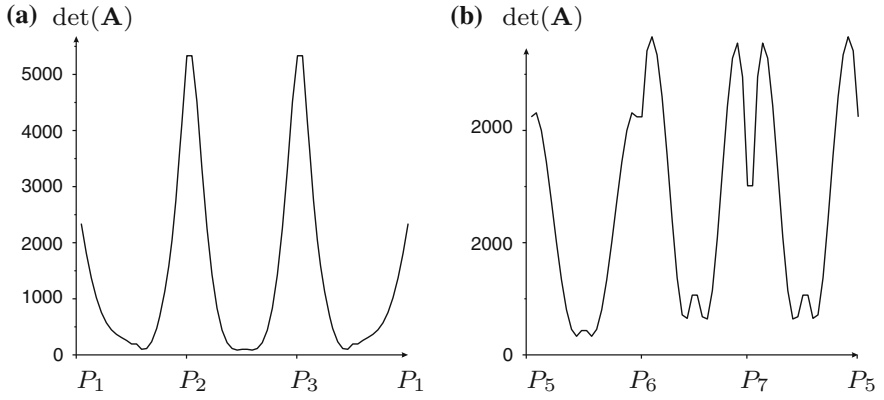


Fig. 6 Variation of det(**A**) along trajectory P_1, P_2, P_3 for OM_1 (**a**) and P_5, P_6, P_7 for OM_2 (**b**)

images of basic components with 8 solutions for the DKP and a single basic region's image of a basic components with 4 solutions for the DKP connects these three previous basic regions. Table 1 presents the roots of the DKP for $\det(\mathbf{A}) > \mathbf{0}$ for a joint position in each operation mode. For each of them, we find out that three roots have their z coordinate close to 3. A non-singular assembly mode changing trajectory can be obtained between three basic regions coming from eight solutions to the DKP. Due to symmetrical properties, there are also three roots of the DKP for $\det(\mathbf{A}) < \mathbf{0}$ with $z = -3$. For OM_1, we construct a path between P_1, P_2, P_3 and for OM_2 between P_5, P_6 and P_7. When a straight line between two poses cross a singularity, we add an intermediate point as shown in Fig. 5. The connections between the basic regions depicted in red are the projections of the cusp points in the workspace, i.e. the tangent between the singularity surface and the characteristic surface [12].

The variation of the $\det(\mathbf{A})$ is plotted in the Fig. 6 and shows the existence of a non-singular assembly mode changing trajectory. The image of this trajectory in the joint space is illustrated in the Fig. 7. The projection of the cyclic trajectory defined by (P_1, P_2, P_3, P_1) (resp. (P_5, P_6, P_7, P_5)) onto the joint space encloses three curves of cusps. This behavior is similar to that of the 3-RPR robot described in [15] or the 3-RPS robot in [8]. The path to connect the fourth solutions is not presented in this chapter. The method introduced in [16] is used to compute the cusp curves.

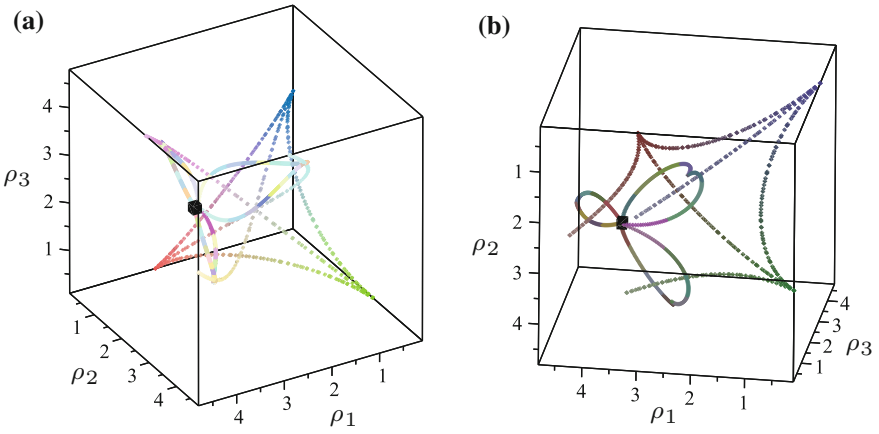


Fig. 7 Projection in Q of the trajectories with the cusp curves for OM_1 (**a**) and OM_2 (**b**)

3 Conclusions

This chapter presents a study of the joint space and workspace of the 3-RPS parallel robot and shows the existence of non-singular assembly mode changing trajectories. First, we have shown that each of the two operation modes is divided into two aspects, which is a necessary condition for non-singular assembly mode changing trajectories. Moreover, it turns out that this mechanism has a maximum of 16 real solutions to the direct kinematic problem, eight for each operation mode. Then, by computing the characteristic surfaces, we have shown that we can describe the basic regions for each operation mode. We construct a path going through several basic regions which are images of the same basic component with 8 solutions for the DKP. The analysis of the determinant of Jacobian shows that a non-singular assembly mode change exists for each motion type.

References

1. Borrel, P.: A study of manipulator inverse kinematic solutions with application to trajectory planning and workspace determination. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1180–1185 (1986)
2. Wenger, Ph., Chablat, D.: Definition sets for the direct kinematics of parallel manipulators. In: 8th International Conference in Advanced Robotics, pp. 859–864 (1997)
3. Chablat, D., Wenger Ph.: Working modes and aspects in fully-parallel manipulator. In: Proceeding IEEE International Conference on Robotics and Automation, pp. 1964–1969, May (1998)
4. Innocenti, C., Parenti-Castelli, V.: Singularity-free evolution from one configuration to another in serial and fully-parallel manipulators. Robotics, Spatial Mechanisms and Mechanical Systems, ASME (1992)

5. Macho, E., Altuzarra, O., Pinto, C., Hernandez, A.: Transitions between multiple solutions of the direct kinematic problem. In: Lenarčič, J., Wenger, P. (eds.) Advances in Robot Kinematics: Analysis and Design, pp. 301–310. Springer Netherlands (2008)
6. Caro, S., Wenger, P., Chablat, D.: Non-singular assembly mode changing trajectories of a 6-DOF parallel robot. In: Proceedings of the ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference IDETC/CIE 2012, Chicago, Illinois, USA, 12–15 August 2012
7. Chablat, D., Wenger, P.: Séparation des solutions aux modèles géométriques direct et inverse pour les manipulateurs pleinement parallèles. Mech. Mach. Theory **36**(6), 763–783 (2001)
8. Husty, M., Schadlbauer, J., Caro, S., Wenger, P.: Non-singular assembly mode change of 3-RPS manipulators. In: Proceedings of CK2013, International Workshop on Computational Kinematics, Barcelone (2013)
9. Babu, S.R., Raju, V.R., Ramji, K.: Design for optimal performance of 3-RPS parallel manipulator using evolutionary algorithms. T. Can. Soc. Mech. Eng. **37**(2), 135–160 (2013)
10. Schadlbauer, J., Nurahmi, L., Husty, M., Wenger, P., Caro, S.: Operation modes in lower-mobility parallel manipulators. Interdisciplinary Applications in Kinematics (2013)
11. Zlatanov, D., Bonev, I.A., Gosselin, C.M.: Constraint singularities as C-space singularities. In: Lenarčič, J., Thomas, F. (eds.) Advances in Robot Kinematics, pp. 183–192. Springer Netherlands (2002)
12. Chablat, D., Moroz, G., Wenger, P.: Uniqueness domains and non singular assembly mode changing trajectories. In: Proceedings of the IEEE International Conference on Robotics and Automation, May (2011)
13. Collins, G.E.: Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition. Springer, New York (1975)
14. Wenger, P.: A new general formalism for the kinematic analysis of all nonredundant manipulators. In: Proceedings of the IEEE Robotics and Automation, pp. 442–447 (1992)
15. Zein, M., Wenger, P., Chablat, D.: Non-singular assembly-mode changing motions for 3-RPR parallel manipulators. Mech. Mach. Theory **43**(4), 480–490 (2008)
16. Moroz, G., Rouillier, F., Chablat, D., Wenger, P.: On the determination of cusp points of 3-RPR parallel manipulators. Mech. Mach. Theory **45**(11), 1555–1567 (2010)

Influence of Spring Characteristics on the Behavior of Tensegrity Mechanisms

Quentin Boehler, Marc Vedrines, Salih Abdelaziz, Philippe Poignet and Pierre Renaud

Abstract There is today a growing interest for tensegrity mechanisms. Their analysis is however challenging because of their self-stress state. The most popular tensegrity mechanisms use linear springs as tensioned elements. Their synthesis for given user requirements is an open issue. In this article, we propose as a first step to better understand the influence of the spring characteristics, that constitute important design parameters. The influence of spring free length is in particular assessed, considering two planar tensegrity mechanisms. Impact of the spring selection on the workspace, the stiffness and the actuation requirements is observed. The simulation results outline that using nonzero free length springs can be of interest, and conclusions are given on further steps towards a synthesis method.

Keywords Tensegrity mechanisms · Stiffness computation · Workspace estimation

1 Introduction and Scope of the Study

Tensegrity systems can be defined [5] as systems in stable self-equilibrated state comprising a discontinuous set of compressed components, i.e. struts, inside a continuum of tensioned components, either cables or springs. Thanks to the self-stress state, prestress can be imposed for a same topology in order to modify the level of internal forces. A high stiffness-to-mass ratio can thus be reached. Moreover, since the components are axially loaded, they can be very light and so does the system [7]. Tensegrity mechanisms use actuators to modify their configuration [3, 6]. The tensioned components are then generally linear springs, in order to

Q. Boehler (✉) · M. Vedrines · P. Renaud
ICube, Strasbourg, France
e-mail: quentin.boehler@icube.unistra.fr

S. Abdelaziz · P. Poignet
LIRMM, Montpellier, France

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_18,

systematically reach equilibrated configurations without tension loss. They have recently received a lot of attention [8, 9, 11], and could be interesting solutions for instance in surgical contexts, where lightweight devices with large workspaces are required [1].

The analysis of tensegrity mechanisms is delicate compared to rigid-body mechanisms. The determination of a mechanism configuration requires for instance to take into account joint variables and external loads while respecting self-equilibrium state conditions. The corresponding static model is obtained using so-called form finding methods [10]. Up to now, related works have been focused on the analysis of tensegrity mechanisms, but not yet on their synthesis. Even the relationship between the design parameters and the mechanism behavior, and the estimation of actuator performances for given specifications remains to be investigated. In this chapter, we wish therefore to analyze the sensitivity of the tensegrity characteristics to their design parameters.

The spring characteristics strongly influence the mechanism behavior. We therefore concentrate our efforts on the parameters related to the linear springs used as tensioned components. The influence of spring stiffness is here considered, and more importantly the free length of the springs. To our knowledge, estimating the influence of nonzero free lengths on mechanism performances has not yet been covered in the literature.

As a result, we introduce in this chapter an analysis of spring stiffness and free length on the behavior of tensegrity mechanisms. Two planar tensegrity mechanisms of Snelson-cross type [2, 3] are considered, that respectively exhibit 1 and 2-DOF. In Sect. 2, the 1-DOF mechanism is considered. The mechanism is simple enough to allow a detailed analysis of free length influence on mechanism workspace, stiffness and criteria related to actuator requirements, namely actuator stroke, force and energy consumption. In Sect. 3, a 2-DOF mechanism is analyzed. This mechanism can be used as a tool holder in a surgical task [1]. Conclusions are finally given with emphasis on further steps towards a generalization for other tensegrity mechanisms.

2 Sensitivity Analysis of a 1-DOF Tensegrity Mechanism

2.1 Mechanism Description

The 1-DOF tensegrity mechanism, based on the Snelson-cross planar tensegrity, is represented in Fig. 1. It consists in four nodes $\{A, B, C, D\}$. Two rigid struts of length L connect respectively A with C and B with D. Three linear springs are installed on the outside of the mechanism, with same stiffness k and free length l_0. An actuator is used to modify the distance $\|AB\| = \rho$, with the node A anchored to the base and B constrained to move in a horizontal direction. In such a situation, the mechanism exhibits 1 DOF, with nodes C and D that have equal vertical displacements. The output variable is defined as y, the Y-coordinate of C and D, and ρ is the

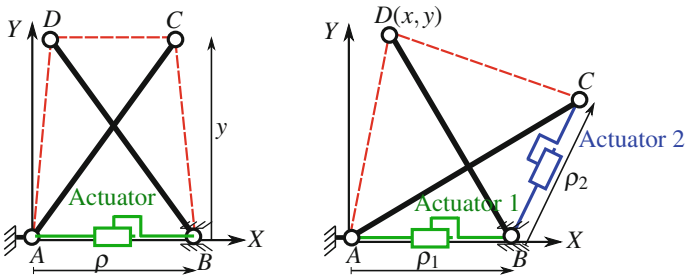


Fig. 1 Kinematic schemes of the 1-DOF (*left*) and 2-DOF (*right*) tensegrity mechanisms. In *black plain lines*, rigid struts and in *red dotted lines*, linear springs

input variable. For sake of simplicity, a so-called neutral configuration is defined as $||AD|| = ||BC|| = ||CD|| = \rho = 100$ mm with perpendicular struts. It implies that the struts have a length $L = 100\sqrt{2} = 141$ mm.

2.2 Modeling and Analysis Criteria

The mechanism modeling consists in deriving expressions of the direct and inverse static models. The output variable y is obtained for a given value of ρ by solving the direct static problem. An energy method, as described in [10], is here considered. The mechanism configuration is then obtained by determining the local minimum of the potential energy U stored in the mechanism springs. In our context, U can be easily computed as

$$U = \frac{1}{2}k(2(l_1 - l_0)^2 + (l_2 - l_0)^2) \tag{1}$$

with $l_1 = ||AD|| = ||BC||$ and $l_2 = ||CD||$. The lengths l_1 and l_2 can be expressed using simple geometrical relationships as functions of the variables ρ and y and the length L. In the following, we are only interested in positive solutions of y. The analytic expression of $y = f(\rho)$ is obtained by solving $\frac{\partial U}{\partial y} = 0$ for a given ρ, using a computer algebra system (Mupad, The Mathworks Inc.). Only one admissible solution is found with the simulated conditions where $\rho \in [70, 130]$ mm and $l_0 \in [20, 80]$ mm. The inverse static model that expresses ρ as a function of y is difficult to obtain if nonzero free length of the springs are considered. As noticed previously in [4], the use of nonzero free length springs in a tensegrity mechanism significantly increases its analysis complexity. In our situation, no analytical expression could be found. A numerical approach is then adopted, with a Levenberg-Marquardt algorithm to compute the ρ value for a given y as the solution of $(y - f(\rho))^2 = 0$. The tensegrity mechanism behavior is assessed by using two types of criteria. The first type corresponds to usual mechanism properties, namely the workspace and the stiffness. The workspace is the range of y values for a given range of ρ. The stiffness

of the mechanism is defined as $K_y = \frac{\delta F_y}{\delta u_y}$ with δu_y the infinitesimal displacement of nodes C and D along the Y-axis when an infinitesimal external vertical force δF_y is evenly applied on these nodes, while the actuator is locked. As stated in [2], the stiffness K_y can be estimated with $\frac{\partial^2 U}{\partial y^2}$. The other type of criteria characterizes the actuation requirements of the mechanism. For a predefined path followed by the mechanism in quasistatic conditions, we compute

- $\Delta\rho$: the corresponding required actuator stroke
- $\overline{F}$ and F_{max}: respectively the mean and maximum forces delivered by the actuator
- ΔU: the variation of potential energy.

Meanwhile, the mean value $\overline{K_y}$ and the variation ΔK_y of the stiffness are also estimated. In the presented results, the path is defined as symmetric with respect to the neutral configuration, with a 20-mm displacement. It is discretized in 11 steps to compute the analysis criteria.

2.3 Results and Discussion

As it can be observed in Eq. (1), the configuration corresponding to the local minimum of the potential energy U does not depend on the springs stiffness k. The equilibrium configuration is therefore invariant with respect to k. Consequently, modifying the spring stiffness does not modify the mechanism workspace. The mechanism stiffness K_y is proportional to k as it can be observed in Eq. (1). The energy variation ΔU and the actuator force F are also proportional to k. The energy variation corresponds to the energy to be delivered by the actuator. The actuation requirements are therefore linearly dependent on the spring stiffness for a given workspace.

The spring free length l_0 has on the contrary a non-linear influence on the analysis criteria. Figure 2 represents the computation of y and K_y according to l_0 when ρ varies between 70 and 130 mm in 25 steps, for $k = 4$ N/mm. In this example, one may observe that the neutral configuration is not dependent on the spring free length. For a given y, increasing l_0 leads to an increase of K_y while the force delivered by the actuator decreases. This may appear unintuitive, but it is explained by the reconfiguration of the mechanism when l_0 is increased (bottom of Fig. 2). The workspace is on the contrary reduced for the same actuation stroke. For given mechanism specifications, interesting compromises therefore exist for nonzero free lengths.

Figure 3 shows the evolution of the criteria related to the actuator requirements. Increasing l_0 leads to the increase of the actuation stroke $\Delta\rho$ while $\overline{F}$, $F_{\max}$ are reduced, as well as the potential energy variation ΔU. An optimal value of l_0 could be searched for in order to adapt the mechanism and the actuator force/displacement characteristics. The influence of the free length on the stiffness is complex, with an increase of the mean value $\overline{K_y}$ but a reduction of the variation ΔK_y. As a conclusionof

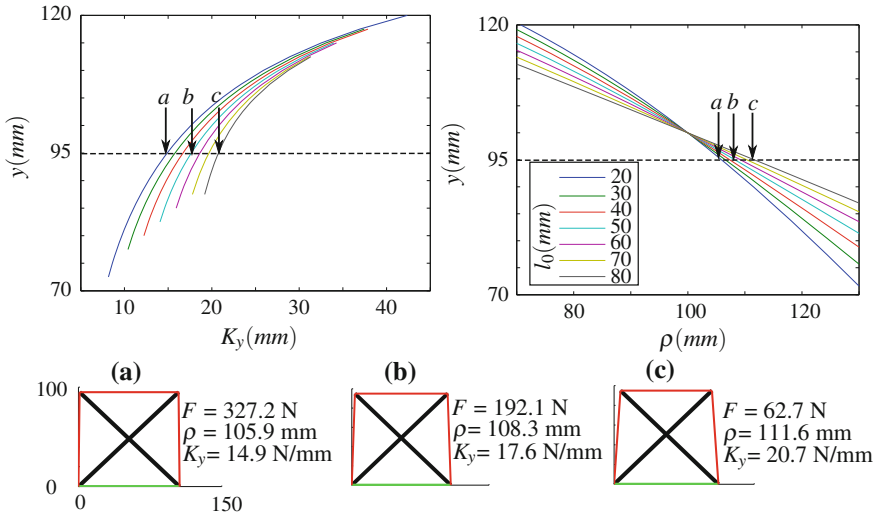


Fig. 2 Relationships between the spring free length l_0 and the stiffness K_y and the workspace. Cases (**a**), (**b**) and (**c**) depict the mechanism at $y = 95$ mm for $l_0 = 20$, 50 and 80 mm respectively

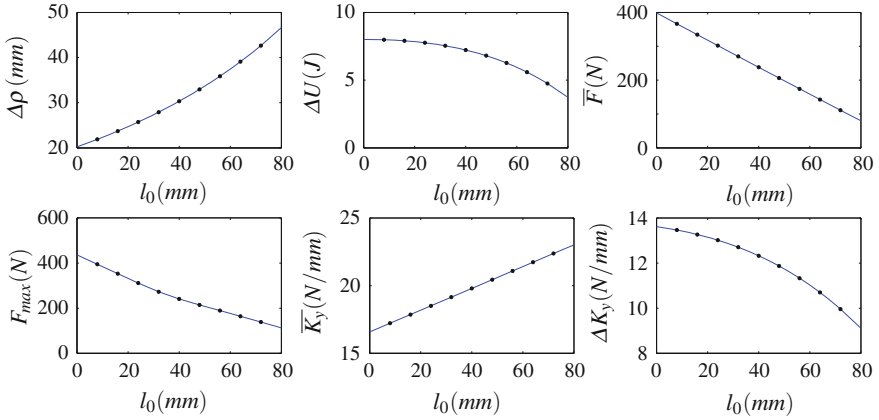


Fig. 3 Evolution of the analysis criteria according to l_0

this case, the analysis suggests first that nonzero free lengths can be of interest for the mechanism performance, and that the choice of l_0 is a compromise between mechanism properties $\overline{K_y}$, ΔK_y and the actuator requirements $\Delta\rho$, $\overline{F}$, $F_{\max}$ and ΔU.

3 Analysis of a 2-DOF Tensegrity Mechanism

3.1 Mechanism Description

A second actuator is integrated between nodes B and C in the previous tensegrity mechanism (Fig. 1). The mechanism becomes a 2-DOF manipulator with two joint variables (ρ_1, ρ_2) and its end-effector in D.

3.2 Modeling and Analysis Criteria

For this 2-DOF mechanism, no analytical direct static model could be obtained if springs have a nonzero free length. Hence, the coordinates of D are determined by minimizing the potential energy U using a Nelder-Mead algorithm. The inverse static model is not implemented since it is not necessary for the evaluation.

Contrary to a rigid-body mechanism, the mechanism configuration does not depend only on (ρ_1, ρ_2), but also on one parameter that can be either the angle $\widehat{ABD}$, the variable x, or y. Therefore $U = U_1(\rho_1, \rho_2, x) = U_2(\rho_1, \rho_2, y)$, and the end-effector stiffnesses K_x and K_y in X- and Y-direction can be computed as follows

$$K_x = \frac{\partial^2 U_1(\rho_1, \rho_2, x)}{\partial x^2} \tag{2}$$

$$K_y = \frac{\partial^2 U_2(\rho_1, \rho_2, y)}{\partial y^2} \tag{3}$$

For this mechanism, analysis criteria are related to the mechanism workspace, stiffness and the actuation requirements:

- A is the area of the reachable workspace
- $\overline{K_x}$ and $\overline{K_y}$ are the mean values of respectively K_x and K_y over the workspace
- ΔK_x and ΔK_y designate the variations of respectively K_x and K_y over the workspace
- $\overline{F}$ and $F_{\max}$ are computed as the mean and maximum forces in each actuator.

The workspace is explored with joint ranges that avoid mechanism singularity and loss of tensegrity configuration. A singularity occurs when $\rho_1 = \rho_2 = L/2$, i.e. BD is vertical. The lower bounds of ρ_1 and ρ_2 are therefore chosen equal to 75 mm. The strut BD must remain in compression to keep the mechanism in a tensegrity configuration. As a result, the upper bound of the joint variables is such that $\rho_1^2 + \rho_2^2 < 5L^2/2$. A 5-mm step for each joint variable is considered for the computation. The area A is computed by extracting the edge of the discrete set of positions obtained after the exploration of the joint space. With such joint ranges, and thanks to the mechanism symmetry, $\overline{F}$ and F_{max} are the same for both actuators.

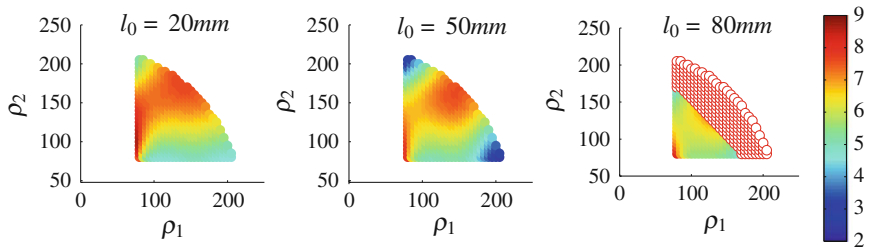


Fig. 4 Variation of the stiffness K_y in N/mm in the joint space for $l_0 = 20$, 50 and 80 mm. In *red circles* unreachable configurations because of tension loss

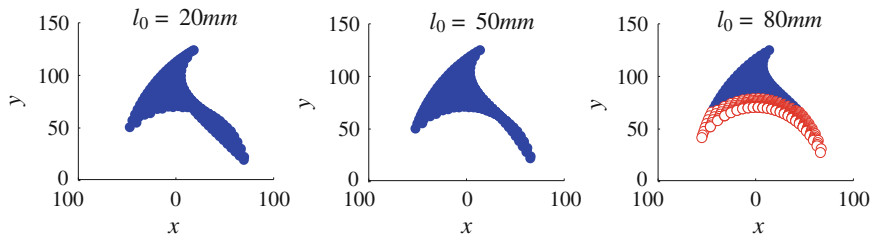


Fig. 5 Reachable workspace for $l_0 = 20$, 50 and 80 mm. *Blue dots* reachable positions/*red circles* unreachable positions because of tension loss

3.3 Results and Discussion

Similarly to Sect. 2, the influence of the spring stiffness k on the analysis criteria is simple. It does not affect the equilibrium configurations, and linearly impacts the required actuators forces and the stiffnesses.

The variation of K_y is represented in Fig. 4, for a spring stiffness $k = 3$ N/mm. For such a mechanism, an end-effector position can sometimes be obtained with two different configurations and hence two different stiffness values. The map is therefore plotted in the joint space. Figure 5 depicts the corresponding reachable workspace. The l_0 value modifies the shape of the workspace boundaries, and the workspace area. In addition, tension losses can occur that limit the workspace, as it can be observed between the middle and right plots of Fig. 5. Further analysis shows that the first tension loss takes place when D is at the middle of AC, and that $l_0 = l_{0\,loss} = L/2 \simeq 70$ mm. The l_0 value also affects the mean value and the variation of the stiffness over the workspace (left and middle plots of Fig. 4).

Figure 6 shows the evolution of the analysis criteria according to l_0. This evolution of the criteria is notably affected by choosing $l_0 > l_{0\,loss}$, with in particular a fast reduction of the workspace area A. If $l_0 < l_{0\,loss}$, $\overline{K_x}$, $\overline{K_y}$, $\overline{F}$ and F_{max} decrease when l_0 increases. Although ΔK_x is also decreasing monotonously with respect to l_0, the evolution of ΔK_y is more complex (as it can be seen on Fig. 4). The lowest value of ΔK_x is indeed obtained for a nonzero value of l_0. The evolution of the

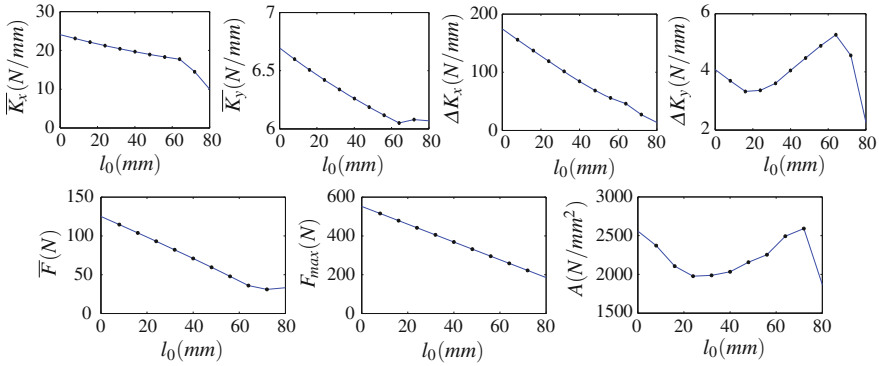


Fig. 6 Evolution of the analysis criteria according to l_0

workspace area A is also non uniform. Increasing l_0 does not necessarily decrease the size of the workspace.

The criteria sensitivity to the free length value is different from the one observed for the 1-DOF mechanism. Increasing l_0 can lead to tension losses, which is quite logical, and affects stiffness and workspace properties. However, the free length can be used to lower the required actuator forces and an adequate choice of l_0 can help increasing the workspace size.

4 Conclusions

In this chapter, influence of the spring characteristics on tensegrity mechanisms was considered. Using analysis criteria related to the mechanism properties and the actuator requirements, two planar mechanisms were analyzed. The first major remark is that increasing the free length of the springs can have a beneficial impact on the mechanism or the selection of an adequate mechanism. Whereas the spring stiffness influence can be easily assessed, relationships between the selected criteria and the free length is much more complex. The determination of the spring free length during a mechanism synthesis will need to be performed simultaneously with other design parameters related to the mechanism geometry. The observed significant nonlinearities of the sensitivity curves outline also that the selection of the optimization strategy will be delicate, and will be investigated as the next step of this work as well as the generalization for other tensegrity mechanisms.

References

1. Abdelaziz, S., Esteveny, L., Renaud, P., Bayle, B., Barbe, L., De Mathelin, M., Gangi, A.: Design considerations for a novel MRI compatible manipulator for prostate cryoablation. Int. J. Comp. Assist. Rad. Surg. **6**(6), 811–819 (2011)
2. Arsenault, M., Gosselin, C.: Kinematic, static, and dynamic analysis of a planar one-degree-of-freedom tensegrity mechanism. ASME J. Mech. Des. **127**(6), 1152–1160 (2005)
3. Arsenault, M., Gosselin, C.: Kinematic, static and dynamic analysis of a planar 2-DOF tensegrity mechanism. Mech. Mach. Theory **41**(9), 1072–1089 (2006)
4. Crane, C., 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, pp. 419–427. Springer (2008)
5. Motro, R.: Tensegrity: Structural Systems for the Future. Elsevier, New York (2003)
6. Shekarforoush, S., Eghtesad, M., Farid, M.: Kinematic and static analyses of statically balanced spatial tensegrity mechanism with active compliant components. J. Intell. Robot Syst. **71**(3–4), 287–302 (2013)
7. Skelton, R., Oliveira, M.d.: Tensegrity Systems. Springer, New York (2009)
8. Skelton, R., Sultan, C.: Controllable tensegrity: a new class of smart structures. In: Proceedings of the SPIE, Smart, Structures and Materials, 3039, pp. 166–177 (1997)
9. Sultan, C., Corless, M., Skelton, R.: Tensegrity flight simulator. J. Guid. Control Dynam. **23**(6), 1055–1064 (2000)
10. Tibert, A., Pellegrino, S.: Review of form-finding methods for tensegrity structures. Int. J. Space Struct. **18**(4), 209–223 (2003)
11. Tran, T., Crane, C., Duffy, J.: The reverse displacement analysis of a tensegrity based parallel mechanism. In: Proceedings of World Automotive Congress, vol. 14, pp. 637–643 (2002)

Human Motion Kinematics Assessment Using Wearable Sensors

Sebastjan Šlajpah, Roman Kamnik and Marko Munih

Abstract This chapter presents a novel sensory fusion algorithm for assessing the orientation of human body segments in long-term human walking based on signals from wearable sensors. The basic idea is to constantly fuse the measured segment's angular velocity and linear acceleration via known kinematic relations between segments. The wearable sensory system incorporates seven inertial and magnetic measurement units and two instrumented shoe insoles. The proposed system was experimentally validated in a long-term walking on a polygon simulating different everyday activities. Results show accurate joint angle measurements (error median below 5°) with no observed drift over time.

Keywords Sensory fusion · Kinematic model · Extended Kalman filter · Inertial measuring units · Long-term walking

1 Introduction

Wearable robotics and monitoring in sports are two recently developed fields where the latest progress in sensory technology contributed essentially. Miniature sensors utilizing MEMS components are convenient to be worn by the user or to be implemented on a wearable robotic structure, such as active prostheses, orthoses, and exoskeletons [7, 10, 12]. The acquired information on kinematic and kinetic

S. Šlajpah (✉) · R. Kamnik · M. Munih
University of Ljubljana, Faculty of Electrical Engineering, Tržaška cesta 25,
1000 Ljubljana, Slovenia
e-mail: sebastjan.slajpah@fe.uni-lj.si

R. Kamnik
e-mail: roman.kamnik@fe.uni-lj.si

M. Munih
e-mail: marko.munih@fe.uni-lj.si

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_19,

parameters is used for providing feedback to the user [2, 3] or for closing the loop in a robot controller [6, 10].

Level-ground walking is considered a basic manoeuver of human locomotion and has as such been a subject of numerous studies in recent decades. At first, the uniaxial gyroscopes, attached to body segments, were used to assess joint angles with a simple integration method [9, 14]. The method is prone to output drift over long-term measurements since it integrates superimposed noise over time. A solution was proposed as a system reset during static conditions and re-initialization of the outputs with regards to a known reference value [14]. Difficulty with this approach is that during fast, uninterrupted walking it is near to impossible to accomplish accurate re-initialization.

The Kalman filter was introduced for assessing the orientation of individual segment by fusing data from gyroscope and accelerometer [8]. The orientation estimate obtained by integration of the 3D angular velocity is continuously corrected by using inclination estimate obtained from measured acceleration of the segment. The results show that due to the heading drift the presented method is not suitable for long-term measurements [4, 13].

To measure three dimensional orientation, supplementary reference axis in addition to gravity needs to be introduced. Commonly, magnetometers have been incorporated into systems using magnetic field as an additional axis [1, 11]. In [1] authors implemented the Unscented Kalman filter for orientation estimation. Estimate obtained by integrating angular velocity is continuously corrected by the orientation estimated via accelerometer and magnetometer. During fast movement this method yields larger errors in orientation since it is not differentiating between gravitational and dynamical components of measured acceleration. Roetenberg et al. [11] presented a complementary Kalman filter which is based on estimation of errors between the orientation obtained by integration of gyroscope data and one obtained from accelerometer and magnetometer.

The aim of this chapter is to present a novel approach for long-term kinematic parameters assessment with a combined use of wearable inertial, magnetic, and insole sensors. The approach is based on fusing measured accelerations and angular velocities with calculated accelerations based on human body kinematic relations. The accuracy and long-term reliability of proposed approach are experimentally evaluated in walking on a polygon.

2 Kinematic Relation in Serial Kinematic Chain

The sensory fusion for estimation of segment orientations in long-term and dynamic motion proposed in this chapter is built upon an extended Kalman filter (EKF) algorithm [5]. The concept is based on a kinematic relation which states that on a rigid body the acceleration of any point can be determined if the angular velocity, angular acceleration and linear acceleration of another point of the body are known.

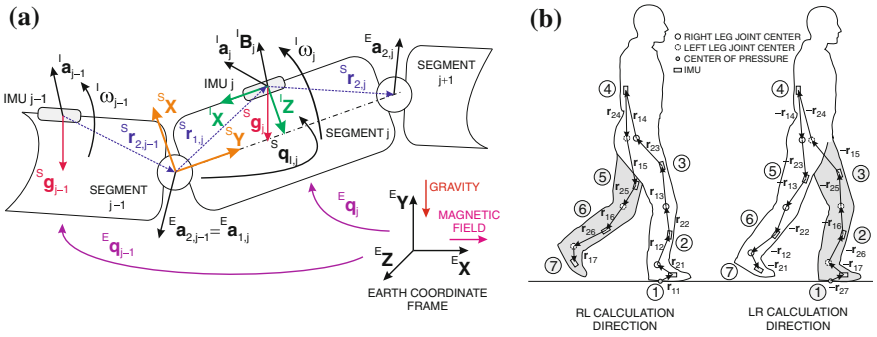


Fig. 1 **a** A serial kinematic chain of segments with marked connections between two neighboring segments and IMU's measured quantities. Quantities denoted with subscripts $_j$ refer to the j-th segment. Quantities marked with left superscripts $^{\mathrm{I}}$, $^{\mathrm{S}}$, and $^{\mathrm{E}}$ are expressed in the coordinate frame of the IMU, segment, and Earth, respectively. **b** *RL* Right–left and *LR* left–right sequence of orientation calculations, denoted with *numbers in circles*, when the right or the left foot is in stance phase, respectively

A serial kinematic chain of rigid bodies mimicking human lower extremities with one wearable inertial and magnetic measurement unit (IMU) placed on each segment is presented in Fig. 1a. Quantities marked with subscript j correspond to the j-th segment. Vectors ${}^{\mathrm{I}}\boldsymbol{\omega}$, ${}^{\mathrm{I}}\boldsymbol{a}$ and ${}^{\mathrm{I}}\boldsymbol{B}$ refer to angular velocity, linear acceleration and magnetic field measured by IMU sensor, respectively. Vector ${}^{\mathrm{E}}\boldsymbol{a}_1$ denotes the linear acceleration of the center point of the joint connecting the j-th segment with the previous $j-1$ segment. Vector ${}^{\mathrm{E}}\boldsymbol{a}_2$ denotes the linear acceleration of the center point of the joint between the j-th segment with the following $j+1$ segment. Accelerations ${}^{\mathrm{E}}\boldsymbol{a}_1$ and ${}^{\mathrm{E}}\boldsymbol{a}_2$ are expressed in the Earth's coordinate frame which is defined by the gravity vector (${}^{\mathrm{E}}Y$ axis) and a normal vector to a plane described with gravity and magnetic vectors (${}^{\mathrm{E}}Z$ axis). The vector pointing from the joint which connects segments $j-1$ and j to the IMU and vector pointing from the IMU to the joint which connects segments j and $j+1$ are denoted as ${}^{\mathrm{S}}\boldsymbol{r}_1$ and ${}^{\mathrm{S}}\boldsymbol{r}_2$, respectively. The vectors describe the geometric model of the segment and the IMU placement on the segment. The vector of gravity is marked with ${}^{\mathrm{S}}\boldsymbol{g}$. Quaternion ${}^{\mathrm{E}}\boldsymbol{q}$ describes the segment's orientation with respect to the Earth coordinate frame.

The relation between measured acceleration by the IMU ${}^{\mathrm{S}}\boldsymbol{a}$ and the linear acceleration of the center point of the joint between segments j and $j-1$ ${}^{\mathrm{S}}\boldsymbol{a}_1$ can be described as

$$ {}^{\mathrm{S}}\boldsymbol{a}_j = {}^{\mathrm{S}}\boldsymbol{a}_{1,j} + {}^{\mathrm{S}}\boldsymbol{\omega}_j \times \left({}^{\mathrm{S}}\boldsymbol{\omega}_j \times {}^{\mathrm{S}}\boldsymbol{r}_{1,j} \right) + {}^{\mathrm{S}}\dot{\boldsymbol{\omega}}_j \times {}^{\mathrm{S}}\boldsymbol{r}_{1,j} - {}^{\mathrm{S}}\boldsymbol{g}_j . \tag{1} $$

The measured quantities are transformed into the segment's coordinate frame using rotation ${}^{\mathrm{S}}\boldsymbol{q}_{I,j}$. In Eq. (1), the measured linear acceleration ${}^{\mathrm{S}}\boldsymbol{a}$ comprises dynamic ${}^{\mathrm{S}}\boldsymbol{a}_{\mathrm{DYN}}$ and gravitational ${}^{\mathrm{S}}\boldsymbol{g}$ contributions: ${}^{\mathrm{S}}\boldsymbol{a}_{\mathrm{I}} = {}^{\mathrm{S}}\boldsymbol{a}_{\mathrm{DYN}} - {}^{\mathrm{S}}\boldsymbol{g}$. Gravity vector is transformed into the segment's local coordinate frame as

$$\left[0, {}^{\mathrm{S}}\boldsymbol{g}\right]^{\mathrm{T}} = {}^{\mathrm{E}}\boldsymbol{q}^{*} \otimes \left[0, {}^{\mathrm{E}}\boldsymbol{g}\right]^{\mathrm{T}} \otimes {}^{\mathrm{E}}\boldsymbol{q} \tag{2}$$

where $\otimes$ represents a quaternion multiplication, * denotes a conjugated quaternion, and ${}^{\mathrm{E}}\boldsymbol{g} = [0, -9.81, 0]^{\mathrm{T}}\ \mathrm{m/s^2}$.

The relationship between measured linear acceleration ${}^{\mathrm{S}}\boldsymbol{a}$ and linear acceleration of the joint center point connecting segment j with segment $j+1$ ${}^{\mathrm{S}}\boldsymbol{a}_2$ is expressed as

$${}^{\mathrm{S}}\boldsymbol{a}_{2,j} = {}^{\mathrm{S}}\boldsymbol{a}_j + {}^{\mathrm{S}}\boldsymbol{\omega}_j \times \left({}^{\mathrm{S}}\boldsymbol{\omega}_j \times {}^{\mathrm{S}}\boldsymbol{r}_{2,j}\right) + {}^{\mathrm{S}}\dot{\boldsymbol{\omega}}_j \times {}^{\mathrm{S}}\boldsymbol{r}_{2,j} + {}^{\mathrm{S}}\boldsymbol{g}_j. \tag{3}$$

If the acceleration at the connection point with the previous segment ${}^{\mathrm{S}}\boldsymbol{a}_{2,\mathrm{j}-1}$ is known and transformed into the Earth's coordinate frame, it represents the linear acceleration ${}^{\mathrm{E}}\boldsymbol{a}_{1,j}$ of the j-th segment. Expressed in the coordinate frame of the j-th segment it is used in (1). The aforementioned equations form a basis for the iterative procedure processing a serial kinematic chain of lower extremities segments.

Vector of the magnetic field ${}^{\mathrm{I}}\boldsymbol{B}$, measured by the IMU and expressed in the segment's frame, is modeled as a rotated magnetic field of the Earth in the Earth's coordinate frame ${}^{\mathrm{E}}\boldsymbol{B}$. It can be described as

$$\left[0, {}^{\mathrm{S}}\boldsymbol{B}\right]^{\mathrm{T}} = {}^{\mathrm{E}}\boldsymbol{q}^{*} \otimes \left[0, {}^{\mathrm{E}}\boldsymbol{B}\right]^{\mathrm{T}} \otimes {}^{\mathrm{E}}\boldsymbol{q}. \tag{4}$$

3 Model-Based Extended Kalman Filter

An extended Kalman filter (EKF) algorithm [5] is used to fuse measured data from the IMU sensors and knowledge about the lengths of the segments and sensor placement. The estimation model combines integration of angular velocity with relations (1) and (4). The model incorporates a non-linear state-space presentation of the state (5) and measurement (6) equations

$$\boldsymbol{x}_k = \boldsymbol{f}\left(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}, \boldsymbol{w}_{k-1}\right) \tag{5}$$

$$\boldsymbol{z}_k = \boldsymbol{h}\left(\boldsymbol{x}_k, \boldsymbol{v}_k\right). \tag{6}$$

The non-linear function $\boldsymbol{f}$ describes the relations between the state vector $\boldsymbol{x}_k$ at time step k and the state $\boldsymbol{x}_{k-1}$, the input vector $\boldsymbol{u}_{k-1}$ and superimposed process noise $\mathbf{w}_{k-1}$ at time step $k-1$. The function $\boldsymbol{h}_k$ relates the state to the measurements $\boldsymbol{z}_k$ at time step k with addition of superimposed measurement noise $\boldsymbol{v}_k$.

The state vector $\boldsymbol{x}_k$ in the proposed algorithm for single segment orientation assessment is defined as $\boldsymbol{x}_k = \left[{}^{\mathrm{S}}\boldsymbol{\omega}\ {}^{\mathrm{S}}\dot{\boldsymbol{\omega}}\ {}^{\mathrm{E}}\boldsymbol{q}\ {}^{\mathrm{E}}\dot{\boldsymbol{q}}\ {}^{\mathrm{E}}\ddot{\boldsymbol{q}}\right]^{\mathrm{T}}$ with ${}^{\mathrm{E}}\dot{\boldsymbol{q}}$ and ${}^{\mathrm{E}}\ddot{\boldsymbol{q}}$ being the first and the second time derivative of quaternion ${}^{\mathrm{E}}\boldsymbol{q}$. The function $\boldsymbol{f}$ of the state equation (5) is given as follows: $\boldsymbol{f}_{[1-3]}$ represents time integration of angular velocity ${}^{\mathrm{S}}\boldsymbol{\omega}$ and $\boldsymbol{f}_{[4-6]}$ forward transfer of angular acceleration. The time integration of quaternion ${}^{\mathrm{E}}\boldsymbol{q}$ is described with $\boldsymbol{f}_{[7-10]}$, while $\boldsymbol{f}_{[11-14]}$ and $\boldsymbol{f}_{[15-18]}$ represent the first and the

second time derivative of the quaternion ${}^{\mathrm{E}}\boldsymbol{q}$. Combined function $\boldsymbol{f}$ is presented with Eq. (7).

$$\begin{bmatrix} \boldsymbol{f}_{[1-3]} \\ \boldsymbol{f}_{[4-6]} \\ \boldsymbol{f}_{[7-10]} \\ \boldsymbol{f}_{[11-14]} \\ \boldsymbol{f}_{[15-18]} \end{bmatrix} = \begin{bmatrix} {}^{\mathrm{S}}\boldsymbol{\omega} + {}^{\mathrm{S}}\dot{\boldsymbol{\omega}}_{\mathrm{I}}\Delta t \\ {}^{\mathrm{S}}\dot{\boldsymbol{\omega}} \\ {}^{\mathrm{E}}\boldsymbol{q} + {}^{\mathrm{E}}\dot{\boldsymbol{q}}\Delta t \\ \frac{1}{2}{}^{\mathrm{E}}\boldsymbol{q} \otimes \left[0,\ {}^{\mathrm{S}}\boldsymbol{\omega}\right]^{\mathrm{T}} \\ \frac{1}{2}{}^{\mathrm{E}}\dot{\boldsymbol{q}} \otimes \left[0,\ {}^{\mathrm{S}}\boldsymbol{\omega}\right]^{\mathrm{T}} + \frac{1}{2}{}^{\mathrm{E}}\boldsymbol{q} \otimes \left[0,\ {}^{\mathrm{S}}\dot{\boldsymbol{\omega}}\right]^{\mathrm{T}} \end{bmatrix} \tag{7}$$

where Δt represents the time difference between two sequential time steps.

The measurement vector incorporates all measured quantities and is defined as $\boldsymbol{z}_k = \left[{}^{\mathrm{S}}\boldsymbol{\omega}\ {}^{\mathrm{S}}\dot{\boldsymbol{\omega}}\ {}^{\mathrm{S}}\boldsymbol{a}\ {}^{\mathrm{S}}\boldsymbol{B}\right]^{\mathrm{T}}$. The function $\boldsymbol{h}$ relates measured quantities with state space variables through (1) and (4). The function $\boldsymbol{h}$ is given as

$$\begin{bmatrix} \boldsymbol{h}_{[1-3]} \\ \boldsymbol{h}_{[4-6]} \\ \boldsymbol{h}_{[7-9]} \\ \left[0, \boldsymbol{h}_{[10-12]}\right]^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} {}^{\mathrm{S}}\boldsymbol{\omega} \\ {}^{\mathrm{S}}\dot{\boldsymbol{\omega}} \\ {}^{\mathrm{S}}\boldsymbol{a}_1 + {}^{\mathrm{S}}\boldsymbol{\omega} \times \left({}^{\mathrm{S}}\boldsymbol{\omega} \times {}^{\mathrm{S}}\boldsymbol{r}_1\right) + {}^{\mathrm{S}}\dot{\boldsymbol{\omega}} \times {}^{\mathrm{S}}\boldsymbol{r}_1 - {}^{\mathrm{S}}\boldsymbol{g} \\ {}^{\mathrm{E}}\boldsymbol{q}^* \otimes \left[0, {}^{\mathrm{E}}\boldsymbol{B}\right]^{\mathrm{T}} \otimes {}^{\mathrm{E}}\boldsymbol{q} \end{bmatrix}. \tag{8}$$

The set of equations for discrete-time EKF algorithm implementation is presented in [5].

The proposed configuration of the EKF is used for estimation of the orientation of a single segment on which an IMU is attached. For a valid orientation estimation the linear acceleration of the joint connecting the previous segment must be known. With a recursive procedure processing segment by segment and known either acceleration of the mounting point of the first segment or it's orientation, the orientations of any number of segments in a kinematic chain can be determined.

4 Implementation of Model-Based EKF for Human Motion Kinematics Assessment

For assessing the kinematic parameters in human walking a seven segment model of human body was utilized incorporating left and right foot, shank, thigh, and head–arm–trunk (HAT) segment. A recursive algorithm composed of seven EKFs is implemented for the estimation of the individual segments orientation. When the right foot is in the stance phase the recursive algorithm calculate the orientations in the right-left (RL) direction as illustrated in Fig. 1b: (1) right foot, (2) right shank, (3) right thigh, (4) HAT, (5) left thigh, (6) left shank, and (7) left foot. In situation when the left foot is in the stance phase, the direction of the recursive calculation is reversed (LR direction).

The insole data are used to determine standing and swinging leg. While the first segment is in contact with the floor, the position of the contact center point (foot center of pressure—COP), measured by instrumented shoe insoles, is used to determine the

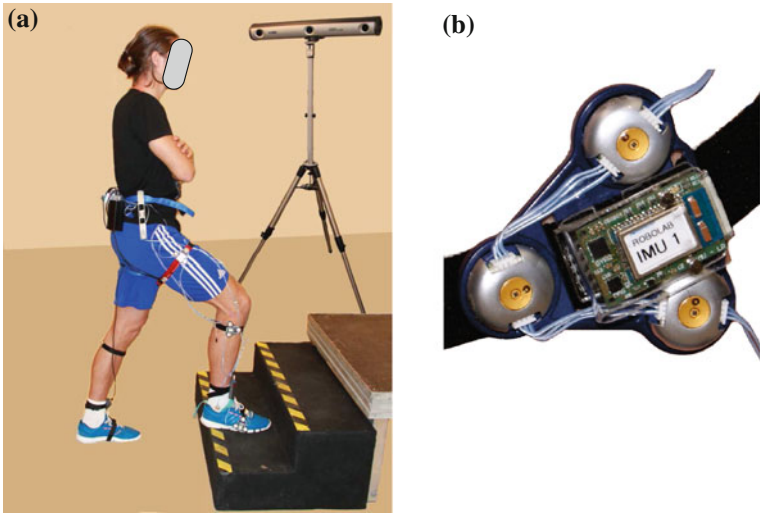


Fig. 2 **a** Subject equipped with sensors walking up the stairs on the polygon. **b** IMU on a measuring plate with corresponding infrared markers

vector ${}^{S}\boldsymbol{r}_{1,1}$ pointing from the sole to the IMU. The acceleration of the contact point is considered as an origin with acceleration $\boldsymbol{a}_{1,1} = 0\,\mathrm{m/s^2}$.

5 Experimental Validation and Results

Wearable sensory system comprises seven IMUs placed on segments of lower extremities and trunk. Two instrumented shoe insoles are used for measuring the feet reaction forces. An IMU (see Fig. 2b) consists of three digital sensors: a three-axial gyroscope (measuring range $\pm 500°/\mathrm{s}$), a three-axial accelerometer (measuring range $\pm 4\,\mathrm{G}$) and a three-axial magnetometer (measuring range $\pm 1.3\,\mathrm{Ga}$) [1]. The size of one unit is $(30 \times 20 \times 5)\,\mathrm{mm}$. Prior measuring, each IMU is calibrated to obtain biases and misalignments of sensors. Measured data from individual IMU are wirelessly transferred to a data acquisition unit. Parotec-System pressure measurement shoe insoles (Paromed GmbH, Neubeuern, Germany) consist of 24 hydrocell sensors each and measure reaction forces of different anatomical points of the foot and the load distribution along the feet. The sensory system operates with a sampling frequency of 100 Hz.

The developed model-based EKF algorithm was tested during long-term walking on a cyclical polygon simulating different types of everyday walking activities: level walking, stair climbing, turning and stair descent (see Fig. 2a). As a reference, an optical measurement system Optotrak Certus (Northern Digital Inc., Waterloo, Canada) was used.

Five males [27.6 ± 3.4 years old, 1.79 ± 0.07 m height, 78 ± 8 kg weight (mean ± standard deviation)] represented a test group. IMUs were mounted on a plastic plate together with three infrared reference markers as shown in Fig. 2b. Measurement plates were fixed on the segments of lower extremities and lower back. Reference markers were also placed over the anatomical landmarks describing the ankle, knee and hip joints. Subjects were fitted with the proper size of the Parotec-System insoles. In the test trials subjects were asked to walk on a polygon for 15 min at a self selected speed. Raw signals from IMUs, insoles, and Optotrak were all sampled synchronously with sample rate of 100 Hz.

Estimates of the segments' orientations ${}^{\mathrm{E}}\boldsymbol{q}_j$ were calculated using the proposed model-based EKF. Reference orientations of the segments were determined from optical measurement system. Reference and estimated joint angles were calculated as the rotational differences between measured orientations of two neighboring segments: ankle angle as difference between foot and shank orientation, knee angle as difference between shank and thigh orientations, and hip angle as difference between thigh and HAT orientations. Absolute errors between estimated and reference joint angles at different time frames (in first, 5th, 10th, and 15th minute) were used to assess the algorithm accuracy and long-term reliability.

5.1 Results

Kinematic analysis of long-term gait encompassed comparison of joint angles of lower extremities assessed by the proposed EKF algorithm and the reference Optotrak system. In the results, joint angles with largest range of motion were analyzed: dorsiflexion/plantar flexion of the ankle, extension/flexion of the knee and extension/flexion of the hip. To test the reliability of the proposed algorithm the error was evaluated in 1st, 5th, 10th and 15th min of the walking.

In tests on a polygon on average a distance of 647 ± 114 m was covered with a speed of 2.6 ± 0.5 km/h. Comparison of typical joint angle trajectories for knees, ankles, and hips for one subject within four time intervals is illustrated in Fig. 3. Trajectories show half of the preparation step and climbing of three stairs with a duration of approximately 3 s. Absolute errors between assessed and reference joint angles for all trails ($n = 5$, 49.8 ± 8.8 rounds per trail, 18.6 ± 3.9 s duration of one round) are presented with boxplots in Fig. 4.

6 Discussion

The presented model-based extended Kalman filter is based on the kinematic relation between linear acceleration and angular velocity of a rigid body. The concept uses accelerometer data for continuous fusion, and not just for inclination estimation during quasi-static conditions as presented in many other papers [8, 9, 15]. With

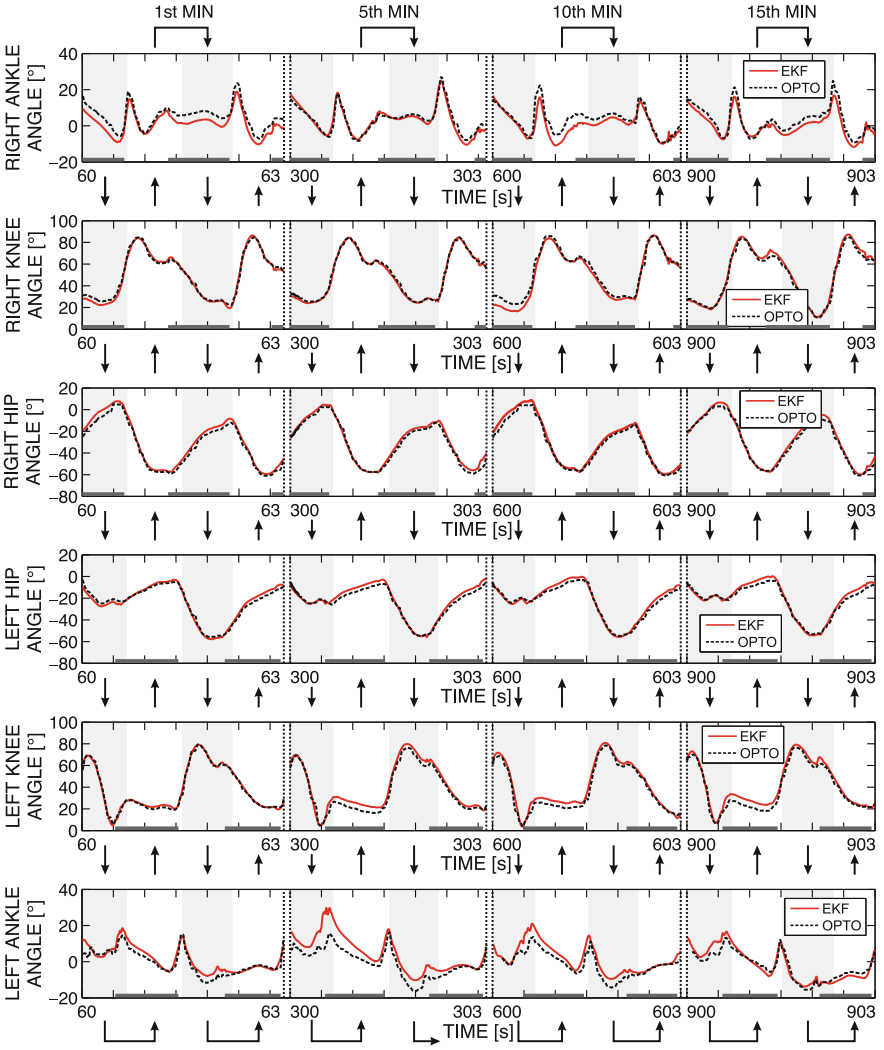


Fig. 3 Right and left ankle, knee, and hip joint angle trajectories during half of the preparation step and stair climbing determined by the proposed model-based EKF algorithm (*solid line*) and reference system (*dashed line*) in walking on polygon. *Grey* and *white areas* denote RL and LR direction of calculation, respectively. *Thick solid lines* represent feet contact durations for each time frame as followed: right foot: floor–1st stair–3rd stair; left foot: floor–2nd stair

a novel EKF, a recursive algorithm is proposed treating the lower extremities of a human as a serial kinematic chain where the base alternates with a swinging leg. In comparison to other methods [1, 11], the measurement insoles are incorporated into the system to distinguish between the stance and the swing leg and to determine the COP which is used for kinematic calculations in Eq. (1) for the first segment.

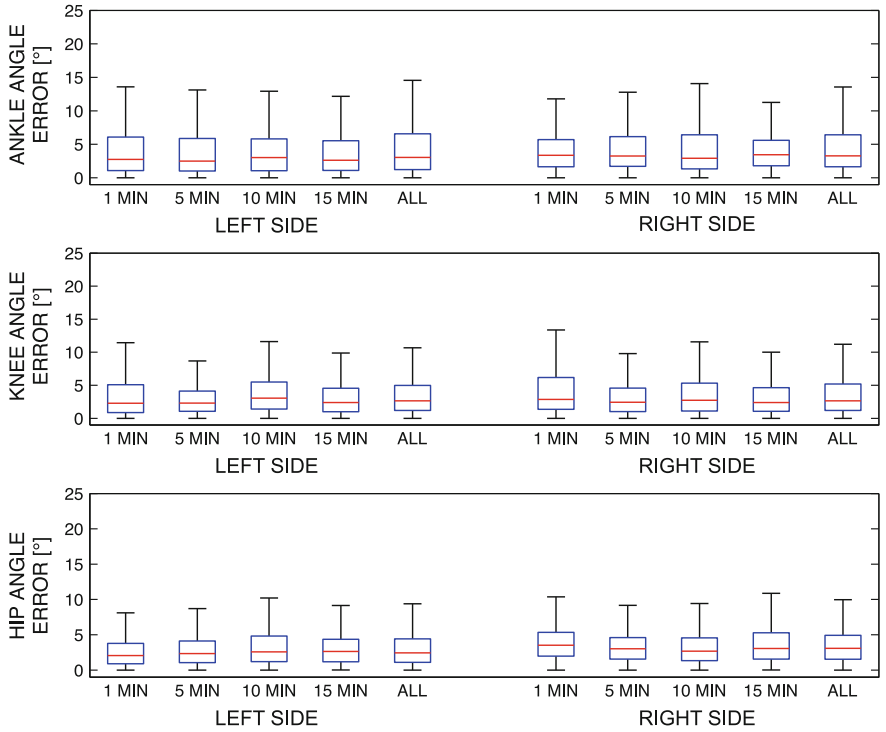


Fig. 4 Absolute error between assessed and reference joint angle trajectories for ankles, knees, and hips during 1st, 5th, 10th, 15th min and during whole trail for all trails accomplished at walking on the polygon

Long-term walking on the polygon simulated different everyday motion types: level walking, stair negotiation, and turning. The experiment tested the performance of the presented algorithm when combining different motion types and evaluated the long-term reliability. The comparison of joint angles trajectories show that the performance of the proposed algorithm is not affected by changing the type of motion. Median errors around 3° for the lower extremities' joint angles validate the adequacy of the model-based EKF algorithm for tracking kinematic parameters in different motion conditions. From results, no tendency of median error increase can be observed.

The novel sensory system and algorithms are appropriate for assessment of joint angles in human walking and other daily activities with improved accuracy and consistency. The only condition to be met for proper operation is that at least one foot must be in contact with the floor at any given time. During regular walking, this condition is inherently met. The proposed approach is also applicable in other areas of mechatronics where orientation assessment of serially linked segments is of interest.

Acknowledgments This study was supported by the Slovenian Research Agency (ARRS) under grant 1000–11–310147 and research programme Motion analysis and synthesis in man and machine (P2–0228), and partially by the EU project CYBERLEGs FP7–ICT–2011–7–287894.

References

1. Beravs, T., Rebersek, P., Novak, D., Podobnik, J., Munih, M.: Development and validation of a wearable inertial measurement system for use with lower limb exoskeletons. In: 11th IEEE-RAS International Conference on Humanoid Robots (Humanoids), 2011, pp. 212–217. IEEE (2011)
2. Chiari, L., Dozza, M., Cappello, A., Horak, F.B., Macellari, V., Giansanti, D.: Audio-biofeedback for balance improvement: an accelerometry-based system. IEEE Trans. Biomed. Eng. **52**(12), 2108–2111 (2005)
3. Crea, S., Vitiello, N., De Rossi, S.M.M., Lenzi, T., Donati, M., Cipriani, C., Carrozza, M.C.: Development of an experimental set-up for providing lower-limb amputees with an augmenting feedback. In: Converging Clinical and Engineering Research on Neurorehabilitation, pp. 321–325. Springer (2013)
4. Favre, J., Jolles, B., Siegrist, O., Aminian, K.: Quaternion-based fusion of gyroscopes and accelerometers to improve 3d angle measurement. Electron. Lett. **42**(11), 612–614 (2006)
5. Greg, W., Gary, B.: An introduction to the Kalman filter. Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill (2006)
6. Kamnik, R., Bajd, T.: Standing-up robot: an assistive rehabilitative device for training and assessment. J. Med. Eng. Technol. **28**(2), 74–80 (2004)
7. Kazerooni, H.: The Berkeley lower extremity exoskeleton. In: Field and Service Robotics, pp. 9–15. Springer, Newyork (2006)
8. Luinge, H., Veltink, P.H.: Measuring orientation of human body segments using miniature gyroscopes and accelerometers. Med. Biol. Eng. Comput. **43**(2), 273–282 (2005)
9. Mayagoitia, R.E., Nene, A.V., Veltink, P.H.: Accelerometer and rate gyroscope measurement of kinematics: an inexpensive alternative to optical motion analysis systems. J. Biomech. **35**(4), 537–542 (2002)
10. Moreno, J.C., de Lima, E.R., Ruíz, A.F., Brunetti, F.J., Pons, J.L.: Design and implementation of an inertial measurement unit for control of artificial limbs: application on leg orthoses. Sens. Actuators, B **118**(1), 333–337 (2006)
11. Roetenberg, D., Luinge, H.J., Baten, C.T., Veltink, P.H.: Compensation of magnetic disturbances improves inertial and magnetic sensing of human body segment orientation. IEEE Trans. Neural Syst. Rehabil.Eng. **13**(3), 395–405 (2005)
12. Sup, F., Bohara, A., Goldfarb, M.: Design and control of a powered transfemoral prosthesis. Int. J. Robot. Res. **27**(2), 263–273 (2008)
13. Takeda, R., Tadano, S., Natorigawa, A., Todoh, M., Yoshinari, S.: Gait posture estimation using wearable acceleration and gyro sensors. J. Biomech. **42**(15), 2486–2494 (2009)
14. Tong, K., Granat, M.H.: A practical gait analysis system using gyroscopes. Med. Eng. Phys. **21**(2), 87–94 (1999)
15. Williamson, R., Andrews, B.: Detecting absolute human knee angle and angular velocity using accelerometers and rate gyroscopes. Med. Biol. Eng. Comput. **39**(3), 294–302 (2001)

Stiffness Matrix of 6-DOF Cable-Driven Parallel Robots and Its Homogenization

Dinh Quan Nguyen and Marc Gouttefarde

Abstract In this chapter, several issues of the stiffness matrix of a general 6-DOF Cable-Driven Parallel Robot (CDPR) are addressed. First, the stiffness matrix of a CDPR with hefty cable is derived analytically. Then, a homogenization of this stiffness matrix is introduced. The discussion on this procedure gives an intuitive meaning of the stiffness matrix of the CDPR and its homogeneous form. This analysis can be used to solve a set of CDPR design problems.

Keywords Cable-driven parallel robots · Stiffness · Homogenization

1 Introduction

Cable-driven parallel robots have been extensively studied and proved to have appealing advantages compared to rigid-link parallel manipulators, such as light weight, large workspace, high load capacity, ease of construction, ease of reconfiguration and low cost. Figure 1 shows a general CDPR whose mobile platform is driven by m cables.

Similar to general robotic manipulators, the design of CDPR, in general, is a process in which many criteria, such as the working volume, positioning accuracy and stiffness, have to be considered. For cable robots, the latter is particularly important especially since CDPR are much more compliant than rigid-link manipulators. The stiffness of a CDPR can be characterized by its stiffness matrix at a given point of its workspace. This matrix relates the forces and torques applied on the mobile

D. Q. Nguyen (✉) · M. Gouttefarde
Laboratoire d'Informatique, de Robotique et de Micro-électronique de Montpellier (LIRMM-CNRS-UM2), 161 rue Ada, 34392 Montpellier Cedex 5, France
e-mail: dinhquan.nguyen@lirmm.fr

M. Gouttefarde
e-mail: marc.gouttefarde@lirmm.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_20,

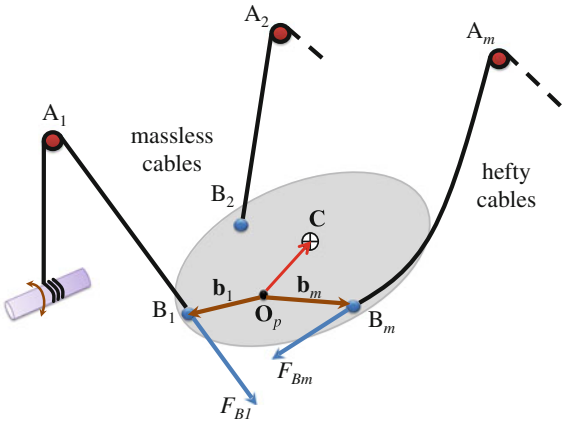


Fig. 1 A general m-cable CDPR

platform to the corresponding linear and angular displacements of the mobile platform. Suppose that $\mathbf{K}$ is the stiffness matrix of a CDPR. In a static equilibrium, an infinitesimal wrench $d\mathbf{f_e}$ applied on the mobile platform will create a small displacement dX that satisfies the relation:

$$\mathbf{K} \cdot dX = d\mathbf{f_e} \tag{1}$$

Many studies on the stiffness of CDPR and its applications have been carried out [1, 3–5, 11, 14]. However, most of them deal with the cases where the cable mass is neglected. These assumptions hold only for CDPR of small dimensions or using light-weight cables. For large-dimension CDPR that are designed to handle heavy payloads, hefty cables are used to drive the mobile platform. Sagging effects of the cables over a wide workspace may have a significant impact on the robot performances. In [3], an analysis of the stiffness of CDPR considering cable mass is proposed. However, this analysis is only applied to a planar 2-DoF CDPR. In [8], Du et al. performed an analysis of the kinematics of a long-span CDPR. Their approach utilized the well-known 2D catenary cable model [7]. From their work, one can also derive the stiffness matrix of a CDPR. However, their derivation is rather complicated and contains errors.

Various means of quantifying the stiffness of a parallel manipulator have been proposed, e.g. in [5, 6, 9, 12]. Among those, characterizing the stiffness matrix $\mathbf{K}$ through its singular values is of interest to us.

Let us consider some of the possible applications of CDPR shown in Fig. 2: a suspended CDPR that can carry workers in an airplane maintenance workshop, or one that carries a robot arm attached to its mobile platform to perform certain tasks over a large volume. In both cases, due to suspended architecture (all cable exit points A_i are located above the mobile platform), the stiffness of the CDPR is determined by its payload (including the mobile platform weight), the cable layout (positions of

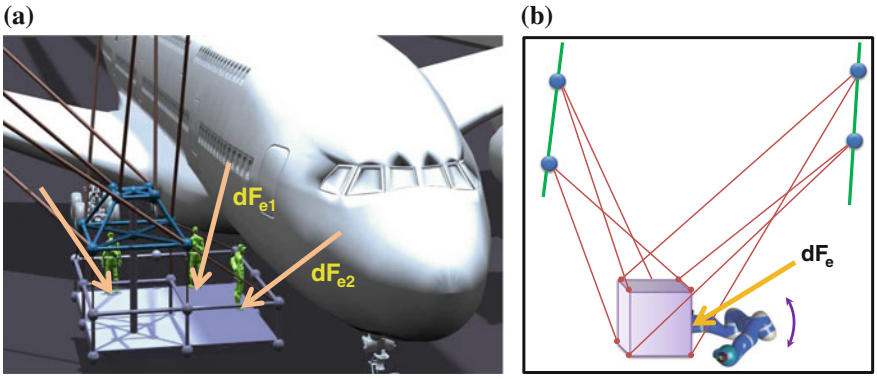


Fig. 2 Examples of applications. **a** Carrying workers. **b** Carrying robot arm

the cable exit points A_i and attachment B_i) and also the cable characteristics. In the design phase of such CDPR, one goal can be to have a high stiffness at the mobile platform. In fact, among a set of design solutions, the solution which has "smaller" resultant displacement dX (with respect to any disturbance wrench $d\mathbf{f_e}$) will have higher stiffness. The disturbance wrench can be caused, e.g., by the workers on the platform (in Fig. 2a) or the robot arm performing a task (in Fig. 2b). In such cases, one need to quantify efficiently the stiffness of the CDPR in order to aid the design process.

From (1), one can derive:

$$\sigma_{\min} = \frac{1}{\|\mathbf{K}^{-1}\|} \leq \frac{\|d\mathbf{f_e}\|}{\|dX\|} \leq \|\mathbf{K}\| = \sigma_{\max} \quad (\forall\, dX \neq \mathbf{0}) \tag{2}$$

where $\sigma_{\min}$ and $\sigma_{\max}$ are the minimum and maximum singular values of the stiffness matrix $\mathbf{K}$, and $\|.\|$ indicates the 2-norm of a vector or a matrix.

In this sense, the stiffness of the CDPR can be quantified by its singular values (specifically $\sigma_{\min}$ and $\sigma_{\max}$) regardless of the magnitude or direction of the infinitesimal wrench $d\mathbf{f_e}$. However, the term $\|d\mathbf{f_e}\|$ and $\|dX\|$ become meaningful only if $d\mathbf{f_e}$ and dX are homogeneous. In [5], Bouchard presented the analysis of the rotational kinematic sensitivity of a CDPR (which is defined as the change in the pose of the mobile platform under a change in the actuation). The infinitesimal position vector dP and the infinitesimal rotation vector $d\Phi$ (here, $dX = (dP,\ d\Phi)$) are quantified separately which helps to avoid the issues due to the difference in units. However, the study focused mainly on the rotational kinematic sensitivity of the CDPR under a change in cable lengths. In [9], Larfourcade et al. presented a way to homogenize the stiffness matrix based on the concept of characteristic lengths [2, 10] to transform angles to lengths in the infinitesimal displacement vector dX and moments to forces in the infinitesimal wrench $d\mathbf{f_e}$. The characteristic length was chosen as the average distance from the anchor points B_i to the origin of the local frame attached to the

mobile platform. This choice, however, may not provide an efficient quantification of the stiffness of CDPR in our applications of interest.

In this chapter, we first derive the analytical form of the stiffness matrix of a general 6-DOF CDPR taking into account cable mass and elasticity. This attempt is in fact an improvement of the work done in [8] by using the extended 3D catenary cable model [13]. Then, we present a procedure to transform the stiffness matrix into its homogeneous form. Based on an analysis of this homogenization, we derive a means to quantify the stiffness of a CDPR.

The chapter is organized as follows. Section 2 describes the stiffness of one hefty cable. Section 3 gives the analytical form of the stiffness matrix of general 6-DOF CDPR. Finally, the homogenization procedure is presented in Sect. 4.

2 Stiffness Matrix of One Cable

Let us consider a steel cable that has unstrained length $L_0\ (m)$, self-weight $w\ (N/m)$, elastic modulus $E\ (Pa)$ and cable cross-section area $A_0\ (m^2)$. Figure 3 shows the relevant coordinates and parameters of a cable in static equilibrium. The cable is fixed between two end-points $A\ (x_a, y_a, z_a)$ and $B\ (x_b, y_b, z_b)$. The term ΔL represents the strain of the cable. f_{bx}, f_{by} and f_{bz} are the cable horizontal and vertical force components at point B and

$$\tau_a = \sqrt{f_{bx}^2 + f_{by}^2 + (f_{bz} - w\,L_0)^2} \tag{3}$$

$$\tau_b = \sqrt{f_{bx}^2 + f_{by}^2 + f_{bz}^2} \tag{4}$$

are the tensions in the cable at point A and B, respectively.

According to the well-known elastic catenary equations [7, 13], we obtain the relation between the coordinates of points A and B:

$$x_a = x_b - \frac{f_{bx}\,L_0}{E\,A_0} + \frac{f_{bx}}{w}\ln\left[\frac{\tau_a + f_{bz} - w\,L_0}{\tau_b + f_{bz}}\right] \tag{5a}$$

$$y_a = y_b - \frac{f_{by}\,L_0}{E\,A_0} + \frac{f_{by}}{w}\ln\left[\frac{\tau_a + f_{bz} - w\,L_0}{\tau_b + f_{bz}}\right] \tag{5b}$$

$$z_a = z_b - \frac{f_{bz}\,L_0}{E\,A_0} + \frac{w\,L_0^2}{2E\,A_0} + \frac{1}{w}\left(\tau_a - \tau_b\right) \tag{5c}$$

If the coordinates of point A are known, then the coordinates of point B can be computed from the cable force components and the cable unstrained length L_0:

$$B = f\left(f_{bx}, f_{by}, f_{bz}, L_0\right) \tag{6}$$

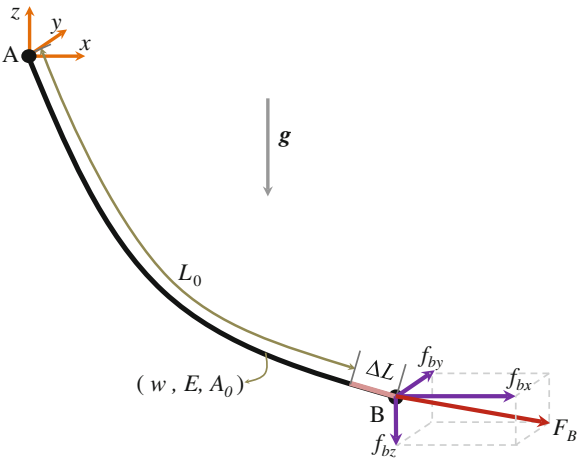


Fig. 3 Sketch of a sagging cable

Differentiating both sides of (6) and applying the chain rule, we have:

$$\begin{aligned} dB &= \frac{\partial B}{\partial F_B} \cdot dF_B + \frac{\partial B}{\partial L_0} \cdot dL_0 \\ \Rightarrow dF_B &= \mathbf{K}_B \cdot dB - K_L \cdot dL_0 \end{aligned} \tag{7}$$

Here, the term $d(\cdot)$ indicates an infinitesimal change in a vector. F_B is the cable force at point B as shown in Fig. 3.

Assume that the end point A is fixed. If the coordinates of point B and the cable force components at B are known, the cable unstrained length L_0 being also known and unchanged ($dL_0 = 0$), then $\mathbf{K}_B$ is the stiffness matrix of the cable at point B:

$$\mathbf{K}_B = \left(\frac{\partial B}{\partial F_B} \right)^{-1} \tag{8}$$

The vector K_L in (7) is:

$$K_L = \left(\frac{\partial B}{\partial F_B} \right)^{-1} \cdot \left(\frac{\partial B}{\partial L_0} \right) = \mathbf{K}_B \cdot F_A \cdot \left(\frac{1}{EA_0} + \frac{1}{\tau_a} \right) \tag{9}$$

where $F_A = \left[f_{bx} \;\; f_{by} \;\; (f_{bz} - wL_0) \right]^T$ is the force applied by the cable at point A (which points toward point B).

3 Stiffness Matrix of 6-DOF CDPR

Let us consider the CDPR shown in Fig. 1. F_{Bi} is the force applied by the mobile platform on the cable i at its end point B_i. The resultant wrench applied by the cables on the mobile platform is:

$$\mathbf{f_c} = \left[\sum_{i=1}^{m} -F_{Bi}, \sum_{i=1}^{m} -(\mathbf{R}\mathbf{b}_i \times F_{Bi}) \right]^{\mathbf{T}} \tag{10}$$

Here, $\mathbf{b}_i$ is the position vector of B_i expressed in the mobile platform local frame. Assume that an infinitesimal external wrench $d\mathbf{f_e}$ is applied on the mobile platform. Then, in static equilibrium:

$$d\mathbf{f_e} = d\left(-\mathbf{f_c}\right) = \begin{bmatrix} \sum_{i=1}^{m} dF_{Bi} \\ \sum_{i=1}^{m} d\left(\mathbf{R}\mathbf{b}_i \times F_{Bi}\right) \end{bmatrix} \tag{11}$$

We have:

$$\begin{aligned} dF_{Bi} &= \mathbf{K}_{Bi} \cdot dB_i - K_{Li} \cdot dL_{0i} \\ &= \mathbf{K}_{Bi} \cdot d\left(\mathbf{R}\mathbf{b}_i + P\right) - K_{Li} \cdot dL_{0i} \\ &= \mathbf{K}_{Bi} \cdot \Big(-(\mathbf{R}\mathbf{b}_i)_\times \cdot d\Phi + dP \Big) - K_{Li} \cdot dL_{0i} \\ &= \mathbf{K}_{Bi} \cdot \left[\mathbb{1}_{3\times 3}, -(\mathbf{R}\mathbf{b}_i)_\times \right] \cdot dX - K_{Li} \cdot dL_{0i} \end{aligned} \tag{12}$$

where $X = \begin{pmatrix} x & y & z & \theta_x & \theta_y & \theta_z \end{pmatrix}$ defines the mobile platform pose, $P = (x \;\; y \;\; z)$ is the position vector in Cartesian space of point O_p, $\theta = \begin{pmatrix} \theta_x & \theta_y & \theta_z \end{pmatrix}$ is a vector of Euler angles, $dP = (dx \;\; dy \;\; dz)$ is the position displacement vector, $d\Phi = \mathbf{S} \cdot d\theta$ is the rotation displacement vector with $d\theta = \begin{pmatrix} d\theta_x & d\theta_y & d\theta_z \end{pmatrix}$ and $dX = (dP, \; d\Phi)$ is the infinitesimal displacement vector of the mobile platform. The term $(\cdot)_\times$ indicates the cross product matrix of a vector, $\mathbb{1}_{3\times 3}$ is the identity matrix. The rotation matrix $\mathbf{R}$ and the transformation matrix $\mathbf{S}$ (which maps the time derivative of the vector of ZYX Euler angles to the mobile platform angular velocity vector) are given as follows:

$$\mathbf{R} = \begin{bmatrix} C_y C_z & C_z S_y S_x - S_z C_x & C_z S_y C_x + S_z S_x \\ C_y S_z & S_z S_y S_x + C_z C_x & S_z S_y C_x - C_z S_x \\ -S_y & C_y S_x & C_y C_x \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} C_y C_z & -S_z & 0 \\ C_y S_z & C_z & 0 \\ -S_y & 0 & 1 \end{bmatrix} \tag{13}$$

with $C_x = \cos(\theta_x)$, $S_x = \sin(\theta_x)$, $C_y = \cos\left(\theta_y\right)$, ...

Moreover, we have:

$$\begin{aligned} d\left(\mathbf{Rb}_i \times F_{Bi}\right) &= d\left(\mathbf{Rb}_i\right) \times F_{Bi} + \left(\mathbf{Rb}_i\right) \times dF_{Bi} \\ &= \left(F_{Bi}\right)_{\times}^{T} \cdot \left(\mathbf{Rb}_i\right)_{\times}^{T} \cdot d\Phi + \left(\mathbf{Rb}_i\right) \times dF_{Bi} \\ &= \left[\left(\mathbf{Rb}_i\right)_{\times} \cdot \mathbf{K}_{Bi}, \quad \mathbf{H}_i - \left(\mathbf{Rb}_i\right)_{\times} \cdot \mathbf{K}_{Bi} \cdot \left(\mathbf{Rb}_i\right)_{\times}\right] \cdot dX \\ &\quad - \left(\mathbf{Rb}_i\right)_{\times} \cdot K_{Li} \cdot dL_{0i} \end{aligned} \tag{14}$$

Here, the matrix $\mathbf{H}_i$ is defined as:

$$\mathbf{H}_i = \left(F_{Bi}\right)_{\times}^{T} \cdot \left(\mathbf{Rb}_i\right)_{\times}^{T} \tag{15}$$

Finally, from (11), (12), (14) we obtain:

$$d\mathbf{f_e} = \mathbf{K} \cdot dX - \mathbf{K}_L \cdot dL_0 \tag{16}$$

where:

$$\mathbf{K} = \left[\begin{array}{c|c} \sum_{i=1}^{m} \mathbf{K}_{Bi} & -\sum_{i=1}^{m} \mathbf{K}_{Bi} \cdot \left(\mathbf{Rb}_i\right)_{\times} \\ \hline \sum_{i=1}^{m} \left(\mathbf{Rb}_i\right)_{\times} \cdot \mathbf{K}_{Bi} & \sum_{i=1}^{m} \mathbf{H}_i - \sum_{i=1}^{m} \left(\mathbf{Rb}_i\right)_{\times} \cdot \mathbf{K}_{Bi} \cdot \left(\mathbf{Rb}_i\right)_{\times} \end{array}\right] \tag{17}$$

and

$$\mathbf{K}_L = \begin{bmatrix} K_{L1} & K_{L2} & \dots & K_{Lm} \\ \left(\mathbf{Rb}_1\right)_{\times} \cdot K_{L1} & \left(\mathbf{Rb}_2\right)_{\times} \cdot K_{L2} & \dots & \left(\mathbf{Rb}_m\right)_{\times} \cdot K_{Lm} \end{bmatrix} \tag{18}$$

where K_{Li} $(i = \overline{1, m})$ is given in (9).

If during the period in which an infinitesimal wrench $d\mathbf{f_e}$ is applied on the mobile platform all the cable unstrained lengths are assumed to be unchanged ($dL_0 = \mathbf{0}$), then we obtain $\mathbf{K}$ as the stiffness matrix of the CDPR at the mobile platform.

Note that, as an additional merit of (16), at the equilibrium state ($d\mathbf{f_e} = \mathbf{0}$), we also obtain the Jacobian matrix of the CDPR:

$$\mathbf{J} = \mathbf{K}^{-1} \cdot \mathbf{K}_L \tag{19}$$

4 Homogenization of Stiffness Matrix

Suppose that from an equilibrium pose, a small disturbance force dF_e is applied on the mobile platform at a point M. This force creates an infinitesimal wrench at O_p, $d\mathbf{f_e} = \left(dF_e, \ \mathbf{r}_M \times dF_e\right)$ $\left(\mathbf{r}_M = \overrightarrow{O_p M}\right)$. From (1), we can write:

$$\mathbf{K} \cdot dX = \begin{bmatrix} \mathbb{1}_{3\times3} & \mathbb{0}_{3\times3} \\ \mathbb{0}_{3\times3} & \|\mathbf{r}_M\| \cdot \mathbb{1}_{3\times3} \end{bmatrix} \cdot \begin{bmatrix} dF_e \\ \dfrac{\mathbf{r}_M}{\|\mathbf{r}_M\|} \times dF_e \end{bmatrix}$$

$$\Leftrightarrow \mathbf{S}_H^{-1} \cdot \mathbf{K} \cdot dX = d\mathbf{f}_{\mathbf{e}H}$$

$$\Leftrightarrow \mathbf{S}_H^{-1} \cdot \mathbf{K} \cdot \mathbf{S}_H^{-1} \cdot \mathbf{S}_H \cdot dX = d\mathbf{f}_{\mathbf{e}H}$$

$$\Leftrightarrow \mathbf{K}_H \cdot dX_H = d\mathbf{f}_{\mathbf{e}H} \tag{20}$$

where

$$\mathbf{S}_H = \begin{bmatrix} \mathbb{1}_{3\times3} & \mathbb{0}_{3\times3} \\ \mathbb{0}_{3\times3} & \|\mathbf{r}_M\| \cdot \mathbb{1}_{3\times3} \end{bmatrix}, \quad d\mathbf{f}_{\mathbf{e}H} = \begin{bmatrix} dF_e \\ \mathbf{u}_M \times dF_e \end{bmatrix}$$

$$dX_H = \mathbf{S}_H \cdot dX \tag{21}$$

$$\mathbf{K}_H = \mathbf{S}_H^{-1} \cdot \mathbf{K} \cdot \mathbf{S}_H^{-1} \tag{22}$$

and $\mathbf{u}_M$ is the unit vector $\dfrac{\mathbf{r}_M}{\|\mathbf{r}_M\|}$.

In (20), the terms dX_H and $d\mathbf{f}_{\mathbf{e}H}$ are homogeneous (the units are meters and Newtons, respectively). The matrix $\mathbf{S}_H$ transforms the stiffness matrix $\mathbf{K}$ into its homogeneous form $\mathbf{K}_H$. The characteristic length used in this transformation is $L_c = \|\mathbf{r}_M\|$. In fact, this transformation preserves the work done by the infinitesimal wrench:

$$(d\mathbf{f}_{\mathbf{e}})^T \cdot dX = (d\mathbf{f}_{\mathbf{e}H})^T \cdot dX_H \tag{23}$$

Obviously, the choice of the characteristic length L_c plays an important role in providing useful physical meaning of $\mathbf{K}_H$.

The matrix $\mathbf{K}$ can be considered homogeneous with the trivial characteristic length $L_c = \|\mathbf{r}_M\| = 1\ (m)$. One can then safely take the 2-norm of the infinitesimal displacement vector dX. However, even if dX_H (or dX) is homogeneous, its components dP_H and $d\Phi_H$ still represent different quantities (position and orientation). Taking the norm of the two terms together in $\|dX_H\|$ and utilize (2) may not be really meaningful. We shall give an interpretation for dP_H and $d\Phi_H$ as follows.

Let us consider the simple example shown in Fig. 4. An infinitesimal wrench $d\mathbf{f}_{\mathbf{e}}$ applied on the mobile platform at point M creates an infinitesimal displacement $dX_H = (dP_H,\ d\Phi_H)$. We will consider the changes of the origin point O_p and point M. Under the action of $d\mathbf{f}_{\mathbf{e}}$, a displacement occurred at O_p and M. One can write:

$$M = O_p + \mathbf{r}_M$$

$$\Rightarrow \|dM\| = \|dO_p + d\Phi \times \mathbf{r}_M\| \leq \|dO_p\| + \|d\Phi \times \mathbf{r}_M\|$$

$$\Rightarrow \|dM\| \leq \|dO_p\| + \|d\Phi\| \cdot \|\mathbf{r}_M\| = \|dP_H\| + \|d\Phi_H\| \tag{24}$$

The magnitude of the displacement at O_p is $\|dP_H\|$. Whereas the magnitude of the displacement at M (in the local mobile platform frame) is bounded by $\|d\Phi_H\|$.

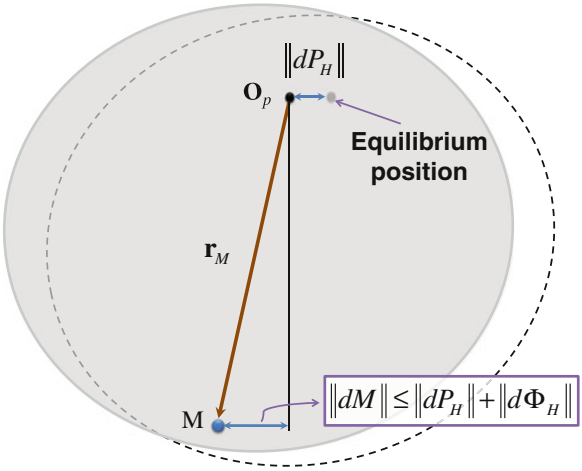


Fig. 4 Displacement of O_p and M at the mobile platform of a CDPR

With this interpretation of the homogeneous infinitesimal displacement vector dX_H, it is proposed to quantify separately the two terms dP_H and $d\Phi_H$. From (20), we can write:

$$\begin{aligned} dX_H &= \mathbf{K}_H^{-1} \cdot d\mathbf{f}_{\mathbf{e}_H} \\ \Rightarrow &\begin{cases} dP_H = \mathbf{C}_P \cdot d\mathbf{f}_{\mathbf{e}H} \\ d\Phi_H = \mathbf{C}_\Phi \cdot d\mathbf{f}_{\mathbf{e}_H} \end{cases} \end{aligned} \tag{25}$$

where $\mathbf{K}_H^{-1} = \left[\mathbf{C}_P^T, \ \mathbf{C}_\Phi^T\right]^T$.

From (25), we have:

$$\frac{\|dP_H\|}{\|d\mathbf{f}_{\mathbf{e}H}\|} \le \|\mathbf{C}_P\| = \sigma_{P\max} \tag{26}$$

$$\frac{\|d\Phi_H\|}{\|d\mathbf{f}_{\mathbf{e}H}\|} \le \|\mathbf{C}_\Phi\| = \sigma_{\Phi\max} \tag{27}$$

$$\Rightarrow \frac{\|dM\|}{\|d\mathbf{f}_{\mathbf{e}H}\|} \le \sigma_M = \sigma_{P\max} + \sigma_{\Phi\max} \tag{28}$$

where $\sigma_{P\max}$ and $\sigma_{\Phi\max}$ are the maximum singular values of the matrices $\mathbf{C}_P$ and $\mathbf{C}_\Phi$, respectively.

Regardless of the magnitude or direction of the infinitesimal wrench $d\mathbf{f}_{\mathbf{e}H}$, the magnitudes of the displacements at points O_p and M are bounded by $\sigma_{P\max} \cdot \|d\mathbf{f}_{\mathbf{e}H}\|$ and $\sigma_{\Phi\max} \cdot \|d\mathbf{f}_{\mathbf{e}H}\|$. The two values $\sigma_{P\max}$ and $\sigma_{\Phi\max}$ can be used as a means to quantify the stiffness of the CDPR.

It is worth noting that the homogeneous stiffness matrix of a CDPR will be determined by the choices of:

- The position of the origin O_p of the local frame attached to the mobile platform.
- The characteristic length L_c (or the choice of point M to be analyzed).

These two factors should be considered together while analyzing the stiffness matrix of a CDPR. For point O_p, a wise selection could be the center of mass C of the mobile platform. In case C falls into a set of possible points $\mathbb{U}_C$, O_p can be chosen as the center point of $\mathbb{U}_C$. For choosing L_c, we propose first to define an area of interest $\mathbb{U}_M$ of point M. Then, L_c can be computed as the average (or the maximum) distance from O_p to the points in $\mathbb{U}_M$. After defining O_p and L_c, by minimizing $\sigma_{P\,\max}$ and $\sigma_{\Phi\,\max}$ (or σ_M), one could obtain an optimal design solution where the displacements at point O_p and at the point(s) M of interest are minimized.

Conclusion

The analytical expression of the stiffness matrix of a general 6-DOF CDPR with hefty cables is presented in this chapter. It provides a useful tool to analyze the stiffness of CDPR. In addition, a homogeneous form of the stiffness matrix is also introduced as a means to quantify the stiffness at the mobile platform of the CDPR. These tools should enable us to solve some design problems for cable-driven robots as mentioned in this chapter.

Acknowledgments The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement No. NMP2-SL-2011-285404 (CABLEBOT).

References

1. Albus, J., Bostelman, R., Dagalakis, N.: The NIST robocrane. J. Robot. Syst. **10**(5), 709–724 (1993)
2. Angeles, J., Chablet, D.: On isotropic sets of points in the plane. Application to the design of robot architectures. In: Advances in Robot Kinematics, pp. 73–82. Portoroz, Slovenia (2000)
3. Arsenault, M.: Stiffness analysis of a planar 2-DoF cable-suspended mechanism while considering cable mass. In: Cable-Driven Parallel Robots, Mechanisms and Machine Science, vol. 12, pp. 405–421 Springer (2013)
4. Behzadipour, Saeed, Khajepour, Amir: Stiffness of cable-based parallel manipulators with application to stability analysis. J. Mech. Des. **128**, 303–310 (2006)
5. Bouchard, S., Gosselin, C.M.: Kinematic sensitivity of a very large cable-driven parallel mechanism. In: ASME International Design Engineering Technical Conferences (2006)
6. Gosselin, C.: Stiffness mapping for parallel manipulators. IEEE Trans. Robot. Autom. **6**(3), 377–382 (1990)
7. Irvine, M.: Cable Structures. MIT Press, Cambridge (1981)

8. Jingli, D., Bao, H., Duan, X., Cui, C.: Jacobian analysis of a long-span cable-driven manipulator and its application to forward solution. Mech. Mach. Theory **45**, 1227–1238 (2010)
9. Lafourcade, P., Zheng, Y.-Q., Liu, X.-W.: Stiffness analysis of wire-driven parallel kinematic manipulators. In: Proceedings of the 11th World Congress in Mechanism and Machine Science, IFToMM, Tianjin, China (2004)
10. Ranjbaran, F., Angeles, J., González-Palacios, M.A.: The mechanical design of a seven-axes manipulator with kinematic isotropy. J. Intell. Robot. Syst. **14**, 21–41 (1995)
11. Surdilovic, D., Radojicic, J., Krüger, J.: Geometric stiffness analysis of wire robots. In: Cable-Driven Parallel Robots, Mechanisms and Machine Science, vol. 12, pp. 389–404. Springer (2013)
12. Svinin, M.M., Hosoe, S., Uchiyama, M.: On the stiffness and stability of Gough-Stewart platforms. In: IEEE International Conference on Robotics and Automation, pp. 3268–3273 (2001)
13. Thai, H.-T., Kim, S.-E.: Nonlinear static and dynamic analysis of cable structures. Finite Elem. Anal. Des. **47**(3), 237–246 (2011)
14. Yu, K., Lee, L.-F., Krovi, V.N.: Simultaneous trajectory tracking and stiffness control of cable actuated parallel manipulator. In: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 1–9 (2009)

Human Motion Mapping to a Robot Arm with Redundancy Resolution

Fanny Ficuciello, Amedeo Romano, Vincenzo Lippiello, Luigi Villani and Bruno Siciliano

Abstract In this chapter the problem of mapping human arm motion to an anthropomorphic robot arm has been addressed using an Xsens MVN motion capture suite and a 7-DoF KUKA LWR. The desired end-effector trajectories of the robot are reconstructed from the human hand, forearm and upper arm trajectories in the Cartesian space obtained from the motion tracking system by means of human arm biomechanical models and sensor fusion algorithms embedded in the Xsens Technology. The desired pose of the robot is reconstructed taking into account the differences between the robot and human arm kinematics and is obtained by suitably scaling the human arm link dimensions. A Cartesian impedance control is designed to replicate, at the robot side, the human wrist motion and a compliant null-space control strategy is applied to solve kinematic redundancy exploiting the compliant behavior of the elbow to obtain suitable body reconfigurations.

Keywords Kinematic model · Kinematic redundancy · Impedance control

F. Ficuciello (✉) · A. Romano · V. Lippiello · L. Villani · B. Siciliano
Dipartimento di Ingegneria Elettrica e Tecnologie dell'Informazione, Università degli Studi di Napoli Federico II, via Claudio 21, 80125 Napoli, Italy
e-mail: fanny.ficuciello@unina.it

A. Romano
e-mail: amedeohavoc@gmail.com

V. Lippiello
e-mail: vincenzo.lippiello@unina.it

L. Villani
e-mail: luigi.villani@unina.it

B. Siciliano
e-mail: bruno.siciliano@unina.it

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_21,

1 Introduction

The development of hardware and software technologies for refining the ability to learn from humans grasping and dynamics manipulation capabilities is of greatest interest in robotics, [3]. A big challenge of humanoid robotics is to provide a robotic arm or a dual-arm system with autonomous and dextrous skills to replicate capacity in performing tasks which are typically executed by humans, [4, 6]. The fields of interest can be in all those applications where the robot closely interacts with humans, namely for surgical, industrial and service applications.

Another important issue in unstructured human-shared environment is to achieve a compliant behavior when interaction occurs in order to realize a proper and safe cooperation between humans and robots. This behavior can be imposed in the joint space to ensure safe interaction, [7]. Moreover, the redundant degrees of freedom can be conveniently used to perform some additional tasks besides the main task [2]. These additional tasks can be a given Cartesian position of a point on the body of robot, such as the elbow position, [9]. A compliant behavior in the null space of the main task can be enhanced to manually adjust robot postures to fit into human targeted ones.

The aim of this work is to integrate a robotic platform able to acquire and transfer human body motion to a robotic system. This platform will be used to learn human motion primitives and to achieve a low-dimensional motion manifold. Replication of human body motion is very important to transfer human knowledge and experience to a robot. For accurate motion capturing a sensor technology that use inertial and magnetic measurement units has been used to measure postures of human bodies. The robot arm used for replicating human motion is a 7-DoF KUKA LWR4 arm, [1]. The control of the robot pose is carried out using an impedance strategy in the Cartesian space which uses, as the setpoint for the position, speed and acceleration, the output coming from a second-order filter that processes the signals from Xsens MVN [10]. The redundant degrees of freedom are used to ensure a compliant behavior of the robot elbow in order to reconfigure its position.

2 Experimental Set-Up

The Xsens MVN suite is a low-cost motion capture system that does not need cameras, emitters or markers, and it is simple to be used both indoors and outdoors. It consists essentially of 17 MTx inertial and magnetic measurement units and comprises 3D gyroscopes, 3D accelerometers and 3D magnetometers sensors through which it is possible to obtain the position measurement and orientation of parts of the body of the wearer. Two Xbus Masters handle the wireless communication with the PC or laptop and synchronize all sensor sampling. The body worn sensors are positioned in correspondence of the feet, legs, thighs, pelvis, shoulders, sternum, head, arm, forearm and hand. In Fig. 1, Xsens MVN is shown and the components

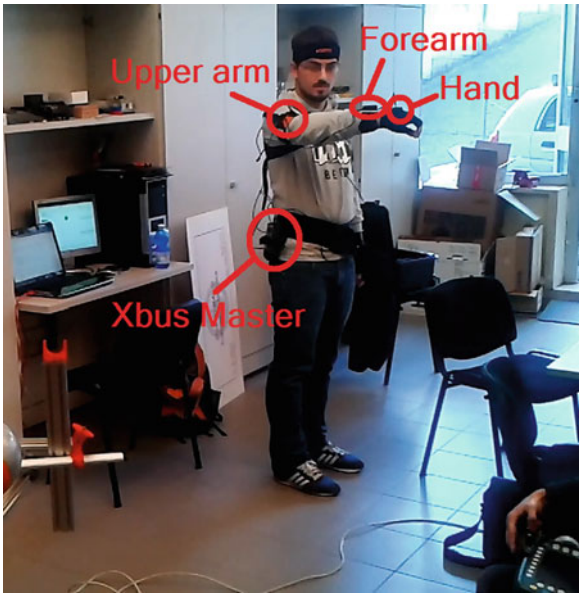


Fig. 1 The Xsens suite worn by the operator is shown. The *Xbus Masters* and the sensors positioned at the *hand*, *forearm* and *upperarm* are highlighted

used for the developed application are highlighted. The total weight of the system is 1.9 Kg. The entire system operates in real-time with a frequency of maximum sampling rate of 120 Hz. By means of a dedicated software (MVN Studio Software), the user can easily observe the movements in 3D, record or export them. The system estimates body segment orientation and position changes by integration of gyroscope and accelerometer signals which are continuously updated by using a biomechanical model of the human body. This allows for tracking of dynamic motion.

When placing the sensors, the initial pose between the sensor and the relative body segment is unknown. Therefore, a calibration procedure, requiring measures such as the height, the arms length and the foot length of the subject, has to be performed. The subject is asked to stand in an a-priori known pose, the rotation from sensor to body segment is determined by matching the orientation of the sensor in the global frame with the known orientation of each segment in this pose.

The robot used to replicate the human arm motion is the 7-DoF KUKA Lightweight Robot (LWR). The kinematic redundancy, similar to the human arm, allows the elbow motion while maintaining the pose of the hand, see Fig. 2. Moreover, the torque sensors, mounted in all joints, allow detecting contact and collisions for safe human-robot interaction and compliant reaction to applied external forces. Thus, the robot can be also manually guided and programmed [1].

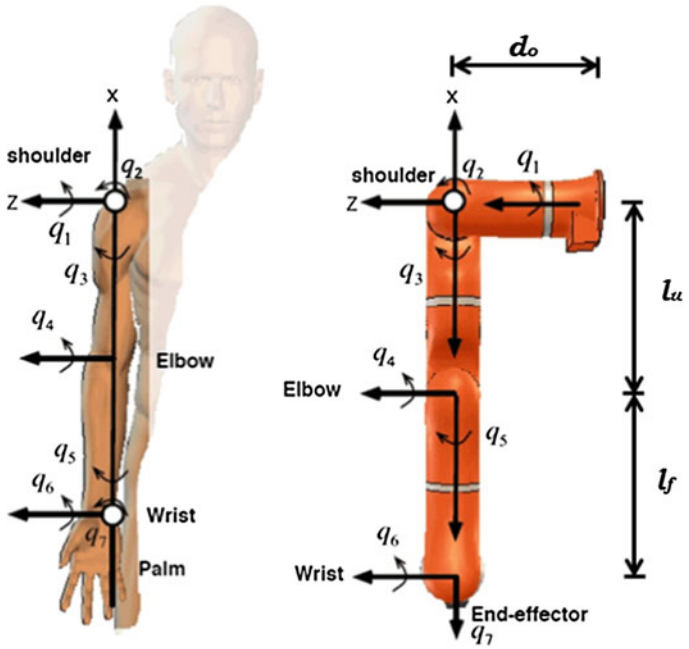


Fig. 2 Human and robot arm comparison, [9]

3 Mapping of the Human Arm Motion

The mapping of movements from the human to the robotic arm requires appropriate considerations regarding the differences in the kinematic chain, size of the two arms and joint limits. The motion tracking system does not provide measurements of the joint angles of the human arm in real time, but allows obtaining that information only from the recorded data. Thus, the mapping of the human arm movement should be referred to measurements in the Cartesian space. Since the human arm positions may be out of the workspace of the robot, to generate a suitable setpoint a bounding box has to be considered for safety issues. Further, to limit the speed and have a smooth motion, the signals coming from the Xsens must be processed by means of interpolation and filtering.

The MTx sensors, mounted on the Xsens suite, provide position and orientation (expressed in terms of a quaternion) of the related segments where they are positioned. Because of the difference between the kinematics of the human arm and that of the robot, we made some assumptions to simplify the mapping. It is assumed that the second joint (q_2) of the robot coincides with the spherical joint of the human shoulder. Such assumption is allowed since the axes of the first three joints of the LWR intersect at a single point corresponding to the center of q_2. The same observation can be made for the sixth joint (q_6) identified as the spherical joint of the human wrist. In

agreement with this assumption the joints that modify the extension of the robot (distance from the shoulder to the wrist) are joints q_2 and q_4 corresponding to the wrist and the elbow of the human arm respectively. During the motion, the distances between joints q_6 and q_4 (robot's wrist and elbow) as well as the distances between joints q_4 and q_2 (robot's elbow and shoulder) remain constant and correspond to the length of the human forearm and upperarm respectively, see Fig. 2. In order to transfer the movement of an operator wearing Xsens suite to the robot, the measurements of interest are the position of the hand, forearm and upperarm, x_h, x_f and x_u respectively, which are provided by the Xsens and are expressed with respect to the global frame of the motion capture system. Since the robot does mount neither a hand nor any tool, only the desired position of the wrist is mapped, while the orientation of the end effector is controlled to be constant at the initial value. Alternatively, it is also possible to map the orientation by means of the composition of the quaternion related to the hand, forearm and upperarm in the same way as for the position. As already stressed, the kinematics of the human and robot are different also in terms of link dimensions. The position setpoints of the robot are generated by modifying the XSens references on the based of the link lengths of the KUKA robot. To compute the wrist reference position of the robot, the versor of the human forearm and upperarm are obtained and multiplied for the robot forearm and upperarm lenghts, $l_f = 0.39\,\text{m}$ and $l_u = 0.40\,\text{m}$ respectively. This quantities are summed with the vector linking joint $q2$ to the base of the robot, $x_b = (0 \quad 0 \quad 0.31)^T$, as in the following equation:

$$x_d = x_b + l_u \frac{x_f - x_u}{\|x_f - x_u\|} + l_f \frac{x_h - x_f}{\|x_h - x_f\|}. \tag{1}$$

The global frame of the Xsens has the same orientation of the base frame of the KUKA LWR, thus no further trasformation is needed. A bounding box is applied to the computed desired wrist position to impose limits on the spatial coordinates, $-0.79 \leq x_x \leq -0.45$, $-0.5 \leq x_y \leq 0.5$, $0.15 \leq x_z \leq 0.6$.

The control of the LWR is designed by means of a Cartesian impedance strategy with compliant null space control for redundancy resolution. The desired position, velocity and acceleration are generated by a second-order filter that processes the signals from Xsens MVN. The output signals of the motion tracking system are first modified according to the robot kinematics in such a way as to generate suitable references for the robot wrist and then filtered in such a way to have stable and feasible setpoints. The control purpose is to let the robot following the pose of the human arm and to allow the reconfiguration of the robot body, relying on a compliant behavior in the null-space of the main task.

3.1 Cartesian Impedance Control

The dynamic model of the robot has the form [8]:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + \tau_f = \tau_c + J^T(q)F_{ext} \tag{2}$$

where $q \in \mathbb{R}^n$, with $n = 7$, is the vector of joint variables, $M(q)$ is the inertia matrix, $C(q, \dot{q})\dot{q}$ is the vector of Coriolis/centrifugal torques, $g(q)$ is the vector of gravitational torques, τ_f is the vector of friction torques, τ_c is the control torque, $J(q)$ is the robot Jacobian, and $\tau_{ext} = J^T F_{ext}$ is the joint torque resulting from external force and torque F_{ext} applied to the end effector, that in this case are null. To design the Cartesian impedance control, it is useful to derive the end-effector dynamics in the operational space [5]:

$$\Lambda(q)\ddot{x} + \mu(q, \dot{q})\dot{x} + F_g(q) + F_f(q) = F_c + F_{ext} \tag{3}$$

where x is the Cartesian pose vector, $\Lambda = (JM^{-1}\mathbf{J^T})^{-1}$ is the (6×6) end effector inertia matrix, while $\mu\dot{x} = \Lambda(JM^{-1}C - \dot{J})\dot{q}$, $F_g = J^{\dagger T} g$, $F_f = J^{\dagger T}\tau_f$ and $F_c = J^{\dagger T}\tau_c$ are the forces, reflected at the end effector, corresponding to the non-inertial joint torques in (2). By neglecting the friction and the Coriolis/centrifugal forces and considering that gravity is compensated, the following impedance control guarantees, in absence of external forces exerted at the robot tip, the tracking of the desired end-effector pose trajectory:

$$\tau_{imp} = J^T(\Lambda(\ddot{x}_d - \dot{J}\dot{q}) + K_v\dot{\tilde{x}} + K_p\tilde{x}, \tag{4}$$

with

$$K_v = \begin{pmatrix} 90I_3 & O \\ O & 10I_3 \end{pmatrix}, \quad K_p = \begin{pmatrix} 2000I_3 & O \\ O & 40I_3 \end{pmatrix}. \tag{5}$$

The impedance gains have been set experimentally. The error between the desired and the effective pose $\tilde{x} = x_d - x_e$ is expressed by means of the position error, $p_d - p_e$, and orientation error expressed in terms of the quaternion $\Delta\varepsilon$:

$$\tilde{x} = \begin{pmatrix} p_d - p_e \\ \Delta\varepsilon \end{pmatrix}, \quad \Delta\varepsilon = \eta_e\varepsilon_d - \eta_d\varepsilon_e - S(\varepsilon_d)\varepsilon_e, \tag{6}$$

where η and ε are the scalar and the vector part of the quaternion and $S(\cdot)$ is the skew-symmetric operator. Snapshots of the experimental results are shown in Fig. 3.

3.2 Compliant Null-Space Control for Redundancy Resolution

Equation (3) describes only the end-effector dynamics and does not include the so-called null space dynamics.

In the presence of redundant degrees of freedom, it is possible to impose a secondary task in the null space of the end effector task, as follows

$$\tau_c = \tau_{imp} + (I - J^T J^{\dagger T})(-k_d\dot{q}), \tag{7}$$

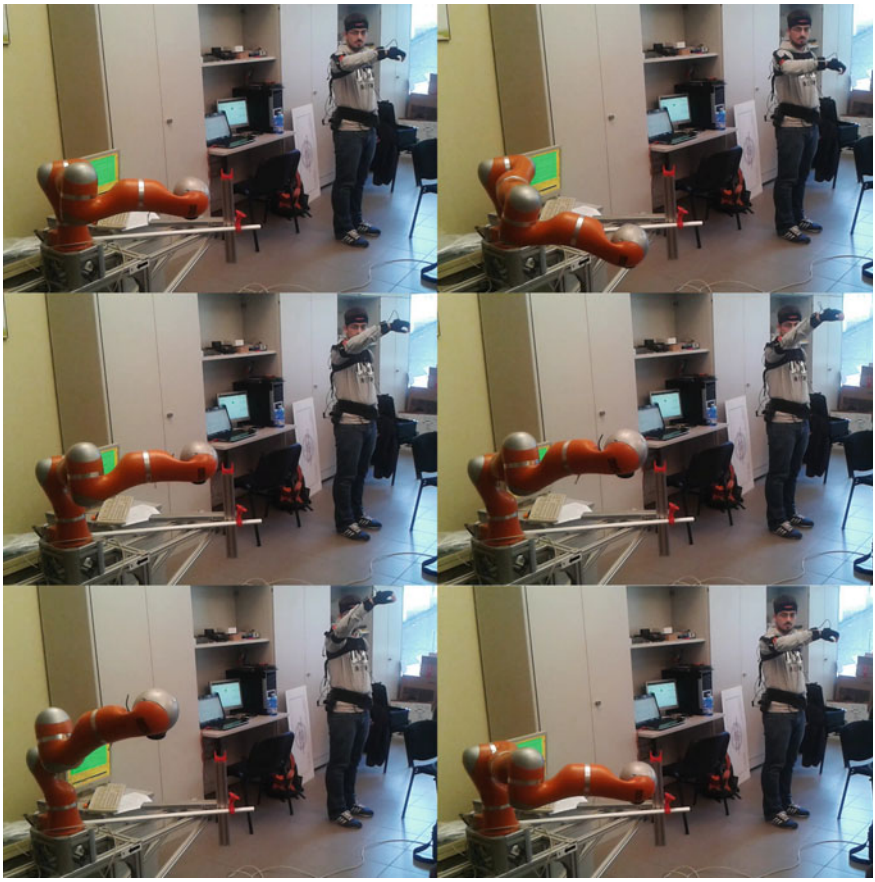


Fig. 3 Snapshots of the experimental set-up during the execution of the motions by the human operator followed by the robot

where $-k_d\dot{q}$, with $k_d = 0.4I_7$, is a suitable damping torque. In this case the secondary task consists in a possible reconfiguration of the arm obtained by applying forces to the robot's body.

Matrix $J^\dagger = M^{-1}J^T[JM^{-1}J^T]^{-1}$ is the dynamically consistent generalized inverse of matrix J [5].

The first image of Fig. 4 represents the configuration assumed by the robot when it is not touched and redundancy is not exploited. In the other two images, two possible configurations, elbow up and elbow down respectively, assumed by the robot, while following the trajectory in the Cartesian space, when complying to external forces exerted on its body are shown.

Fig. 4 From *left* to *right*, the first image represents the robot configuration when redundancy is not exploited, the second and the third images represent other two possible configurations, *elbow up* and *elbow down* respectively

4 Conclusion and Future Work

In this work, a kinematic mapping algorithm has been implemented in order to replicate the movements made by a human arm on an anthropomorphic robot arm with seven degrees of freedom.

The desired trajectories are generated in the Cartesian space in terms of position and orientation and are obtained by taking into account the differences between the kinematics of the robot and of the human arm. The control of the robot pose is carried out using an impedance strategy in the Cartesian space which uses, as the setpoint for the position, speed and acceleration, the output coming from a second order filter that processes the signals from Xsens MVN.

Since the robot is kinematically redundant, a compliant null-space control strategy is employed to adjust the configuration of the robot manually in order to generate anthropomorphic configurations.

The objective of this work is to create a robotic platform to learn from human grasping and manipulation tasks. Future work consists on learning primitives of motion in a low-dimensional manifold for a simplified and human-like control of humanoids robots.

Acknowledgments The research leading to these results has been supported by the SAPHARI and RoDyMan projects, which have received funding from the European Community's Seventh Framework Programme under grant agreements ICT CP-287513 and ERC AdG-320992, respectively. 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. Bischoff, R., Kurth, J., Schreiber, G., Koeppe, R., Albu-Schaffer, A., Beyer, A., Eiberger, O., Haddadin, S., Stemmer, A., Grunwald, G., Hirzinger, G.: The KUKA-DLR lightweight robot arm a new reference platform for robotics research and manufacturing. In: International Symposium on Robotics (2010)

2. Caccavale, F., Lippello, V., Muscio, G., Pierri, F., Ruggiero, F., Villani, L.: Grasp planning and parallel control of a redundant dual-arm/hand manipulation system. Robotica **31**(7), 1169–1194 (2013)
3. Ficuciello, F., Palli, G., Melchiorri, C., Siciliano, B.: A model-based strategy for mapping human grasps to robotic hands using synergies. In: Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 1737–1742. Wollongong (2013)
4. Ficuciello, F., Palli, G., Melchiorri, C., Siciliano, B.: Postural synergies of the UB hand iv for human-like grasping. In: Robotics and Autonomous Systems (2014, in press)
5. Khatib, O.: A unified approach for motion and force control of robot manipulators: the operational space formulation. IEEE J. Robot. Autom. **3**(1), 1115–1120 (1987)
6. Sadeghian, H., Ficuciello, F., Villani, L., Keshmiri, M.: Global impedance control of dual-arm manipulation for safe human-robot interaction. In: 10th International IFAC Symposium on Robot, Control (2012)
7. Sadeghian, H., Villani, L., Keshmiri, M., Siciliano, B.: Task-space control of robot manipulators with null-space compliance. IEEE Trans. Robot. (2014)
8. Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G.: Modelling, Planning and Control. Springer, New York (2009)
9. Wang, Y., Artemiadis, P.: Closed-form inverse kinematic solution for anthropomorphic motion in redundant robot arms. Adv. Robot. Autom. **2**, 110 (2013). doi:10.4172/2168-9695.1000110
10. Xsens Technologies. http://www.xsens.com

Analysis of Geometrical Force Calculation Algorithms for Cable-Driven Parallel Robots with a Threefold Redundancy

Katharina Müller, Christopher Reichert and Tobias Bruckmann

Abstract In this chapter geometrical force calculation algorithms, namely the Barycentric Approach, the Weighted Average Approach, the Corner Projection Method, the Closed Form Method and the Puncture Method are discussed. The last four were implemented for a threefold redundancy and their performance regarding their practical applicability is investigated. The analysis includes the covered workspace, the characteristics of the resulting cable forces, the needed computation time and the adaptability to varying degrees of redundancy.

Keywords Cable-driven parallel robot · Force distribution · Threefold redundancy · Geometrical approaches

1 Introduction

Cable-driven parallel robots (CDPR) move an end effector using flexible cables instead of stiff struts in a parallel topology. Each cable is wound up by a motor-driven winch. The cable can be wound up very quickly and thus, the end effector can reach high velocities and accelerations. Besides a large workspace can be covered. Furthermore the well known advantages of higher stiffness and better load spreading of conventional parallel robots apply for cable driven parallel robots as well [1, 7]. Generally, with the number of cables the stiffness increases, the load distribution

K. Müller (✉) · C. Reichert · T. Bruckmann
Chair of Mechatronics, University Duisburg-Essen, Essen, Germany
e-mail: mueller@imech.de

C. Reichert
e-mail: reichert@imech.de

T. Bruckmann
e-mail: bruckmann@imech.de

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_22,

gets better and even the workspace can be enlarged depending on the configuration of the robot. Nonetheless, a drawback is that the computation of the cable forces gets more complicated.

CDPRs can be classified by the number of cables m, the degree of freedom of the end effector n and the resulting degree of redundancy $r = m - n$ [8, 12]. Because cables can only transmit tension forces, to fully tense a system which has n degrees of freedom at least $m = n + 1$ cables are needed. In this case the term CRPM (Completely Restrained Parallel Manipulator) has gained the most acceptance to characterize the case $r = 1$.

In the case of $m < n + 1$ cables the system is never fully tensed. This class is denoted as IRPM (Incompletely Restrained Parallel Manipulator) whereas a robot with $m > n + 1$ cables is called RRPM (Redundantly Restrained Parallel Manipulator) [2].

Due to the fact that the solution space of the cable force distributions has the dimension r, the computation gets more complicated and more time consuming for higher degrees of redundancy. To be usable as set-point values for control purposes, the results have to be continuous when the end effector is following a continuous trajectory while they have to lie within an upper and a lower force limit to prevent slackness and cable breaks, respectively.

2 Problem Definition

For the static equilibrium, the sum of the forces (cable forces and external forces) and the sum of the torques which affect the end effector have to be zero. In short form this can be written as:

$$\mathbf{A^T f} + \mathbf{w} = \mathbf{0} \tag{1}$$

$$\Leftrightarrow \quad \mathbf{f} = \underbrace{-\mathbf{A}^{+\mathbf{T}}\mathbf{w}}_{\mathbf{f_0}} + \mathbf{H}\lambda. \tag{2}$$

Here $\mathbf{A^T}$ is the structure matrix, $\mathbf{f}$ is the cable force vector, $\mathbf{w}$ is the vector of the external forces and torques, $\mathbf{A}^{+\mathbf{T}}$ is the Moore-Penrose pseudoinverse of the structure matrix and $\mathbf{H}$ is the kernel of the structure matrix.

$\mathbf{f_0}$ is a minimum solution, solving the system of equations Eq. 1 with the smallest Euclidean norm. Because the minimum solution usually lies outside the admissible solution space defined by the force limits defined in Eq. 3, it needs to be moved by choosing an appropriate value for $\mathbf{H}\lambda$.

For a CDPR, Eq. 2 is additionally bounded by a maximum and a minimum permissible tendon force as already introduced:

$$\mathbf{0} < \mathbf{f}_{\min} \leq \mathbf{f} \leq \mathbf{f}_{\max}. \tag{3}$$

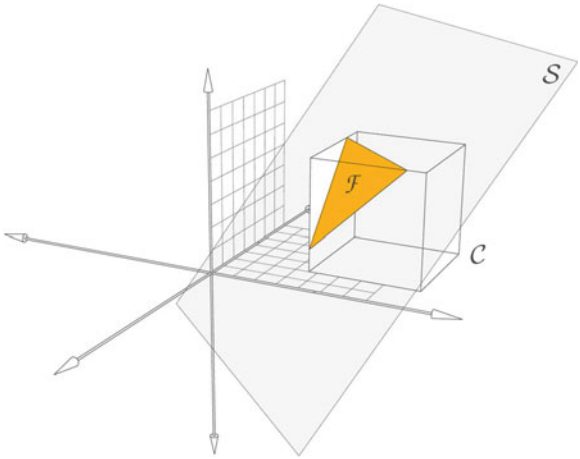


Fig. 1 Geometrical visualization of the problem for $n = 1$ and $m = 3$. *Source* Bruckmann and Reichert [6]

3 Force Calculation Algorithms

Generally, there are two different approaches to calculate feasible cable forces: Either the presented problem can be handled as an optimization problem or it can be interpreted geometrically.

Because optimization problems are usually solved iteratively and the worst-case number of computational steps may be very high, real-time capability usually cannot be guaranteed. For this reason, geometrical approaches are very promising.

In Fig. 1 the problem is visualized geometrically for a RRPM with three cables and one degree of freedom, i.e. $r = 2$. Solutions of Eq. 2 lie in the solution space $\mathscr{S}$ which is a plane in this case (dimension two because of $r = 2$). Solutions of the inequality Eq. 3 lie inside the m-dimensional hypercube $\mathscr{C}$. The intersection $\mathscr{F} = C \cap S$ contains the feasible cable forces.

For a higher number of cables the hypercube gets a higher dimension and for a higher degree of redundancy the solutions space gets a higher dimension as well.

Existing geometric approaches are introduced now:

3.1 Barycentric Approach

For the Barycentric Approach (BA), first the vertices of the intersection have to be identified, then the barycenter of the intersection is computed (for example by triangulation) [1, 3, 4].

3.2 Weighted Average Approach

For the Weighted Average Approach (WAA), also at first the vertices of the intersection have to be identified. Then simply a weighted average is computed. In [1, 5], the weighting coefficient was defined as:

$$a_i = \frac{\sum_{j=1}^{r} \left\| \lambda_{v_i} - \lambda_{u_j} \right\|_p}{\left\| \lambda_{v_i} \right\|_p^{p_n}}, \text{ with } p_n = 1. \tag{4}$$

In the numerator, the distance to the neighbor is included in order to compensate varying numbers of vertices, because—following a continuous trajectory—vertices appear by splitting or disappear by uniting.

In the denominator, the distance to the origin is considered in order to get results having a small norm, leading to a low tension level and thus, low power demands for the winch drives. During research it was found out that it is very useful to set $p_n > 1$ to strengthen the minimizing influence [9].

3.3 Closed Form Method

For the Closed Form Method (CFM), no vertices have to be computed. Instead the center point of the hypercube $\mathscr{C}$ is projected onto the solution space $\mathscr{S}$. Then it is checked, whether the projected center point lies within the cable force boundaries (within the intersection $\mathscr{F}$) [11]. Recently Pott presented an improving extension to this approach which is currently evaluated [10].

3.4 Corner Projection Method

At the Corner Projection Method (CPM), the vertices of the hypercube are projected onto the solution space. The ones which lie inside the cable force boundaries are kept and used to compute another weighted average. This time they are weighted with their smallest distance to one of the cable force boundaries, also with the objective of preventing jumps, because following a continuous trajectory the number of vertices may change: Vertices appear by entering the intersection and disappear by leaving [5].

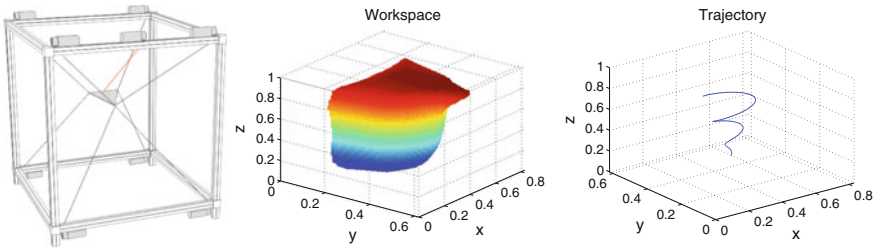


Fig. 2 Chosen robot configuration with its workspace and the used trajectory

3.5 Puncture Method

To the authors' best knowledge, the Puncture Method (PM) has not been presented before. Initially, two points on the solution space $\mathscr{S}$ are needed: One which lies inside the intersection $\mathscr{F}$ (e.g. the solution of the CFM) and one which lies outside, preferably near the origin. Therefore, the projection of the origin onto the solution space ($\mathbf{f_0}$) lends itself to being used. Then these two points get connected by a line and at the intersection point of the line with one of the minimum cable force boundaries contains the wanted cable force distribution.

4 Discussion of Results

WAA, CFM, CPM and PM were analyzed for threefold redundancy and examined for the resulting tendon forces, the covered workspace, the needed computation time and their adaptability to varying redundancies. These are major properties, having a large influence on the practical applicability of any force computation algorithm.

In Fig. 2, the chosen configuration for a parallel cable robot with nine tendons and $n = 6$ degrees of freedom is illustrated. Additionally it shows the trajectory for which the resulting force distributions where computed with the different methods. It was supposed that the end effector (EE) moves along the spiral path from the lowest to the highest end. Using this example, the introduced methods were implemented in MATLAB®. The simulation results are presented below:

4.1 Workspace

To measure the covered workspace, the available space inside of the frame (0.83 m × 0.61 m × 1.00 m) was discretized so that the distance between the grid points was 0.01 m. The cable force boundaries were defined as $f_{\min} = 10\,\mathrm{N}$ and $f_{\max} = 100\,\mathrm{N}$. For each method it was checked whether feasible cable force distributions could be

found for the grid points. The number of covered points was used as an indication of the workspace volume.

The largest workspace is covered by the WAA and contains 108,765 points. This is followed by the CPM which covers only about 85 % of that volume. The CFM covers the smallest workspace with less than 50 % of the size of the WAA's workspace.

Due to the fact that the PM is based on the CFM here, its workspace is as small as well. But there is a possibility to enlarge it (by 15 % in this example): In two steps of the algorithm, the upper cable force limit is manipulated:

In the first step, the upper bounds get increased by a factor. This doesn't affect the position of the intersection point: Due to the proportional enlargement, the new projected center point lies on the same line. But as a result of the wider boundaries, more solutions inside of the manipulated acceptable area can be found. In the second step, the initial limits have to be applied again, because the intersection point contains the resulting cable force distribution and it has to be checked, if it lies inside of the origin boundaries.

The same results can be achieved by completely ignoring the cable force boundaries after the orthogonal projection of the center point onto the solution space and only considering them when checking the intersection.

By this means, the CPM's workspace could be enlarged and now contains 62,542 points. Still this is just 57 % of the size of the WAA's workspace.

4.2 Characteristics of the Cable Force Distributions

In Fig. 3 the resulting cable force distributions of the analyzed methods are presented compared to those computed by Quadratic Programming (QP) (Fig. 3a). The cable force boundaries again were defined as $f_{\min} = 10$ N and $f_{\max} = 100$ N.

The WAA provides cable forces which converge close to the results of QP when the exponent p_n increases (Figs. 3b, c). The resulting cable forces of the PM are close to the QP results, too. Contrarily, the force values generated by the CP and the CFM are quite high.

Similar to the WAA, the CPM may use a minimizing denominator with an exponent, but its influence is much less efficient and may even cause jumps in the force values for higher powers of denominator (remarkable for $p_n < 5$ in this example). This is because here not the vertices of the intersection, but the hypercube-vertices projected onto the intersection are used. These projections usually lie within the intersection and not at the boundaries, so even the ones with smaller norm values are far away from optimal solutions (less effective). They appear or disappear by entering or leaving the intersection and not by splitting or uniting, so at that moment when the projection crosses the boundary no other point with similar values exists. That leads to jumps when for higher powers the denominator outweighs the smoothing influence of the numerator.

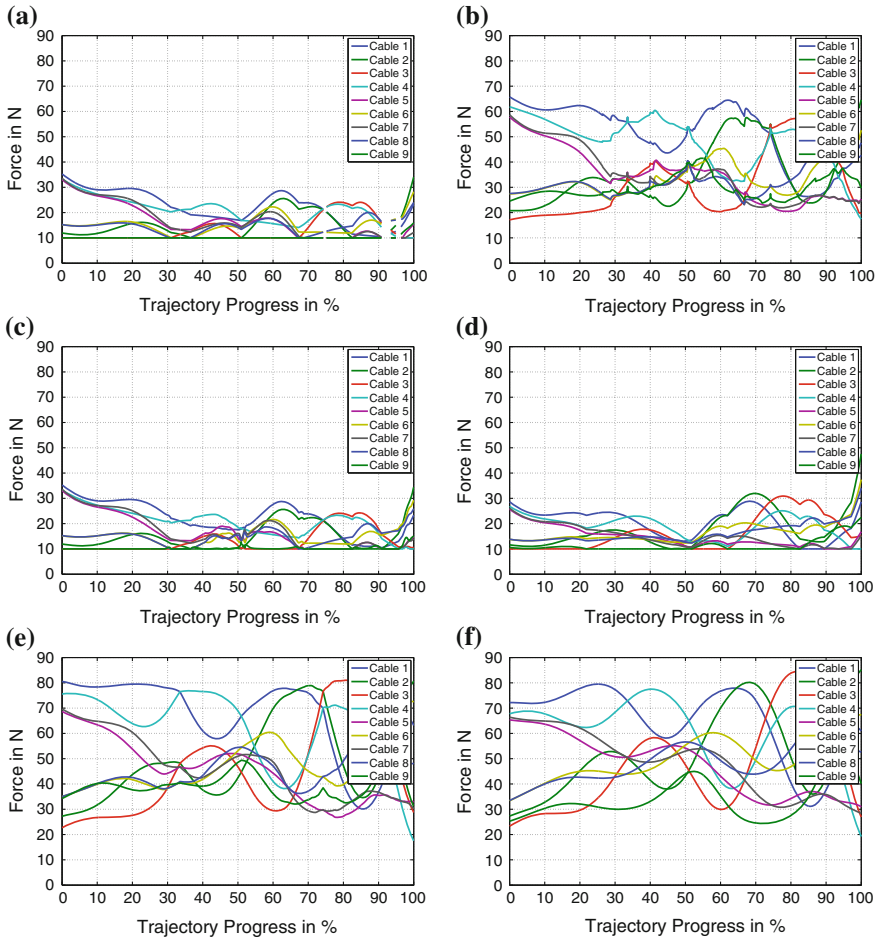


Fig. 3 Resulting cable force distributions. **a** Quadratic Optimization **b** Weighted Average **c** Weighted Average with $p_n = 10$ **d** Puncture Method **e** Corner Projection Method **f** Closed Form Method

4.3 Computation Time

The needed computation time was determined by running each method on a real time system using TwinCAT3® by BECKHOFF on an Intel® Core2™ Duo CPU T9400 processor.

The CPM (maximum value 875 µs, mean value 745 µs) and the WAA (maximum value 743 µs (outlier), mean value 652 µs) needed by far the most computation time. As shown in Fig. 4a, b, the number of vertices correlates with the changes in the needed computation time for both methods.

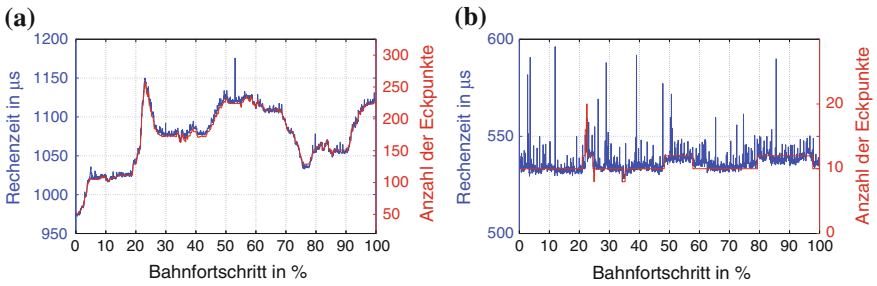


Fig. 4 Computation time. **a** Corner Projection **b** Weighted Average

The fastest algorithm is the CFM, which needs only around 21 µs with little oscillation of <4 µs that can be led back to system latency. The PM based on the CFM is just slightly slower (around 24 µs, oscillation <4 µs).

4.4 Adaptability to Varying Redundancies

The presented methods were implemented and analyzed for a threefold redundancy. CPM, CFM and PM are adaptable to varying redundancies without changes of the code. They even work for higher redundancies what was exemplarily tested up to $r = 6$.

Only the WAA's code has to be adjusted for higher redundancies because by now its structure directly depends on the degree of redundancy.

5 Conclusion

Concluding can be said that the WAA and the PM provided outstanding results concerning most practical aspects. They both deliver nearly optimal tendon forces, but while the PM is very fast and thereby very well suitable for real time applications, the WAA needs approximately 25 times more computation time. On the other hand, the WAA covers a significantly larger workspace. So for applications where the trajectory is known and the tendon forces can be computed offline, this method might be the first choice.

Acknowledgments The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement No. NMP2-SL-2011-285404 (CableBOT).

References

1. Bruckmann, T.: Auslegung und Betrieb redundanter paralleler Seilroboter. Ph.D. Dissertation, Universität Duisburg-Essen (2010)
2. Bruckmann, T., Mikelsons, L., Brandt, T., Hiller, M., Schramm, D.: Wire robots Part I—kinematics, analysis and design. In: Lazinica, A. (ed.) Parallel Manipulators—New Developments, pp. 109–132. I-Tech Education and Publishing, Austria (2008)
3. Bruckmann, T., Mikelsons, L., Brandt, T., Hiller, M., Schramm, D.: Wire robots Part II—dynamics, control and application. In: Lazinica, A. (ed.) Parallel Manipulators—New Developments, pp. 133–153. I-Tech Education and Publishing, Austria (2008)
4. Bruckmann, T., Mikelsons, L., Hiller, M., Schramm, D.: A new force calculation algorithm for tendon-based parallel manipulators. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2007, pp. 1–6 (2007)
5. Bruckmann, T., Mikelsons, L., Pott, A., Abdel-Maksoud, M., Brandt, T., Schramm, D.: A novel tensed mechanism for simulation of maneuvers in wind tunnels. In: ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE2009) August 30–September 2, 2009. San Diego, California, USA: 33rd ASME Mechanics and Robotics Conference (MECH 2009) 7, 17–24 (2009)
6. Bruckmann, T., Reichert, C.: Real-time determination of set point cable force distributions. European Robotics Forum, Lyon (2013)
7. Landsberger, S.E.: Design and construction of a cable-controlled, parallel link manipulator. Masterthesis, Massachusetts Institute of Technology (1984)
8. Ming, A., Higuchi, T.: Study on multiple degree-of-freedom positioning mechanism using wire (Part 1). Int. J. Jpn Soc. Precis. Eng. **28**, 131–138 (1994)
9. Müller, K.: Methoden zur Berechnung der Seilkraftverteilungen höherredundanter paralleler Seilroboter. Master's Thesis (2013)
10. Pott, A.: An improved force distribution algorithm for over-constrained cable-driven parallel robots. In: F. Thomas, A. Perez Gracia (eds.) Computational Kinematics, Mechanisms and Machine Science, vol. 15, pp. 139–146. Springer, Netherlands (2014)
11. Pott, A., Bruckmann, T., Mikelsons, L.: Closed-form force distribution for parallel wire robots. In: A. Kecskemethy, A. Müller (eds.) Computational Kinematics, pp. 25–34. Springer, Berlin (2009)
12. Verhoeven, R.: Analysis of the workspace of tendon-based stewart platforms. Dissertation, Gerhard-Mercator-Universität -GH- Duisburg (2004)

Kinetostatic Analysis of Cable-Driven Parallel Robots with Consideration of Sagging and Pulleys

Marc Gouttefarde, Dinh Quan Nguyen and Cédric Baradat

Abstract Cable-driven parallel robots manipulating heavy payloads typically use cables having non-negligible diameters and mass. The associated cable guiding pulleys may then have non-negligible radii whose influence on the robot kinematics can hardly be neglected. This chapter focuses on the output pulleys from which the cables extend to the robot mobile platform. A kinetostatic analysis of cable-driven parallel robots considering both cable mass and output pulleys is presented.

Keywords Cable-driven parallel robots · Kinematics · Static · Hefty cable modeling

1 Introduction

The mobile platform of a cable-driven parallel robot is driven by a number of cables. The cable lengths are generally modified by means of winches. Each cable is wound around a winch drum and is routed to the mobile platform by means of a set of pulleys. This chapter focuses on the output pulley, i.e., the "last" pulley from which the cable extends to the mobile platform. As shown in Fig. 1, this pulley should be mounted on a revolute joint whose (vertical) axis is coincident with the cable segment exiting the pulley toward the winch. The revolute joint allows the pulley to align with the cable segment which exits the pulley toward the mobile platform. Such output pulley

M. Gouttefarde (✉) · D. Q. Nguyen
LIRMM, CNRS - University Montpellier 2, Montpellier, France
e-mail: marc.gouttefarde@lirmm.fr

D. Q. Nguyen
e-mail: dinhquan.nguyen@lirmm.fr

C. Baradat
Tecnalia France, Montpellier, France
e-mail: cedric.baradat@tecnalia.com

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_23,

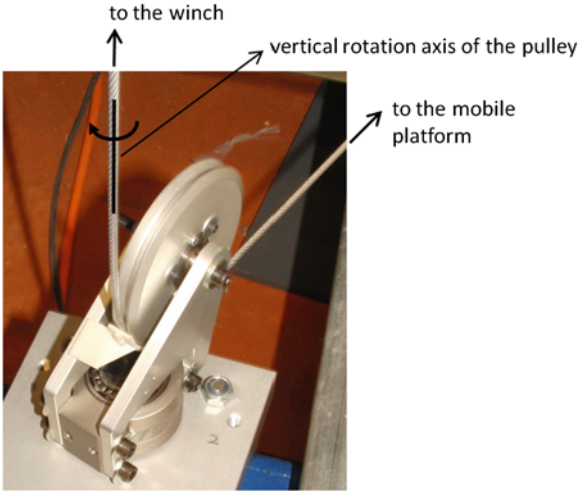


Fig. 1 Example of an output pulley mounted on a *vertical axis* revolute joint

mechanisms are, for example, used in the cable-driven parallel robots described in [1] and [8].

The radii of the pulleys used to guide the cables should be greater than a minimum value to ensure that the cable bending radius remains greater than an acceptable minimum value. Typically, the diameter of the pulley should be greater than ten times or more that of the cable. For large-dimension cable-driven parallel robots, heavy payloads and/or in the case of high safety factors, the cables have a diameter of about 10 mm or more. Hence, both the non-negligible cable mass and output pulley radius may have to be taken into account in the robot kinematics.

Several previous works deal with the static analysis of large-dimension cable-driven parallel robots, e.g. [2–4, 6, 10], using the well-known elastic catenary cable modeling [5] in order to account for the cable mass and elasticity. Besides, a few papers, e.g. [7, 9], takes into account the influence of an output pulley in their analysis of cable-driven parallel robots, but considering the cables as being massless. To the best of our knowledge, a kinetostatic (or geometrico-static) analysis of cable-driven parallel robots considering both cable mass and output pulleys has never been proposed. Such an analysis is thus the contribution of the present chapter.

The chapter is organized as follows. Section 2 presents the well-known elastic catenary hefty cable modeling. Section 3 introduces the equations of the static equilibrium of a sagging cable segment, described by the elastic catenary, this segment being attached at a given point of the robot mobile platform at one extremity and wound on a pulley at its other extremity. Section 4 briefly presents the corresponding kinetostatic analysis in which the kinematics and statics are coupled. Section 5 concludes the chapter.

2 Elastic Catenary Cable Modeling

Let us consider a 6-DOF cable-driven parallel robot whose mobile platform is driven by m cables. In a fixed reference frame $\mathscr{R} = (O, X, Y, Z)$, the position vector of the reference point P of the mobile platform is denoted $\mathbf{p}$. Defining a mobile frame $\mathscr{R}_P = (P, X_P, Y_P, Z_P)$ attached to the mobile platform at P, the orientation of the platform is defined by the rotation matrix $\mathbf{Q}$ from (X, Y, Z) to (X_P, Y_P, Z_P). The position vectors in $\mathscr{R}$ of the cable drawing points A_i are denoted $\mathbf{a}_i$. In this section, like in most previous works, the points A_i are supposed to be fixed in space. In practice, this assumption is correct when the cables extend from eyelets or for output pulleys having a small radius. The positions in the mobile frame $\mathscr{R}_P$ of the cable attachment points B_i on the platform are denoted $\mathbf{b}_i$. In the fixed frame $\mathscr{R}$, the vector $\overrightarrow{B_i A_i}$ from the cable attachment point to the cable drawing point is defined as $\mathbf{l}_i = (l_{ix}, l_{iy}, l_{iz})^T = \mathbf{a}_i - \mathbf{Q}\mathbf{b}_i - \mathbf{p}$.

As detailed in several previous works, notably in [6], in order to account for cable mass and elasticity, the elastic catenary cable modeling [5] can be considered. Under static loading conditions, cable i lies in the vertical plane $\mathscr{P}_i$ containing both points A_i and B_i. As shown in Fig. 2, let us consider the local frame $\mathscr{R}_i = (A_i, X_i, Y_i, Z_i)$ such that $Z_i \equiv Z$ (directed vertically upward) and with X_i pointing toward the mobile platform point B_i. The angle α_i between X and X_i is given by

$$\alpha_i = \text{atan2}(-l_{iy}, -l_{ix}). \tag{1}$$

Angle α_i depends only on the pose (position and orientation) of the mobile platform and on the constant position vectors $\mathbf{a}_i$ and $\mathbf{b}_i$. The rotation matrix $\mathbf{Q}_i$ defining the orientation of (X_i, Y_i, Z_i) with respect to (X, Y, Z) is

$$\mathbf{Q}_i = (\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i) = \begin{pmatrix} \cos\alpha_i & -\sin\alpha_i & 0 \\ \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 1 \end{pmatrix} = \frac{1}{\sqrt{l_{ix}^2 + l_{iy}^2}} \begin{pmatrix} -l_{ix} & l_{iy} & 0 \\ -l_{iy} & -l_{ix} & 0 \\ 0 & 0 & \sqrt{l_{ix}^2 + l_{iy}^2} \end{pmatrix}. \tag{2}$$

Under the actions of its own weight and of the forces applied at its extremities, the part of the cable located between A_i and B_i sags and stretches. According to the elastic catenary modeling, the following two equations give the coordinates ${}^i B_{i_x}$ and ${}^i B_{i_z}$ of the cable attachment point B_i in the local frame $\mathscr{R}_i$ [6]

$${}^i B_{i_x} = \frac{\tau_{li_x} l_{0i}}{E A_0} + \frac{\tau_{li_x}}{\rho_0 g}\left[\sinh^{-1}\left(\frac{\tau_{li_z}}{\tau_{li_x}}\right) - \sinh^{-1}\left(\frac{\tau_{li_z} - \rho_0 g l_{0i}}{\tau_{li_x}}\right)\right] \tag{3}$$

$${}^i B_{i_z} = \frac{\tau_{li_z} l_{0i}}{E A_0} - \frac{l_{0i}^2 \rho_0 g}{2 E A_0} + \frac{1}{\rho_0 g}\left[\sqrt{\tau_{li_x}^2 + \tau_{li_z}^2} - \sqrt{\tau_{li_x}^2 + \left(\tau_{li_z} - \rho_0 g l_{0i}\right)^2}\right]. \tag{4}$$

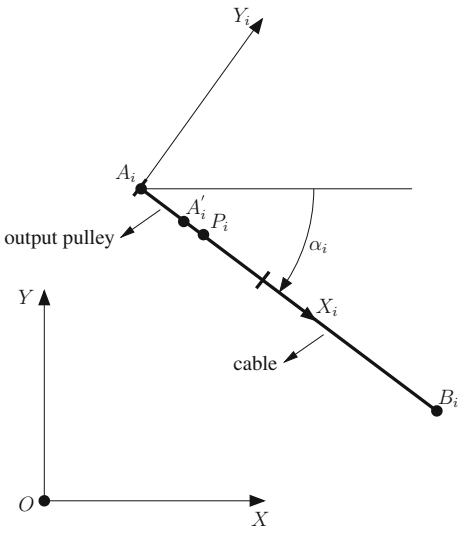


Fig. 2 *Top view* of cable i extending from point A_i (or from A_i' if the output pulley is considered) to the platform point B_i

In these equations, E, A_0, ρ_0 and l_{0i} are the Young's modulus, unstrained cross-sectional area, unstrained linear density and unstrained length between points A_i and B_i of cable i, respectively. g is the gravity acceleration. τ_{li_x} and τ_{li_z} are the components in the local cable frame $\mathscr{R}_i$ of the force $\boldsymbol{\tau}_{li} = (\tau_{li_x}, 0, \tau_{li_z})^T$ applied by the platform to the cable at its end point B_i. Note that $\tau_{li_x} \geq 0$ since, because of the choice of the direction of X_i, $\tau_{li_x} < 0$ would mean that cable i is working in compression.

Besides, the position vector of point B_i in the local frame $\mathscr{R}_i$ can be written

$$\begin{pmatrix} {}^iB_{i_x} \\ 0 \\ {}^iB_{i_z} \end{pmatrix} = -\mathbf{Q}_i^T \mathbf{l}_i = \begin{pmatrix} \sqrt{l_{ix}^2 + l_{iy}^2} \\ 0 \\ -l_{iz} \end{pmatrix}. \tag{5}$$

3 Elastic Catenary and Output Pulley

As illustrated in Fig. 3, when an output pulley is used and considered in the modeling, the position of point A_i' from which the cable exits the output pulley and extends toward the mobile platform is not fixed in space. The output pulley can rotate about the vertical axis Z_i shown in Fig. 3 so that its circumference (pitch diameter) should always lie in the vertical plane $\mathscr{P}_i$ containing the cable. Point A_i' lies on the circumference of the output pulley at a position which depends on the position of point B_i and on the force $\boldsymbol{\tau}_{li}$ applied to the cable at B_i. The origin A_i of the local frame $\mathscr{R}_i$

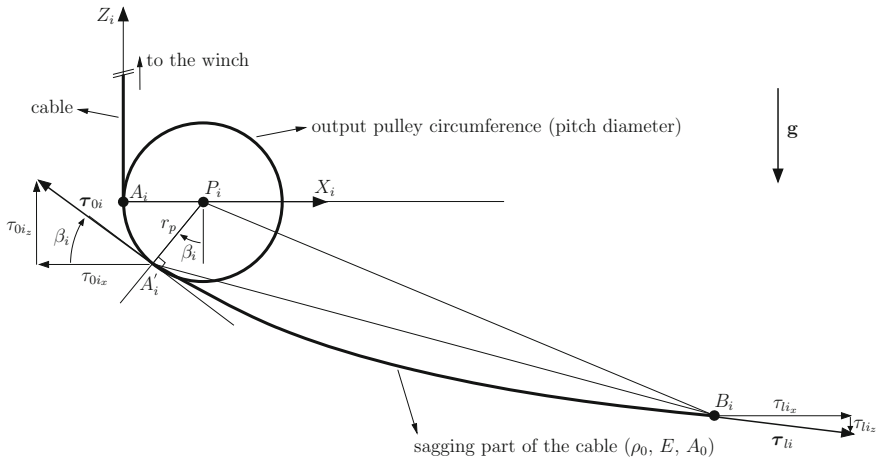


Fig. 3 Cable routing by means of an output pulley

is placed at the point from which the cable exits the output pulley toward the winch. Since A_i lies on the pulley vertical rotation axis Z_i, its position is fixed in space.

The position vector of A'_i in $\mathscr{R}_i$ can be written

$$\overrightarrow{A_i A'_i} = r_p \begin{pmatrix} 1 + \sin(\beta_i) \\ 0 \\ -\cos(\beta_i) \end{pmatrix}_{\mathscr{R}_i} \tag{6}$$

where β_i is the angle shown in Fig. 3 (angle β_i is positive in the counterclockwise direction) and r_p denotes the radius of the output pulley. When point B_i is located below the output pulley (case of Fig. 3), we have $-\pi/2 \leq \beta_i \leq 0$. When B_i is located above the pulley, we have $0 \leq \beta_i \leq \pi/2$.

The part of the cable which sags under its own weight is located between A'_i and B_i whereas the cable segment between A_i and A'_i is wound around the output pulley. Hence, the two non-zero coordinates (along X_i and Z_i) of vector $\overrightarrow{A'_i B_i}$ are given by the right-hand sides of Eqs. (3) and (4). Moreover, according to Eq. (6) and since $\overrightarrow{A_i B_i} = \overrightarrow{A_i A'_i} + \overrightarrow{A'_i B_i}$, the non-zero coordinates ${}^i B_{i_x}$ and ${}^i B_{i_z}$ of the cable attachment point B_i in the local frame $\mathscr{R}_i$ satisfy

$${}^i B_{i_x} = r_p(1+\sin(\beta_i)) + \frac{\tau_{li_x} l_{0i}}{E A_0} + \frac{\tau_{li_x}}{\rho_0 g}\left[\sinh^{-1}\left(\frac{\tau_{li_z}}{\tau_{li_x}}\right) - \sinh^{-1}\left(\frac{\tau_{li_z} - \rho_0 g l_{0i}}{\tau_{li_x}}\right)\right] \tag{7}$$

$$ {}^{i}B_{i_z} = -r_p \cos(\beta_i) + \frac{\tau_{li_z} l_{0i}}{EA_0} - \frac{l_{0i}^2 \rho_0 g}{2EA_0} + \frac{1}{\rho_0 g}\left[\sqrt{\tau_{li_x}^2 + \tau_{li_z}^2} - \sqrt{\tau_{li_x}^2 + \left(\tau_{li_z} - \rho_0 g l_{0i}\right)^2}\right]. \tag{8} $$

Angle β_i can be removed from Eqs. (7) and (8). Indeed, the force $\boldsymbol{\tau}_{0i}$ applied to the cable segment $A_i' B_i$ at point A_i' is tangent to the cable at point A_i'. Hence, it is also tangent to the pulley at this same point and we have

$$ \sin(\beta_i) = \frac{-\tau_{0i_z}}{\sqrt{\tau_{0i_x}^2 + \tau_{0i_z}^2}}, \quad \cos(\beta_i) = \frac{-\tau_{0i_x}}{\sqrt{\tau_{0i_x}^2 + \tau_{0i_z}^2}} \tag{9} $$

where ${}^{i}\boldsymbol{\tau}_{0i} = (\tau_{0i_x}, 0, \tau_{0i_z})^T$ is the force $\boldsymbol{\tau}_{0i}$ expressed in $\mathscr{R}_i$ and the minus signs are due to the fact that β_i is negative in Fig. 3. Moreover, the static equilibrium of the cable segment $A_i' B_i$ gives

$$ \tau_{0i_x} = -\tau_{li_x}, \quad \tau_{0i_z} = -\tau_{li_z} + \rho_0 g l_{0i} \tag{10} $$

where $\rho_0 g l_{0i}$ is the weight of the cable segment $A_i' B_i$. Equations (9) and (10) imply

$$ \sin(\beta_i) = \frac{\tau_{li_z} - \rho_0 g l_{0i}}{\sqrt{\tau_{li_x}^2 + \left(\tau_{li_z} - \rho_0 g l_{0i}\right)^2}} \tag{11} $$

$$ \cos(\beta_i) = \frac{\tau_{li_x}}{\sqrt{\tau_{li_x}^2 + \left(\tau_{li_z} - \rho_0 g l_{0i}\right)^2}}. \tag{12} $$

Finally, putting together Eqs. (7) and (11), as well as (8) and (12), we obtain

$$ \begin{aligned} {}^{i}B_{i_x} = {} & r_p \left(1 + \frac{\tau_{li_z} - \rho_0 g l_{0i}}{\sqrt{\tau_{li_x}^2 + \left(\tau_{li_z} - \rho_0 g l_{0i}\right)^2}}\right) \\ & + \frac{\tau_{li_x} l_{0i}}{EA_0} + \frac{\tau_{li_x}}{\rho_0 g}\left[\sinh^{-1}\left(\frac{\tau_{li_z}}{\tau_{li_x}}\right) - \sinh^{-1}\left(\frac{\tau_{li_z} - \rho_0 g l_{0i}}{\tau_{li_x}}\right)\right] \end{aligned} \tag{13} $$

$$ \begin{aligned} {}^{i}B_{i_z} = {} & \frac{-r_p \tau_{li_x}}{\sqrt{\tau_{li_x}^2 + \left(\tau_{li_z} - \rho_0 g l_{0i}\right)^2}} \\ & + \frac{\tau_{li_z} l_{0i}}{EA_0} - \frac{l_{0i}^2 \rho_0 g}{2EA_0} + \frac{1}{\rho_0 g}\left[\sqrt{\tau_{li_x}^2 + \tau_{li_z}^2} - \sqrt{\tau_{li_x}^2 + \left(\tau_{li_z} - \rho_0 g l_{0i}\right)^2}\right] \end{aligned} \tag{14} $$

which describe the static equilibrium of an hefty cable segment attached, at one extremity, at point B_i to the robot mobile platform and, at its other extremity, entering tangentially the output pulley at A'_i.

In this section, we dealt with the case in which the part of the cable going to the winch exits the output pulley upwardly as shown in Fig. 3. The other case in which the cable exits the output pulley downwardly to the winch leads to equations much similar to Eqs. (13) and (14). These equations are not presented here due to space limitations but can be obtained by following exactly the same reasoning as the one proposed in this section.

4 Kinetostatic Analysis

Similarly to the case of a model including cable mass but neglecting the influence of output pulleys, e.g. [6], Eqs. (13) and (14) involve the mobile platform pose (through the coordinates ${}^iB_{i_x}$ and ${}^iB_{i_z}$ of point B_i), the cable lengths l_{0i} and the forces $\boldsymbol{\tau}_{li}$ applied by the platform to the cables. Therefore, in order to solve an inverse or direct kinematic problem, the static equilibrium of the mobile platform needs also to be considered.

The force applied by the platform to the cable at point B_i is equal to $-{}^i\boldsymbol{\tau}_{li} = (-\tau_{li_x}, 0, -\tau_{li_z})^T$. In the basis (X, Y, Z) of the fixed reference frame, the corresponding wrench at the platform reference point P is

$$-\begin{pmatrix} \mathbf{Q}_i\,{}^i\boldsymbol{\tau}_{li} \\ \mathbf{Qb}_i \times \mathbf{Q}_i\,{}^i\boldsymbol{\tau}_{li} \end{pmatrix}_{6\times 1} = \begin{pmatrix} \mathbf{x}_i & \mathbf{z}_i \\ \mathbf{Qb}_i \times \mathbf{x}_i & \mathbf{Qb}_i \times \mathbf{z}_i \end{pmatrix}_{6\times 2} \begin{pmatrix} \tau_{li_x} \\ \tau_{li_z} \end{pmatrix} \tag{15}$$

where $\mathbf{x}_i$ and $\mathbf{z}_i$ denote the first and third column vectors of $\mathbf{Q}_i$ (Eq. 2), respectively. The net wrench applied by the m cables on the platform at point P is then

$$\mathbf{f} = -\mathbf{W}_s\boldsymbol{\tau}_l \tag{16}$$

with

$$\boldsymbol{\tau}_l = \begin{pmatrix} \tau_{l1_x} & \tau_{l1_z} & \dots & \tau_{lm_x} & \tau_{lm_z} \end{pmatrix}^T_{2m} \tag{17}$$

$$\mathbf{W}_s = \begin{pmatrix} \mathbf{x}_1 & \mathbf{z}_1 & \mathbf{x}_2 & \mathbf{z}_2 & \dots & \mathbf{x}_m & \mathbf{z}_m \\ \mathbf{Qb}_1 \times \mathbf{x}_1 & \mathbf{Qb}_1 \times \mathbf{z}_1 & \mathbf{Qb}_2 \times \mathbf{x}_2 & \mathbf{Qb}_2 \times \mathbf{z}_2 & \dots & \mathbf{Qb}_m \times \mathbf{x}_m & \mathbf{Qb}_m \times \mathbf{x}_m \end{pmatrix}_{6\times 2m}. \tag{18}$$

Note that a minus sign appears in (16) because $\boldsymbol{\tau}_{li}$ is defined as the force applied by the platform on the cable. Let us also note that $\mathbf{W}_s$ depends only on the pose of the platform.

The static equilibrium of the mobile platform can be written

$$-\mathbf{W}_s\boldsymbol{\tau}_l + \mathbf{f}_e = \mathbf{0}. \tag{19}$$

The external wrench $\mathbf{f}_e$ is simply given by

$$\mathbf{f}_e = (0, 0, -m_p g, 0, 0, 0)^T \tag{20}$$

when the reference point P coincides with the platform center of mass, m_p being the platform mass (including the payload, if any).

To conclude this section, let us consider the case of an inverse problem for a 6-DOF parallel robot driven by $m = 6$ cables: Knowing the position $\mathbf{p}$ and orientation $\mathbf{Q}$ of the platform, we want to determine the unstrained lengths l_{0i} of the cable segments from the platform to the output pulley. The unknowns of the problem are the six cable lengths l_{0i} but also the 12 force components gathered in vector $\boldsymbol{\tau}_l$ which need to be considered because they are involved in Eqs. (13) and (14). This total of 18 unknowns is equal to the number of available equations: 6 equations in (19) plus 12 equations in (13) and (14) for $i = 1, \ldots, 6$. The determination of the unstrained lengths l_{0i} thus requires the resolution of this "square" system of 18 non-linear equations in 18 unknowns subjected to the 12 inequality constraints $l_{0i} \geq 0$ and $\tau_{li_x} \geq 0$, $i = 1, \ldots, 6$. The inequalities $\tau_{li_x} \geq 0$ account for the inability of the cables to push on the mobile platform.

5 Conclusion

This chapter introduced a kinetostatic analysis of cable-driven parallel robots considering both non-negligible cable mass and elasticity and output pulley radii. The corresponding non-linear system of equations turns out to have the same structure as the one obtained when the cable mass is accounted for but the pulley influence is disregarded. The sole difference is one additional term in each one of the two equations describing the static equilibrium of the cable segment extending from the output pulley to the robot mobile platform.

Acknowledgments The research leading to these results has received funding from the European Communitys Seventh Framework Programme under grant agreement No. NMP2-SL-2011-285404 (CABLEBOT).

References

1. Bruckmann, T., Lalo, W., Sturm, C., Schramm, D., Hiller, M.: Design and realization of a high rack storage and retrieval machine based on wire robot technology. In: Savi, M. (ed.) DINAME 2013. ABCM, Brazil (2013)
2. Du, J., Bao, H., Duan, X., Cui, C.: Jacobian analysis of a long-span cable-driven manipulator and its application to forward solution. Mech. Mach. Theory **45**, 1227–1238 (2010)

3. Duan, B.Y., Qiu, Y.Y., Zhang, F.S., Zi, B.: On design and experiment of the feed cable-suspended structure for super antenna. Mechatronics **19**, 503–509 (2009)
4. Gouttefarde, M., Collard, J.F., Riehl, N., Baradat, C.: Simplified static analysis of large-dimension parallel cable-driven robots. In: Proceedings of IEEE International Conference Robotics and Automation, pp. 2299–2305. Saint Paul (2012)
5. Irvine, M.: Cable Structures. MIT Press, Cambridge (1981)
6. Kozak, K., Zhou, Q., Wang, J.: Static analysis of cable-driven manipulators with non-negligible cable mass. IEEE Trans. Robot. **22**(3), 425–433 (2006)
7. Pott, A.: Influence of pulley kinematics on cable-driven parallel robots. In: Lenarčič, J., Husty, M.L. (eds.) Advances in Robot Kinematics, pp. 197–204. Springer, Dordrecht (2012)
8. Pott, A., Mutherich, H., Kraus, W., Schmidt, V., Miermeister, P., Verl, A.: IPAnema: a family of cable-driven parallel robots for industrial applications. In: Bruckmann, T., Pott, A. (eds.) First International Conference on Cable-Driven Parallel Robots, pp. 119–134. Springer (2013)
9. Surdilovich, D., Radojicic, J., Kruger, J.: Geometric stiffness analysis of wire robots: a mechanical approach. In: Bruckmann, T., Pott, A. (eds.) First International Conference on Cable-Driven Parallel Robots, pp. 389–404. Springer (2013)
10. Zi, B., Duan, B., Du, J., Bao, H.: Dynamic modeling and active control of a cable-suspended parallel robot. Mechatronics **18**, 1–12 (2008)

Direct and Inverse Second Order Kinematics for Hyper-Redundant Parallel Robots

Georges Le Vey

Abstract Second order direct and inverse continuous recursions for parallel hyper-redundant robots are derived in a purely deductive manner. The method extends previous results by the author, that generalized to a single, completely actuated, flexible beam the Newton–Euler recursive approach for serial robots. This is done thanks to a convenient formulation as an optimal control problem in the spatial coordinate. Consideration of such abstract models of robots and all the more getting both recursions at once appear to be quite new. Thus, only the theory is presented here, deferring simulations to future publication. Indications for future implementations are given, relying upon software from control theory.

Keywords Parallel hyper-redundant robot · Newton–Euler · Continuous inverse direct recursions · Optimal control · Cosserat theory · Nonlinear beam

1 Introduction

Parallel robots with rigid links have been widely studied for several decades and those with flexible links are also currently under active study [1, 2], In the present rather prospective work, one departs from the usual problems and the focus is on atypical hyper-redundant parallel robots for which the links are modelled as nonlinear Cosserat beams with distributed controls, under the hypothesis of neither extensibility nor shearing. On another line, analogy of recursive Newton–Euler equations for multibody systems with optimal filtering [3] and discrete-time optimal control [4] has been evidenced. A purely deductive approach within multistage optimal control theory brought a significant improvement [5] for getting this classical case of discrete Newton–Euler formalism. Results of [5] were generalized to continuous theories

G. Le Vey (✉)
Ecole des Mines de Nantes/IRCCyN, 4, rue A. Kastler, 44300 Nantes, France
e-mail: levey@emn.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_24,

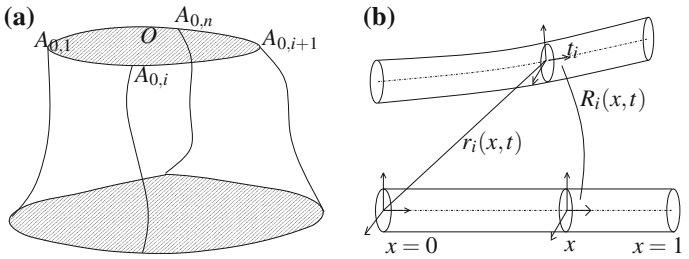


Fig. 1 **a** A hyper-redundant parallel robot. **b** Model of one leg

in [6] under the above kinematical hypothesis. In the present work, this approach to robotic systems through optimal control theory is further extended to derive direct and inverse second order continuous recursions for the dynamics of a parallel hyper-redundant robot, as an example of complex robots. Such mechanisms are obviously abstractions at the present stage but they can be thought of as higher level models for mechanisms where complex hybrid (serial-parallel) links would lead otherwise to very complex equations. Such models can be useful e.g. as simulation tools, at the preliminary stage of design.

2 Kinematics of the Legs and Connection to the Platform

Notations: dots (resp. primes) over some quantity indicate time (resp. space) differentiation. For a vector $y \in \mathbb{R}^3$, $\hat{y}$ is the antisymmetric matrix such that $\forall z \in \mathbb{R}^3$, $\hat{y}z = y \times z$ and $\times$ is the usual vector product. $\mathscr{I}$ is the identity matrix in $\mathbb{R}^3$. The dependence on time and space is implicit for all variables, except when needed.

Consider a parallel robot with n deformable legs, labelled with index i, each linked to the mobile rigid platform labelled with index 0 (see Fig. 1a). Each leg, a slender continuous deformable body, is actuated on its whole length and modelled as a flexible nonlinear Cosserat beam [7, 8] (see Fig. 1b). This means that it is a one dimensional medium with a continuous set of three *directors* (i.e. a frame) attached to each point. One restricts here to the case where only one of these directors, the one tangent to the neutral line, is considered: this amounts to consider an unstretchable, unshearable beam [8], the cross-section remaining orthogonal to the neutral line. Significantly more work would be needed in the most general case. With this hypothesis, all legs can be assumed to have the same unit length: for leg i with length l_i and arclength parameter x_i, the scaling $x_i = l_i x$ implies $x \in [0, 1]$. Thus space derivatives will be understood with respect to this normalized x from now on, and denoted by primes. The configuration space of each beam is the principal bundle $\mathbb{R}^3 \times SO(3)$, a cross section at x being described by $r_i(x,t) \in \mathbb{R}^3$, the position of its mass center in a reference frame, and $R_i(x,t) \in SO(3)$, its attitude (see Fig. 1b). The basis of the robot is at $x = 0$ and the connection to the

platform is at $x = 1$. Its kinematics is described first by a twist-curvature tensor field C_d, ruling the deformation of the leg, hence being an input to the dynamics, and second by the constraint imposed by the spherical kinematics, i.e. formally: $\forall t, \forall x \in [0,1] \ : \ C_d(x,t) = R_i(x,t)^T R_i'(x,t)$ and $r_i'(x,t) = t_i(x,t)$, where t_i is the unit vector tangent to the neutral line of the beam, R_i is the rotation matrix mapping the mobile basis at x before deformation onto that after (see Fig. 1b). Two time differentiations of the previous set of equations give successively a kinematic model of velocities and accelerations. As the focus is here on the dynamics, rewrite the second one in matrix-vector form for the ease of analogy with optimal control [6]:

$$\begin{pmatrix} \ddot{r}_i \\ \dot{\omega}_i \end{pmatrix}' = \begin{pmatrix} 0 & -\hat{t}_i \\ 0 & \hat{k}_i \end{pmatrix} \begin{pmatrix} \ddot{r}_i \\ \dot{\omega}_i \end{pmatrix} + \begin{pmatrix} 0 \\ \ddot{k}_i \end{pmatrix} + \begin{pmatrix} \omega_i \times (\omega_i \times t_i) \\ \dot{k}_i \times \omega_i \end{pmatrix} \tag{1}$$

where ω_i is the axial vector corresponding to the matrix $\hat{\omega}_i = \dot{R}_i R_i^T$, and k_i, the axial vector corresponding to the matrix $\hat{k}_i = R_i C_d R_i^T$. Notice that in (1), $\ddot{r}_i$ and $\dot{\omega}_i$ appear as elements of a *state* vector and $\ddot{k}_i$ as a *control input*, of a linear *state equation*, in the language of control systems theory that will be used below, with independent variable the space dimension x, not the time. Also, the last term is an inhomogeneity, not depending on this so defined state, and can be computed before, for each x. Now, as the legs are connected to the mobile platform, their motions are constrained. The platform has total mass M_0, inertia tensor I_0 and linear (resp. angular) velocity V_{g0} (resp. ω_0) of (resp. around) mass center O. Set $V_{0,i}$ (resp. $\omega_{0,i}$) to be the linear (resp. angular) velocity of $A_{0,i}$, the point where the ith leg connects to the platform (see Fig. 1a), d_i being the distance from the mass center of the platform to $A_{0,i}$. The twist of this rigid body allows to write: $V_{0,i} = V_{g0} + \omega_0 \times d_i$ and, after one time differentiation:

$$\dot{V}_{0,i} = \dot{V}_{g0} + \dot{\omega}_0 \times d_i + \omega_0 \times (\omega_0 \times d_i). \tag{2}$$

For the rotation velocities: $\omega_{0,i} = \omega_0$ and thus $\dot{\omega}_{0,i} = \dot{\omega}_0$. It is then useful for the formulation in the next section to rewrite (2) as: $\dot{V}_{0,i} + \hat{d}_i \dot{\omega}_{0,i} = \dot{V}_{g0} + \omega_0 \times (\omega_0 \times d_i)$ i.e. in matrix form:

$$\begin{pmatrix} \mathscr{I} & \hat{d}_i \end{pmatrix} \begin{pmatrix} \dot{V}_{0,i} \\ \dot{\omega}_{0,i} \end{pmatrix} = \dot{V}_{g0} + \omega_0 \times (\omega_0 \times d_i). \tag{3}$$

3 Appell's Function Computation

Recall [9, 10] that Appell's approach to Gauss least constraint consists in building first the acceleration energy $\mathscr{R}$. Then, including active efforts through a work functional gives the so-called *Appell function* [9], a quadratic function of the accelerations. The kinematical constraints are then added at the level of accelerations using

multipliers. The minimum of the resulting functional gives the dynamical equations. This approach is well-suited to derive dynamical equations as it works directly with accelerations. This section is dedicated to the computation of $\mathscr{R}$ for the overall parallel hyper-redundant robot. Firstly, the Appell's function for the platform is [9, 11]:

$$\mathscr{R}_0 = \frac{1}{2}\dot{\omega}_0^T I_0 \dot{\omega}_0 + \frac{1}{2} M_0 \dot{V}_{g_0}^T \dot{V}_{g_0}. \tag{4}$$

Consider now the ith leg, with mass density ρ_i, section area A_i and inertia matrix ρI_i, all at section x. The external uncontrolled applied efforts (modelling e.g. gravity, a fluid resistance or a magnetic field in dedicated environments) at abcissa x are summarized in the vector $\left(f_i^T \,|\, c_i^T\right)^T$. Consider also the torque density Γ_i, for actuating the leg, and the corresponding generalized coordinate k_i, the curvature density. In the next section, either $\ddot{k}_i$ or Γ_i will be considered as unknown control hence introducing external energy to the system. The corresponding work is $\Gamma_i^T \ddot{k}_i$. It has been shown in [6] that the Appell function of section at x for one leg is:

$$R_i(x) = \left[\tfrac{1}{2} \begin{pmatrix} \ddot{r}_i^T & \dot{\omega}_i^T \end{pmatrix} \begin{pmatrix} \rho_i A_i \mathscr{I} & 0 \\ 0 & \rho_i I_i \end{pmatrix} \begin{pmatrix} \ddot{r}_i \\ \dot{\omega}_i \end{pmatrix} + \begin{pmatrix} -f_i \\ -c_i + (\omega_i \times (\rho_i I_i \omega_i)) \end{pmatrix}^T \begin{pmatrix} \ddot{r}_i \\ \dot{\omega}_i \end{pmatrix} - \Gamma_i^T \ddot{k}_i \right]. \tag{5}$$

As a scalar additive function, the Appell function of one leg is the integral: $\mathscr{R}_i = \int_0^1 R_i(x)dx$ and that of the whole parallel robot is then:

$$\mathscr{R} = \mathscr{R}_0 + \sum_{i=1}^{n} \mathscr{R}_i. \tag{6}$$

4 Inverse and Direct Dynamics as Solution of a Control Problem

One can now pose the optimization problem to be solved resulting from applying the Gauss principle of least constraint, while taking into account the kinematic constraints. To get compact formulations, define the following intermediate quantities:

$$\begin{aligned}
&\xi_i = \begin{pmatrix} \ddot{r}_i \\ \dot{\omega}_i \end{pmatrix}; \upsilon_i = \ddot{k}_i; F_i = \begin{pmatrix} 0 & -\hat{t}_i \\ 0 & \hat{k}_i \end{pmatrix}; G_i = \begin{pmatrix} 0 \\ \mathscr{I} \end{pmatrix}; h_i = \begin{pmatrix} \omega_i \times (\omega_i \times t_i) \\ \dot{k}_i \times \omega_i \end{pmatrix} \\
&\Sigma_i = \begin{pmatrix} \sigma_i \mathscr{I} & 0 \\ 0 & \tau_i \end{pmatrix}; b_i = \begin{pmatrix} -f_i \\ -c_i + (\omega_i \times (\rho_i I_i \omega_i)) \end{pmatrix}; c_i = -\Gamma_i; \sigma_i = \rho_i A_i; \tau_i = \rho_i I_i \\
&Q_i = \begin{pmatrix} \mathscr{I} & \hat{d}_i \end{pmatrix}; q_i = \omega_0 \times (\omega_0 \times d_i); p_i = \dot{V}_{g_0} + q_i \\
&\xi = (\xi_i)_i; \upsilon = (\upsilon_i)_i; h = (h_i)_i; b = (b_i)_i; c = (c_i)_i \\
&F = \mathrm{diag}(F_i); G = \mathrm{diag}(G_i); \Sigma = \mathrm{diag}(\Sigma_i); Q = diag(Q_i); q = (q_i)_i; p = (p_i)_i
\end{aligned} \tag{7}$$

where $\xi = (\xi_i)_i$ e.g. stands for the vector with components ξ_i, $i = 1, \ldots, n$ and so on. These are not merely a notational device. As one will see below, they make appear useful connections between Newton–Euler recursions in robotics and control theory

concepts, allowing for simple derivation and formulation of otherwise complex and tricky equations. Notice that the boundary value $\xi_i(1)$ is given by $(\dot{V}_{0,i}, \dot{\omega}_{0,i})$. With these notations, $\mathscr{R}_0$ (4) given as a quadratic function of $(\dot{V}_{g_0}, \dot{\omega}_0)$ is also, by a mere substitution, a quadratic function of $\xi(1)$: $\frac{1}{2}\xi^T(1)\tilde{Q}\xi(1) + T^T\xi(1) + e$ where the matrix $\tilde{Q}$, vector T and scalar e are not detailed here for saving space. The overall Appell function $\mathscr{R}$ (6) then writes:

$$\mathscr{R} = \left(\frac{1}{2}\xi^T(1)\tilde{Q}\xi(1) + T^T\xi(1) + e\right) + \int_0^1 (\frac{1}{2}\xi^T\Sigma\xi + b^T\xi + c^T\upsilon)dx. \tag{8}$$

The kinematics (1) of all the legs are gathered in:

$$\xi' = F\xi + G\upsilon + h \tag{9}$$

which is a dynamical equation in the space variable x. The connecting equations (3) write:

$$Q\xi(1) - p = 0. \tag{10}$$

Boundary conditions have to be added at the basis ($x = 0$): each leg is fixed to it by some joint thus the linear velocities vanish: $r_i(0,t) = \dot{r}_i(0,t) = \ddot{r}_i(0,t) \equiv 0$. As for the rotations at $x = 0$, when actuated their value is imposed; when unactuated the corresponding costate (defined below) vanishes. Eventually, using the previous notations, Gauss principle in Appell's formulation implies that the dynamical equations are the solution of the following optimization problem:

$$(\mathscr{P}) \quad \begin{cases} \min_\upsilon \mathscr{R} \\ \text{such that } \xi' = F\xi + G\upsilon + h \\ \text{and: } Q\xi(1) - p = 0 \end{cases} \tag{11}$$

together with boundary conditions at $x = 0$. This actually is an optimal control problem in the space variable. Although this problem has variable coefficients and is non standard, writing the first order necessary conditions is a routine task thus only the main steps of the procedure are given here for computing the optimal control (see [12, 13] for details). Introducing vector Lagrange multipliers (costate) λ and μ, define the hamiltonian:

$$H = \frac{1}{2}\xi^T\Sigma\xi + b^T\xi + c^T\upsilon + \lambda^T(F\xi + G\upsilon + h) \tag{12}$$

the function:

$$\Phi(\xi) = (\frac{1}{2}\xi^T\tilde{Q}\xi + T^T\xi + e) + \mu^T(Q\xi - p) \tag{13}$$

for the boundary part at $x = 1$ and the performance index:

$$J(\xi, \upsilon) = \Phi(\xi(1)) + \int_0^1 (H - \lambda^T \xi^{'}) dx. \tag{14}$$

Then, the optimization problem ($\mathscr{P}$) is equivalent to the minimization of J [12]. First order necessary conditions (that are also sufficient as ($\mathscr{P}$) is quadratic) for optimality then write:

$$\mathscr{N}.\mathscr{C}. \begin{cases} \xi^{'} & = \frac{\partial H}{\partial \lambda} = F\xi + G\upsilon + h \\ \lambda^{'} & = -\frac{\partial H}{\partial \xi} = -F^T\lambda - \Sigma\xi - b \\ 0 & = \frac{\partial H}{\partial \upsilon} = c + G^T\lambda \\ \xi_i(0) & = \xi_{i0} \text{ or } \lambda_i(0) = 0 \\ \lambda(1) & = \frac{\partial \Phi}{\partial \xi}_{|x=1} \\ \Phi(\xi(1)) & = 0 \end{cases} \tag{15}$$

As observed in [6], the condition $\frac{\partial H}{\partial \upsilon} = H_\upsilon = 0$ does not allow to explicit the optimal control, because υ appears linearly in the functional J. This makes the problem singular ([12], Chap. 8). But, differentiating H_υ twice with respect to x one gets after some computations the expression of the optimal control υ^*:

$$\upsilon^* = (G^T \Sigma G)^{-1}(K_1\lambda + K_2\xi + K_3) \tag{16}$$

with $G^T \Sigma G$ shown to be nonsingular and where K_1, K_2, K_3 are intermediate quantities that are not explicited here. (see [13]) for details on a single leg). A well-known, numerically efficient method of solution is the *sweep method* [12]. It uses the fact that, in the present linear-quadratic situation, the costate λ can be written as an affine function of the state: $\lambda = \zeta\xi + \kappa$. Then υ^* is computed thanks to the following two matrix-vector differential equations for ζ (a matrix *Riccati* equation) and κ [6]:

$$\begin{cases} \zeta^{'} + \zeta G(G^T \Sigma G)^{-1} K_1 \zeta + \zeta(F + G(G^T \Sigma G)^{-1} K_2) + F^T \zeta + \Sigma = 0 \\ \kappa^{'} + (F^T + \zeta G(G^T \Sigma G)^{-1} K_1)\kappa + \zeta h + \zeta G(G^T \Sigma G)^{-1} K_3 + b = 0 \end{cases} \tag{17}$$

that are integrated backward from $x = 1$ with the conditions: $\zeta(1) = \frac{\partial \Phi}{\partial \xi}_{|x=1}$ and $\kappa(1) = 0$. With all this material at hand, one is now able to explicit the *continuous Newton–Euler second order direct and inverse algorithms* for a hyper-redundant parallel robot, remarkably in a purely deductive way.

4.1 Direct Algorithm

The direct algorithm aims at finding the platform accelerations when torques at the legs are given. Notice that forces and torques will appear in a natural way as dual

variables (λ) of the accelerations in the optimization problem, as one expects in a variational context. Decompose λ as: $\lambda = (\lambda_i)_i$ with $\lambda_i^T = (n_i^T, M_i^T)^T$, in accordance with the decomposition of ξ_i. Firstly, writing down the optimality condition of the hamiltonian with respect to the control (third equation of (15)) gives: $-\Gamma_i + M_i = 0$, thus M_i is the distributed torque to be applied for getting a certain deformation of leg i, an expected interpretation. The direct algorithm is then:

1. *Data*: the distributed torques along all the legs, i.e. the costate λ , $\forall x \in [0, 1]$.
2. *Compute* the optimal control υ^* (16) as a function of ξ.
3. *Substitute* for υ^* into the state equation (9) which becomes an ODE.
4. *Integrate* forward the state equation (9) from $x = 0$ until $x = 1$ and get $\xi(1)$.
5. *Compute* the platform accelerations through: $p = Q\xi(1)$ and $\dot{\omega}_0 = \dot{\omega}_{0,i}$.

Thus, the direct algorithm is obtained straightforwardly in a purely deductive fashion, through one forward integration.

4.2 Inverse Algorithm

The inverse algorithm (*computed torque*) takes the accelerations of the platform as inputs and aims at giving the necessary torques of the legs as outputs. For parallel robots, it is known to be significantly more difficult to obtain than the direct algorithm and to ask for one recursion more. This is equally true in our context, as one will see. In the present continuous situation, it would be a really huge task, if even possible, to obtain an analogous algorithm when proceeding with a classical approach as in the discrete case. It is in that respect that the analogy with optimal control shows real power, making these derivations systematic and straightforward. The inverse algorithm is as follows:

1. *Data*: the platform accelerations, through p, giving $\xi(1)$ from (10): $Q\xi(1) = p$.
2. *Compute* the free parameter μ in order that $\Phi(\xi(1)) = 0$ is satisfied.
3. *Compute* the terminal condition $\zeta(1) = \frac{\partial \Phi}{\partial \xi}_{|x=1} = (\tilde{Q}\xi + T + Q^T\mu)_{|x=1}$.
4. *Solve* the two-point boundary value problem (15) along the following steps:
   a. *Integrate backward* the set of equations (17) from $x = 1$ until $x = 0$, with $\zeta(1)$ as computed at step 3 and $\kappa(1) = 0$.
   b. *Compute* the initial value $\lambda(0)$ through $\lambda(0) = \zeta(0)\xi(0) + \kappa(0)$
   c. *Integrate forward* the first two equations of (15) from $x = 0$ until $x = 1$.
5. *Output*: the accelerations and control torques all along the n legs.

Fundamental observation: the continuous second order Newton–Euler inverse recursion is nothing else than the sweep method for solving the two-point boundary value problem (15) coming from the constrained quadratic optimization problem ($\mathscr{P}$) (compare with [11]).

5 Conclusion

Gauss principle in Appell interpretation as a quadratic optimization problem in the accelerations is a powerful tool to derive in a systematic and deductive fashion the second order direct and inverse continuous recursions for complex robots such as a hyper-redundant parallel robot with actuated flexible legs presented here. It would certainly be a formidable task, if even possible, to obtain at once as here both Newton–Euler type recursions by standard means, especially for the direct algorithm, which is new and for which the only alternative seems to use finite element computations for solving the involved PDEs, of high computation cost. These results rely upon our reformulation of problems into the optimal control formalism. Such an approach to typical problems in robotics, even for atypical robots, allows for using this whole body of control theory, opening very interesting perspectives in robotics, for example at the preliminary design stage of complex robots. Obviously, the algorithms presented at the theoretical level in this work cannot be implemented as such, by brute force. Computer implementations have to take into account the special structure of involved matrices in order to take the whole benefit known for Newton–Euler recursions in the discrete case. Thus, further research will deal with computer implementation but also with control questions related to such complex robots. Nevertheless, implementations in high level, Matlab-like softwares appear straightforward as these include efficient built-in matrix Riccati equations solvers, a central computation node in our method, efficient ODE solvers and well tested control toolboxes. Thus, far from being only of theoretical interest, the formulation and resolution of complex robotics questions in the language of control theory allows for using efficient practical methods and software developed for other purposes, in a cross fertilization process of both domains.

References

1. Boyer, F., Khalil, W., Benosman, M., Le Vey, G.: Robot manipulators, modeling: performance, analysis and control. In: ISTE, vol. 7, pp. 337–394 (2007)
2. Dwivedy, S., Eberhard, P.: Dynamic analysis of flexible manipulators, a literature review. Mech. Mach. Theory **41**(7), 749–777 (2006)
3. Rodriguez, G.: Kalman filtering, smoothing and recursive robot arm forward and inverse dynamics. IEEE Trans. Robot. Autom. **3**(6), 624–639 (1987)
4. d'Eleuterio, G.M.T., Damaren, C.J.: The relationship between recursive multibody dynamics and discrete-time optimal control. IEEE Trans. Robot. Autom. **7**(6), 743–749 (1991)
5. Le Vey, G.: The Newton-Euler formalism for general multibody systems as the solution of an optimal control problem. In: Technical Report 05/4/AUTO, IRCCyN/Ecole des Mines de Nantes (2005)
6. Le Vey, G.: Optimal control theory and Newton-Euler formalism for cosserat beam theory. Comptes Rendus de l'Académie des Sciences de Paris, CR-Mécanique **334**, 170–175 (2006)
7. Cosserat, E., Cosserat, F.: Théorie des corps déformables. Hermann, Paris (1909)
8. Antman, S.S.: Nonlinear problems of elasticity. In: Applied Mathematical Sciences, vol. 107, 2nd edn. Springer, Berlin (2005)

9. Appell, P.: Traité de Mécanique Rationnelle. Gauthier-Villars, Paris (1921)
10. Baruh, H.: Analytical Dynamics. McGraw Hill, New York (1999)
11. Le Vey, G.: Dynamics and control of actuated parallel structures as a constrained optimization problem through gauss' principle and appell's equations. In: Proceedings of 2007 IEEE International Conference on Robotics and Automation, pp. 1480–1485. Roma (2007)
12. Bryson, A.E., Ho., Y.C.: Applied optimal control. In: Hemisphere Publishing Corporation, New York (1975) (Revised printing)
13. Le Vey, G.: Hyperredundant manipulators, continuous Newton-Euler algorithms and optimal control theory. In: Technical Report 05/3/AUTO, IRCCyN/Ecole des Mines de Nantes (2005)

Kinematic Design of Miura-Ori-Based Folding Structures Using the Screw Axis of a Relative Displacement

Kassim Abdul-Sater, Tim C. Lueth and Franz Irlinger

Abstract This chapter provides a kinematic design approach for specific folding structures, consisting of vertices with four intersecting creases, such that they can achieve two given folding configurations. These configurations are defined in terms of up to two planar curves, which are approximated by a polyline that is a particular part of the folding structure. We call these structures Miura-ori-based structures or linkages because the design approach makes use of the particular motion characteristics of the 1-DOF mechanism equivalent of the Miura-ori folding pattern. To achieve the design goal we apply a two-configuration synthesis, also provided in a previous work, which is based on the screw axis of a specific relative displacement. A classification of the slide along the screw axis allows it to determine the creases connecting consecutive links of the folding structure as revolute joints.

Keywords Screw axis · Linkage design · Dimensional synthesis · Origami · Miura-ori

1 Introduction

The class of folding patterns that remain moveable even if the numerous symmetrically arranged creases and chapter segments are replaced by revolute joints and rigid plates is known as rigid origami, [6]. A well-known example is the

K. Abdul-Sater (✉) · T. C. Lueth · F. Irlinger
Faculty of Mechanical Engineering, Institute of Micro Technology and Medical Device Technology, Technische Universität München, 85748 Garching, Germany
e-mail: kassim.abdul-sater@tum.de

T. C. Lueth
e-mail: tim.lueth@tum.de

F. Irlinger
e-mail: irlinger@tum.de

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_25,

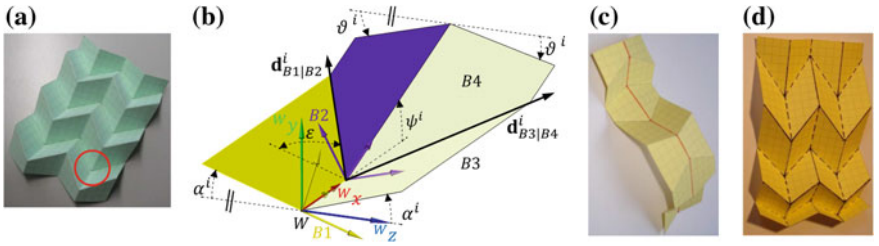


Fig. 1 **a** The Miura-ori folding pattern, **b** The Miura building block 4-bar, **c** Serial assembly of Miura building blocks, **d** A modified Miura pattern with different angles ε

Miura-ori pattern ([3], Fig. 1a), which moves with 1-DOF mobility and is used to fold maps or to deploy solid solar panels. Furthermore, this pattern is also considered to realize transformable roofs or foldable spaces in architecture (e.g. [5, 8]).

The Miura pattern belongs to the class of so called quadrilateral meshes because it is built from a recurring arrangement of four intersecting creases that form a special symmetric spherical 4-bar linkage, Fig. 1b. Tachi [6] provides conditions that allow it to change the kinematic dimensions among the hinges of quadrilateral meshes, while preserving rigid foldability. Based on this property of quadrilateral meshes we will use the kinematic properties of the building block linkage of the Miura pattern in combination with the screw axis of a specific relative displacement, to design structures that originate from the Miura pattern. The approach can be seen as a finite position synthesis approach, which allows it to determine a crease pattern so that it is able to reach two folding configurations. These are pre-defined by using a polyline of hinges that is a particular part of the folding structure. Thereby we extend some results that were reported in our previous work [1] and we aim to strengthen the bridge between rigid origami design and linkage design. Within Sects. 4 and 5 we address practical applications of the structures designed here.

2 Kinematic Analysis of Miura-Ori Building Blocks (Mbb)

The plane-symmetric halves of the Mbb shown in Fig. 1b satisfy

$$\alpha^i = -\vartheta^i \quad \text{and} \quad -\alpha^i = \vartheta^i, \quad i = 1, \ldots, n, \tag{1}$$

(The right superscript denotes a linkage configuration i).

To retrace the first relation, compute two different representations of the vector $\mathbf{d}^i_{B1|B2}$, measured in a fixed world frame W: $\mathbf{d}^i_{B1|B2} = \mathbf{R}^i_{B1}\mathbf{g}$ and $\mathbf{d}^i_{B1|B2} = \mathbf{R}^i_{B2}\mathbf{g}$. Herein $\mathbf{R}^i_{B1} = \mathbf{X}(\alpha^i) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix}^i$ (Rotation of frame $B1$), $\mathbf{R}^i_{B2} =$

$$\mathbf{Z}(\psi^i)\mathbf{X}(-\vartheta^i) = \begin{pmatrix} \cos\psi & -\cos\vartheta\sin\psi & -\sin\psi\sin\vartheta \\ \sin\psi & \cos\psi\cos\vartheta & \cos\psi\sin\vartheta \\ 0 & -\sin\vartheta & \cos\vartheta \end{pmatrix}^i \text{(Rotation of frame } B2\text{)}$$

and $\mathbf{g} = \begin{pmatrix} \sin\varepsilon & 0 & -\cos\varepsilon \end{pmatrix}^T$. The nomenclature $B1|B2$ denotes the coupling of the frames. W and $B1$ have their x-axes coinciding with a specific fixed crease; the angles α^i are measured from the dashed line coinciding with z-axis of W (Fig. 1b). The flat, links shall lie in the xz-planes of $B1$ and $B2$, which shall coincide with that of W in the unfolded configuration. Comparing the z-coordinates of the different representations yields $\cos\alpha^i \cos\varepsilon = \cos\vartheta^i \cos\varepsilon$ and we obtain the first relation of Eq. (1) because of the negative angle $-\vartheta$ in $\mathbf{R}^i_{B2}$. The second relation is obtained for the opposite half of Mbb when one uses $\mathbf{R}^i_{B3} = \mathbf{X}(-\alpha^i)$ and $\mathbf{R}^i_{B4} = \mathbf{Z}(\psi^i)\mathbf{X}(\vartheta^i)$ and then computes two expressions of $\mathbf{d}^i_{B3|B4}$ (Note that $B3$ and $B4$ are not shown in Fig. 1b).

From $\cos\alpha^i \cos\varepsilon = \cos\vartheta^i \cos\varepsilon$ we can conclude, that also larger couplings made of two or more Mbb assembled one after the other along the x-axis of W preserve Eq. (1) even if they have different angles ε. We call such an coupling a *serial assembly of Mbb* and it has hinges which form a polyline in the xy-plane of W (Fig. 1c). Compared to this type of coupling, in [4] a coupling of two spherical 4-bars is studied, whose axes share a common vertex. Because the edges of a serial assembly of Mbb move in planes that remain parallel to the xy-plane of W one can place several such structures along the z-axis and connect their edges using revolute joints. This idea may yield modified Miura-ori folding patterns, such as shown in Fig. 1d.

Another kinematic property of Mbb which will be needed in the following is the relation among α and ψ. This can be calculated using the constant scalar product $\langle \mathbf{d}^i_{B1|B2}, \mathbf{d}^i_{B2|B4} \rangle = \cos(\frac{\pi}{2} - \varepsilon) = \sin\varepsilon$. Because we have defined a symmetric folding process, we have $\mathbf{d}^i_{B1|B2} = \begin{pmatrix} \sin\varepsilon, \sin\alpha^i\cos\varepsilon, -\cos\alpha^i\cos\varepsilon \end{pmatrix}^T$ and $\mathbf{d}^i_{B2|B4} = \begin{pmatrix} \cos\psi^i, \sin\psi^i, 0 \end{pmatrix}^T$, which yields

$$\langle \mathbf{d}^i_{B1|B2}, \mathbf{d}^i_{B2|B4} \rangle = \cos\psi^i + k\sin\alpha^i\sin\psi^i = 1, \qquad k = \cot\varepsilon. \tag{2}$$

This can be seen as a constraint equation of Mbb because we obtain the same result for the symmetric counterpart of the structure. Then, the well known tan-half-angle solution yields the simple relation: $\tan\frac{\psi^i}{2} = k\sin\alpha^i$.

3 The Screw Axis of a Relative Displacement and a Two-Configuration Synthesis Method for C and R Joints

A spatial relative displacement can be written as

$$\mathbf{T}^{12}_B = \begin{pmatrix} \mathbf{R}^{12}_B & \mathbf{t}^{12}_B \\ 0\,0\,0 & 1 \end{pmatrix}, \text{ where } \mathbf{R}^{12}_B = \mathbf{R}^2_B \left(\mathbf{R}^1_B\right)^T \text{ and } \mathbf{t}^{12}_B = \mathbf{t}^2_B - \mathbf{R}^{12}_B \mathbf{t}^1_B. \tag{3}$$

(see [2]). $\mathbf{T}_B^{12}$ transports the moving frame B from an initial pose 1 to a goal pose 2. The 3×3 rotation matrix $\mathbf{R}_B^i$ may be constructed using a certain composition of coordinate rotations $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$, and $\mathbf{t}_B^i$ can be an arbitrary 3×1 translation vector.

The same displacement is obtained, if B performs screw motion about the *Plücker coordinate vector* of the *screw axis* $\hat{\mathbf{l}} = (\mathbf{d}, \mathbf{p} \times \mathbf{d})$, which satifies the equation:

$$\left(\hat{\mathbf{E}} - \hat{\mathbf{T}}_B^{12}\right) \hat{\mathbf{l}} = \hat{\mathbf{0}}, \tag{4}$$

where $\hat{\mathbf{E}}$ is a 6×6 identity matrix and $\hat{\mathbf{T}}_B^{12} = \begin{pmatrix} \mathbf{R}_B^{12} & \mathbf{0} \\ \tilde{\mathbf{t}}_B^{12} \mathbf{R}_B^{12} & \mathbf{R}_B^{12} \end{pmatrix}$ is a screw transformation matrix, where $\tilde{\mathbf{t}}_B^{12}$ is a skew symmetric matrix assembled from the parameters of $\mathbf{t}_B^{12}$. Solving (4) (see [2, p. 294]) yields the normalized direction $\mathbf{d}$ and the location $\mathbf{p}$ of $\hat{\mathbf{l}}$, measured in a fixed frame W. Note that we do not necessarily require the right superscript here since $\hat{\mathbf{l}}$ is not affected by $\hat{\mathbf{T}}_B^{12}$.

The translation or slide along $\hat{\mathbf{l}}$ shall be denoted as s^{12} and can be calculated by constructing a vector $(\mathbf{t}_B^{12})^{\parallel}$ which is parallel to $\mathbf{d}$: $(\mathbf{t}_B^{12})^{\parallel} = (\mathbf{t}_B^{12}\mathbf{d})\mathbf{d} = s^{12}\mathbf{d}$ (see e.g. [2, p. 285]). A slide $s^{12} \neq 0$ means a spatial motion and allows it to produce the movement $\mathbf{T}_B^{12}$ by a rotation around and the slide along $\hat{\mathbf{l}}$. $s^{12} = 0$ means a non-spatial motion and allows it to produce $\mathbf{T}_B^{12}$ by a pure rotation about $\hat{\mathbf{l}}$. These cases amount to a certain finite position synthesis approach, where the calculation of $\hat{\mathbf{l}}$ may yield the axis of a cylindrical (C) or either a revolute (R) joint, which satisfies the given spatial poses 1 and 2 of B.

By considering the coupling among two consecutive frames Ba and Bb, introduced by the axis $\hat{\mathbf{l}}_{Ba|Bb}$ of either a C or a R joint, this simple synthesis idea can be generalized. To see this, consider such a coupling described by two spatial relative displacements of the line $\hat{\mathbf{l}}_{Ba|Bb}$: $\hat{\mathbf{l}}_{Ba|Bb}^2 = \hat{\mathbf{T}}_{Ba}^{12}\, \hat{\mathbf{l}}_{Ba|Bb}^1$ and $\hat{\mathbf{l}}_{Ba|Bb}^2 = \hat{\mathbf{T}}_{Bb}^{12}\, \hat{\mathbf{l}}_{Ba|Bb}^1$. Rearranging yields the equation of the screw axis in terms of $\hat{\mathbf{l}}_{Ba|Bb}^1$:

$$\left(\hat{\mathbf{E}} - \hat{\mathbf{T}}_{Ba|Bb}^{12}\right) \hat{\mathbf{l}}_{Ba|Bb}^1 = \hat{\mathbf{0}}, \tag{5}$$

where

$$\hat{\mathbf{T}}_{Ba|Bb}^{12} = \left(\hat{\mathbf{T}}_{Ba}^{12}\right)^{-1} \hat{\mathbf{T}}_{Bb}^{12} = \begin{pmatrix} (\mathbf{R}_{Ba}^{12})^T \mathbf{R}_{Bb}^{12} & \mathbf{0} \\ (\mathbf{R}_{Ba}^{12})^T \left((\tilde{\mathbf{t}}_{Ba}^{12})^T + \tilde{\mathbf{t}}_{Bb}^{12}\right) \mathbf{R}_{Bb}^{12} & (\mathbf{R}_{Ba}^{12})^T \mathbf{R}_{Bb}^{12} \end{pmatrix}. \tag{6}$$

In order to classify, whether $\hat{\mathbf{l}}_{Ba|Bb}^1 = (\mathbf{d}_{Ba|Bb}^1, \mathbf{p}_{Ba|Bb}^1 \times \mathbf{d}_{Ba|Bb}^1)$ represents a C or a R joint, we can calculate the slide $s_{Ba|Bb}^{12}$ which is produced by the components of the skew symmetric translational submatrix of $\hat{\mathbf{T}}_{Ba|Bb}^{12}$. To obtain this matrix, we first introduce the matrix $\mathbf{R}^* = (\mathbf{R}_{Ba}^{12})^T \mathbf{R}_{Bb}^{12}$. Rearranging, so that $\mathbf{R}_{Bb}^{12} = \mathbf{R}_{Ba}^{12} \mathbf{R}^*$ and re-substituting into $\hat{\mathbf{T}}_{Ba|Bb}^{12}$ allows it to extract the skew symmetric matrix $\tilde{\mathbf{t}}^* = (\mathbf{R}_{Ba}^{12})^T \left((\tilde{\mathbf{t}}_{Ba}^{12})^T + \tilde{\mathbf{t}}_{Bb}^{12}\right) \mathbf{R}_{Ba}^{12}$, which contains the translation. Hence, we can

now assemble a vector $\mathbf{t}^*$ and compute $s^{12}_{Ba|Bb} = \mathbf{t}^* \mathbf{d}^1_{Ba|Bb}$ in order to check whether $\hat{\mathbf{l}}^1_{Ba|Bb}$ represents a C or a R joint. This discussion on the slide specifies our results from [1], where only a detailed and pictured derivation of Eq. (5) is provided.

If we consider a complete linkage, the approach described here may be used to synthesize the structure, such that it can be assembled in two pre-defined configurations, defined by two spatial poses of each link that make up the structure. For this reason we call the approach a two-configuration synthesis, which will be applied in the following to design the Miura-ori based folding structures.

4 Serial Assemblies of Mbb Patterns

Figure 2a shows a serial assembly of Mbb, in a flat unfolded configuration with frames $B1$ to $B(2m)$ located at the different vertices. The goal will be now to design such a structure, so that the lines $\hat{\mathbf{l}}_{B1|B(m+1)}, \ldots, \hat{\mathbf{l}}_{Bm|B(2m)}$ can approximate the shape of a specific predefined planar curve in the xy-plane of W. Figure 2b shows such a design with $m = 4$ in its folded configuration $i = 2$. The structure approximates one half of a parabola (red-colored) using the creases $\hat{\mathbf{l}}_{B1|B2}, \ldots, \hat{\mathbf{l}}_{B4|B8}$ and different angles ψ^2 at each vertex. The creases define a polyline of tangents to this curve with pre-defined length $l_1, \ldots, l_4$. This basic design idea was already described by others (see e.g. [5]). However, in order to achieve this goal here, we will use the results from Sects. 2 and 3, which will yield the dimensions of a structure.

In order to perform two-configuration synthesis, two poses of each link that make up a structure need to be pre-defined. For one half of a Mbb such as shown in Fig. 1b this yields relations $\alpha^i = -\vartheta^i$ and $\tan \frac{\psi^i}{2} = k \sin \alpha^i$, $i = 1, 2$. The second relation defines an overdetermined linear system for the dimensional parameter k, which shows that two configurations of a structure cannot be pre-defined in general. However, if we define $\alpha^1 = 0$ or $\psi^1 = 0$, the linear system degenerates and we obtain the condition $\psi^1 = 0$ or $\alpha^1 = 0$. This case defines the flat unfolded configuration and we can select arbitrary angles α^2 and ψ^2 and solve $\tan \frac{\psi^2}{2} = k \sin \alpha^2$ directly for k. This yields the direction $\mathbf{d}_{B1|B2}$ of the crease $\hat{\mathbf{l}}_{B1|B2}$ and the reflection at the xy-plane of W then yields a complete Mbb. To design a complete serial assembly of $m - 1$ such building blocks so that it can approximate the shape of a specific predefined planar curve, one may perform the upper procedure at each vertex. Note that in this case one cannot use absolute angles 'ψ', but has to calculate relative angles among $l_1, \ldots, l_m$.

Apart from this approach, applying the two-configuration synthesis from Sect. 3 provides a way to directly determine the Plücker coordinates $\hat{\mathbf{l}}_{B1|B2}$, $\hat{\mathbf{l}}_{B2|B3}$, …, $\hat{\mathbf{l}}_{B(2m-1)|B(2m)}$. As an example, we provide the synthesis procedure for the structure from Fig. 2b using the position data from Table 1. From this we can derive a general formula for the definition of a pose i of a certain frame Bj, $j = 2, \ldots, m$ in an assembly of $m - 1$ Mbb:

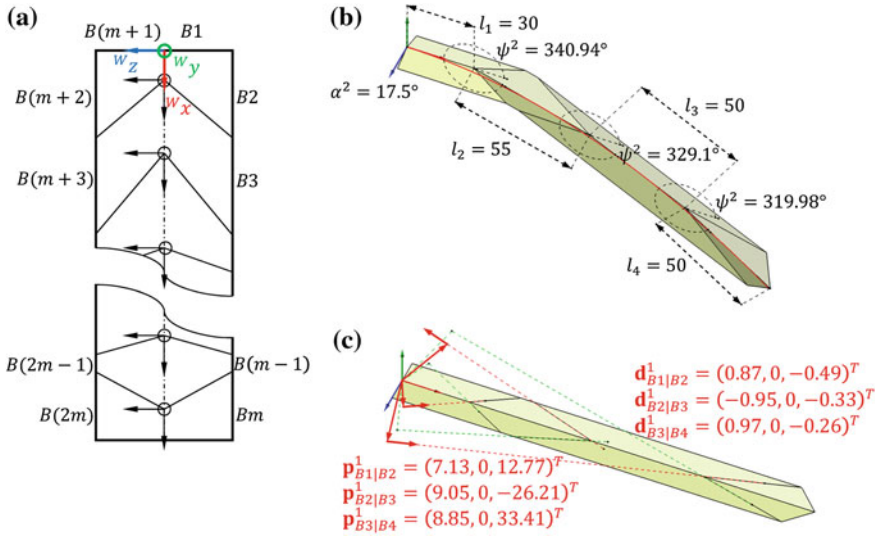


Fig. 2 **a** Assembly of a Miura-ori building block pattern with a flat unfolded configuration. **b** Folding configuration of a specific design which approximates the shape of a parabola. **c** Line coordinates of the creases $\hat{\mathbf{l}}^1_{B1|B2}$, $\hat{\mathbf{l}}^1_{B2|B3}$ and $\hat{\mathbf{l}}^1_{B3|B4}$ calculated using two-configuration synthesis and the position data from Table 1

Table 1 Position data for the frames $B1$ to $B4$, which form one half of a serial assembly of Mbb that approximates the shape of a parabola

	α^1	α^2	ψ^1	ψ^2	$\mathbf{t}^1_B$	$\mathbf{t}^2_B$
$B1$	0°	−17.5°	0°	0°	$(0\ 0\ 0)^T$	$(0\ 0\ 0)^T$
$B2$	0°	17.5°	0°	340.94°	$(l_1\ 0\ 0)^T$	$(l_1\ 0\ 0)^T$
$B3$	0°	−17.5°	0°	329.1°	$\mathbf{t}^1_{B2} + (l_2\ 0\ 0)^T$	$\mathbf{t}^2_{B2} + l_2(\cos\psi^2_{B2}\ \sin\psi^2_{B2}\ 0)^T$
$B4$	0°	17.5°	0°	319.98°	$\mathbf{t}^1_{B3} + (l_3\ 0\ 0)^T$	$\mathbf{t}^2_{B3} + l_3(\cos\psi^2_{B3}\ \sin\psi^2_{B3}\ 0)^T$

$$\mathbf{R}^i_{Bj} = \mathbf{Z}(\psi^i_{Bj})\mathbf{X}(-\alpha^i_{B(j-1)}), \qquad \mathbf{t}^i_{Bj} = \mathbf{t}^i_{B(j-1)} + l_{j-1}\begin{pmatrix}\cos\psi^i_{B(j-1)}\\ \sin\psi^i_{B(j-1)}\\ 0\end{pmatrix} \tag{7}$$

For $B1$ we can add here $\mathbf{R}^i_{B1} = \mathbf{X}(\alpha^i_{B1})$ and $\mathbf{t}^i_{B1} = \mathbf{0}$ to have a complete description. This allows it to compute the creases $\hat{\mathbf{l}}_{B1|B2}$, $\hat{\mathbf{l}}_{B2|B3}$, …, $\hat{\mathbf{l}}_{B(m-1)|Bm}$ using Eq. (5). Afterwards the other half of the structure, defined by $\hat{\mathbf{l}}_{B(m+1)|B(m+2)}$, …, $\hat{\mathbf{l}}_{B2(m-1)|B(2m)}$ is found by a reflection at the xy-plane of W.

The example procedure for the structure with $m = 4$ yielded the lines $\hat{\mathbf{l}}^1_{B1|B2}$, $\hat{\mathbf{l}}^1_{B2|B3}$ and $\hat{\mathbf{l}}^1_{B3|B4}$. The results are shown in Fig. 2c and one can see, that the Plücker coordinates have y-components equal to zero, which corresponds to the unfolded configuration. Furthermore, we computed the slides along the different creases, which

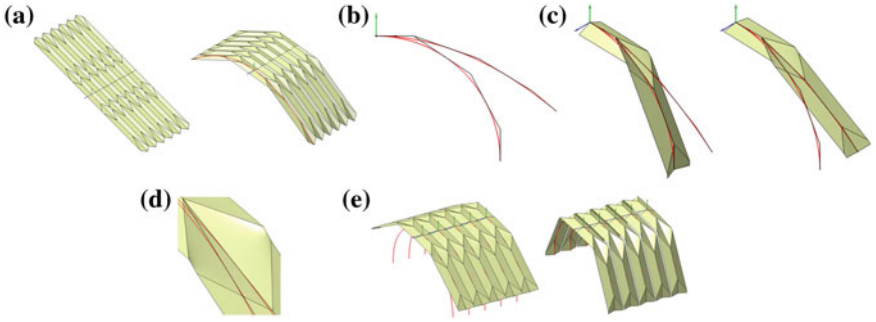


Fig. 3 **a** A number of serial assemblies of Mbb forming a transformable closed roof, **b** two planar curves approximated by a polygonal chain, **c** folding configurations of the non-flat serial assembly of Miura-based spherical 4-bars, **d** non-flat design of links, induced by hinges that do not lie in a plane, **e** a number of non-flat serial assemblies forming a transformable roof, which is closed in configuration 1 (*left*)

yielded $s^{12}_{B1|B2} = s^{12}_{B2|B3} = s^{12}_{B3|B4} = 0$. This shows that the structure could be build up using revolute joints, which corresponds to origami patterns, where no sliding is observed at the creases. However, for a designer the results may be confusing because the lines $\hat{\mathbf{l}}^1_{B1|B2}$ and $\hat{\mathbf{l}}^1_{B3|B4}$ define their corresponding creases at the opposite half of the pattern. Furthermore, directions can point in the opposite direction of the actual crease, which is the case for $\hat{\mathbf{l}}^1_{B2|B3}$.

Figure 3a shows a complete Miura-based mesh, build from several serial assemblies and their mirrored counterparts that are hinged at their common edges. This forms a closed moveable structure, which could be used as a transformable roof.

5 Non-flat Serial Assemblies of Miura-Based Spherical 4-Bars

If we ignore the condition $\alpha^1 = 0$ or $\psi^1 = 0$ from Sect. 4 and set $\{\alpha^1, \alpha^2\} \neq 0$ and $\{\psi^1, \psi^2\} \neq 0$ in a two-configuration design procedure, a Mbb will not be able to fulfil the desired task. However, an appropriate modification of the structural dimensions may enable the resulting structure to reach the desired configurations. This refers to another spherical 4-bar, which is still plane-symmetric to the xy-plane of W but cannot achieve a flat unfolded configuration.

Compared to the original Mbb a 4-bar without a flat unfolded configuration means that we obtain a more complex constraint equation (see Sect. 2). Hence, we obtain a more complex tan-half-angle solution, which does not allow a simple solution for the dimensional parameters as shown in Sect. 4. However, if we use the rotations $\mathbf{R}^i_{B1}$, $\mathbf{R}^i_{B2}$, $\{\alpha^1, \alpha^2\} \neq 0$ and $\{\psi^1, \psi^2\} \neq 0$, and the relation $\alpha^i = -\vartheta^i$ already introduced in Sect. 2 to define two poses $i = 1, 2$, we will find the axis $\hat{\mathbf{l}}^1_{B1|B2}$ of a revolute joint using two-configuration synthesis from Sect. 3. This is because we can

compute $s^{12}_{B1|B2} = 0$ and we find the axis $\hat{\mathbf{l}}^{1}_{B3|B4}$ of the symmetric counterpart by a reflection at the xy-plane of W.

This approach also holds for a serial assembly of such structures and finally allows it to approximate the shape of two different planar curves in the xy-plane of W. An example is shown in Fig. 3b, c, where a polyline of pre-defined length is used to define tangents to a parabola (configuration 1) and a circle (configuration 2). To perform the synthesis procedure of the complete structure we used Eq. (7) to pre-define poses. Here we used position data obtained from the polyline in both configurations in combination with arbitraily chosen angles α^i, $i = 1, 2$.

The different links of the structure cannot be designed as flat plates because the hinges in a link do not lie in one plane anymore (see Fig. 3d). Even though this means that we cannot hinge two or more adjacent serial assemblies, an isolated actuation of each serial assembly is still possible, which could yield a transformable roof that can be closed in one configuration (Fig. 3e). Another speculative application could be grasping devices, i.e. robotic fingers formed from serial assemblies.

6 Conclusion

In this chapter the kinematic design method from [1] for C or R joints is used to compute the axes of R joints of a class of rigid folding structures for two pre-defined configurations. These configurations are defined in terms of a polyline, which is a particular part of the folding structure and which is used to approximate up to two pre-defined planar curves. Furthermore, for the design method we provide a criterion to clearly distinguish between R and C joints, which may also be useful for the design of folding structures with non-zero plate thickness. This is because this case may require C joints [8]. As an example, it can happen that a spherical 4-bar mechanism equivalent of a non-symmetric single vertex pattern with non-zero thickness needs to degenerate to a spatial RCCC linkage to preserve 1DOF mobility. Our results can also be seen in the context of results such as Tachi [7], where rigid-foldable, so called tubular structures are designed for a given space curve.

Acknowledgments The author expresses his gratitude to Dr. Leri Datashvili from the institute of lightweight structures at the Technische Universität München for several fruitful discussions and supervision in an early stage of this work.

References

1. Abdul-Sater, K., Irlinger, F., Lueth, T.C.: Two-configuration synthesis of origami-guided planar, spherical and spatial revolute-revolute chains. In: McCarthy J.M. (ed.) ASME J. Mech. Robot. **5**(3), 10 (2013)
2. McCarthy, J.M., Soh, G.S.: Geometric design of linkages. Springer Interdisciplinary Applied Mathematics, 2nd edn. Springer, New York (2010)

3. Miura, K.: Method of packaging and deployment of large membranes in space. In: Science Report No. 618, Institute of space and astronautical science (1985)
4. Nawratil, G., Stachel, H.: Composition of spherical four-bar-mechanisms. In: Pisla D. et al. (eds.) New Trends in Mechanism Science, pp. 99–106. Springer, New York. ISBN 978-90-481-9688-3 (2010)
5. Stavric, M., Wiltsche, A.: Investigations on quadrilateral patterns for rigid folding structures. In: Stouffs R., Janssen P., Roudavski S., Tuncer B. (eds.) Open Systems: Proceedings of the 18th International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA 2013), pp. 893–902 (2013)
6. Tachi, T.: Generalization of rigid foldable quadrilateral mesh origami. In: Domingo A., Lazaro C. (eds.) Proceedings of the International Association for Shell and Spatial Structures (2009)
7. Tachi, T.: Composite rigid-foldable curved origami structure. In: Escrig F., Sanchez J. (eds.) Proceedings of the 1st Conference 'Transformables' (2013)
8. Trautz, M., Künstler, A.: Deployable folded plate structures folding patterns based on 4-fold-mechanism using stiff plates. In: Proceedings of the International Association for Shell and Spatial Structures (IASS), symposium, Valencia, (2009)

On the Limitations on the Lower and Upper Tensions for Cable-Driven Parallel Robots

Andreas Pott

Abstract In this chapter we discuss the feasible range for the cable forces of cable-driven parallel robots. Numerous papers on algorithms have been presented that assume certain values for the minimum and maximum cable forces. In this contribution we summarize criteria to derive the tension limits from kinematic, physical, and technological parameters point of view. We discuss a number of effects leading to the interval of feasible cable forces and provide formula and technical parameters that allow the determination of the limits for the cable forces. Typical values for the cable force limits are exemplified based on use-cases of different robots of the IPAnema system family.

Keywords Cable-driven parallel robots · Cable tension · Limits · Sagging

1 Introduction

Cable-driven parallel robots are a special class of parallel robots where the rigid legs are replaced by light weight steel or synthetic fibre cables. The use of cables leads to unilateral constraints, i.e. the robot can only pull but not push the platform or mathematically speaking the tension f in the cables must be nonnegative.

As carried out by a number of authors [16] a minimum tension $f_{\min}$ must be maintained to hinder the cables from being slack and a maximum tension $f_{\max}$ must not be violated to prevent cables and motors from overload. Although there is a rich development for algorithms to compute tension distributions for given limits [5, 7, 14], little was published on how to derive the actual limits. In practise it turns out that there are many more limiting factors both for the minimum and maximum force

A. Pott (✉)
Fraunhofer IPA, Stuttgart, Germany
e-mail: andreas.pott@ipa.fraunhofer.de

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_26,

which are hardly discussed in the literature. Therefore, we dedicate this contribution to review and quantify these limits.

Analysis, design, and control of cable robots are mostly based on assuming straight and inelastic cables. Typical issues to be solved are direct and inverse kinematics, workspace computation [2, 16], tension distribution [1, 14], control, or calibration of cable-driven parallel robots. During the recent years it becomes clear that such assumptions do not hold under some circumstances such as large-scale systems like CoGiRo [9] or FAST [10]. Therefore, it is a matter of design and configuration of the robot to choose the tension limits such that the error induced by the simplifications on the model used for the real-time control system is acceptable.

2 Upper Bounds on the Tension

In the following we briefly recall the concept to characterize a pose of the robot that is based on the tension in the cables. A pose $(\mathbf{r}, \mathbf{R})$ is called *wrench-feasible* [3, 6] or *acceptable* [16] for a given wrench $\mathbf{w}$ if

$$\mathbf{A}^{\mathrm{T}}(\mathbf{r}, \mathbf{R})\mathbf{f} + \mathbf{w} = \mathbf{0} \quad \text{for } f_{\max} \geq f_i \geq f_{\min} > 0, \quad i \in [1, m], \tag{1}$$

where $\mathbf{A}^{\mathrm{T}}$ is the structure matrix (wrench matrix), $\mathbf{w}$ is the vector of wrench applied to the platform, and $\mathbf{f} \in \Re^m$ is the vector of cable forces. Clearly, more poses are wrench-feasible and thus the workspace is larger if the interval between $f_{\min}$ and $f_{\max}$ is wider. In the following we will investigate criteria leading to restriction in the cable forces.

As pointed out above, the lower and upper limits on the forces in the cables must be taken into account when considering the wrench-feasibility of a pose. This section will investigate what effects must be taken into account when choosing the upper $f_{\max}$ bound for the cable force. To determine the maximum tension in the cables the following effects must be considered.

Safety and mechanical limits: An obvious limitation is the breaking load of the cable. Clearly, if the tension exceeds the cable's breaking load, the robot cannot be safely operated. For real cables one has to additionally take into account a safety factor where relatively high factors are typical in the magnitude around 10 in lifting applications such as cranes, elevators, cable cars, and thrill rides.

We have to take into account that the breaking load of the cable applies to ideally applied forces at the end of the cable. Fixing the cable by clamping, inappropriate knots, and too small bending radii, reduces significantly the cable's payload, in some cases such as clamping and bending to less than 2 % (!) of its nominal load. This effect can be seen from the evaluation of the cable's breaking load in Fig. 1.

Another mechanical limitation is the maximum load on the winches and pulley mechanisms. The forces in the cable must not damage the mechanical structures. In typical applications the frame, winches, and pulleys are appropriately dimensioned after the maximum tension of the cables was chosen. As a matter of safety, the

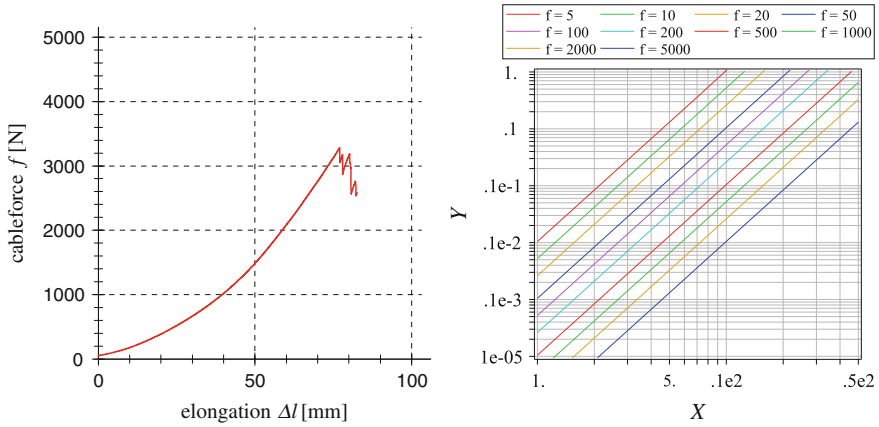


Fig. 1 *Left* measured cable force f [N] over the absolute elastic elongation Δl [mm]. *Right* computed sagging s in the middle of a horizontal cable over distances l in logarithmic scaling. Different cable forces f. Considered cable: Dyneema cable D-PRO, diameter $d = 2.5$ mm, probe length $l_0 = 300$ mm, specified breaking load $f_{spec,\max} = 5{,}800$ N, measured breaking load 3,300 N

cable's breaking load is chosen to be at least twice that of the nominal load of the winches. Therefore, overload will potentially immobilize the winch rather than cut the connection between winch and platform.

The maximum forces exerted by cables on the mobile platform should not deform or damage the platform. This is important for robots that can be reconfigured by exchanging the platform. Different platforms may be designed for largely different processes e.g. inspection and handling. A light-weight sensor platform may get destroyed when applying crane like forces that are necessary for the handling operation. This effect is mostly a matter of the configuration of the controller and the safety system.

Fatigue: The maximum feasible tension considering fatigue heavily differs from the static forces. Even if the braking load of a certain cable may be very high, one has to take into account the conditions under which the robot is operated. While the robot moves along a trajectory the cable gets bent every time it is guided around a pulley or coiled onto the drum. For linear actuators with pulley mechanisms each motion cycle causes a multiple of bending cycles in the cable. Given a certain lifetime for the cable, the maximum feasible tension in the cable must be chosen to avoid failure due to fatigue. Since the fatigue of different cables of the same type may heavily vary, a second safety factor must be chosen to care for the statistic variations. The first factor depends on mechanical and material properties of the cable, while the second safety factor additionally depends on the level of safety required for the targeted application. Today, figures on fatigue can only be determined by empirical studies [4], while models and simulations such as finite elements method are not yet reliable enough to compute such data without experimental validation.

Actuators: The maximum force of the motor often puts the effective upper limit of the cable force since the actuators are usually the most costly component in the robot system. If the robot has an emergency braking system, one has to additionally consider the maximum force that can be generated by the brakes. In case of an emergency stop, the forces generated by the brakes may generate a severe danger for the cables to break down. A general method to safely predict upper limits on the braking forces is still an open problem.

Additionally, one has to consider a safety margin caused by control errors. The width of the safety margin depends on the quality of the control system.

Summary: Determining the maximum cable force is mostly a matter of mechanical design. Since the costs for the drive-train presents often the highest costs in the robot design (around 50 %) this is the limiting factor in most cases. More precisely, the maximum force generated by the motor should be the limit because the other criteria presented above have to be chosen in an appropriate relation to the motor forces. However, if we reconfigure a cable robot and reuse a winch with a different cable, the effective maximum force may be determined from the other criteria described in this section.

3 Lower Bounds on the Tension

The lower limit on the cable force is considered in the this section. On first sight one might consider any positive tension feasible. In practice the following effects require the cables to be under a minimum force.

Slackness: Ensuring pretension and thus preventing the cables from *slackness* is a premise to assume straight cables. However, every cable is subject to sagging under the effect of gravity [8]. If the tension in the cables is too low, sagging of the cable cannot be neglected. This is often not acceptable because the effective length of the cable and the distance between A_i and B_i heavily differs in this case. In the presence of large sagging the real direction of the cable force vector differs largely from the ideal direction. Furthermore, the robot becomes unsensitive to control changes in case of slack cables: Changes in the cable length lead to smaller changes in the cable tension. Therefore, slack cables can hardly be used to control the motion of the mobile platform. To find the minimum cable tension one has to define the maximum error for the cable length or the maximum displacement from the ideal linear form. We consider the minimum tension required to keep the maximum sagging below a given upper bound and the minimum tension required to bound the deviation in the real and effective length by an upper limit.

The maximum sagging s for a horizontal cable with a distance between the ends of length l is

$$s = \frac{H}{g}\left(\cosh\left(\frac{gl}{2H}\right) - 1\right), \tag{2}$$

where H is the horizontal cable force and g is the gravity force caused by the cable's mass per length. Unfortunately, it is difficult to solve for the sought force H but a double logarithmic plot reveals simple figures (Fig. 1) and we can easily pick the required pre-tension $f_{\min}$ for a given length l, a given cable mass g, and an acceptable sagging of $s_{\max}$ either from diagrams or by numerical solving the implicit equation.

Considering the length error is straightforward using the equation of the sagging cable. Computing the actual length l_s of the cable from the catenary line can be solved in closed-form and we receive the simple equation

$$l_s = \frac{2H}{g} \sinh\left(\frac{gl}{2H}\right). \tag{3}$$

For selecting the minimal force we have to face the same situation as above. While the equation for length can be written in closed-form, we cannot find a closed from express for H from Eq. (3).

Even if sagging is acceptable there is a lower bound for the cable force. The minimum tension depends on the weight of the cable and the tension cannot be reduced below a value that is coupled to the density and length of the cable. Consider the following situation: The distal end of the cable is fixed in space and the cable is uncoiled starting from a perfect line. Then the additional length of the cable uncoiled from the winch contributes to the effective weight that is acting on its distal anchor point. Therefore, a lower bound on the cable force depends on the density of the cable. In between, we find a minimal positive cable force than cannot be undercut. This limit depends on the distance between the proximal and distal anchor points as well as on the density of the cable.

Reliable Operation: Pretension is required for reliable operation of the winch. If the tension of the cable is very low the bending stiffness of the cable cannot be neglected. The effect is more important for steel cables but even fibre ropes have a finite bending stiffness that may cause uncontrollable coiling errors in the winches. Also for steel cables a very low tension may cause the cables to leave the pulleys or drum grooves. Beside coiling errors that might affect the accuracy of the robot, the cable might leave the guiding pulleys which in turn causes severe safety problems.

In the field of stage equipment 1–2 % of the cables breaking load is desirable as minimal tension for coiling.

Elasticity: The elastic elongation of the cables is in general nonlinear (see Fig. 1), even if there is a nearly linear region around the operational point. To operate the cables within this desired interval, a certain pretension is required and at the same time a maximum tension may not be exceeded to stay within this preferred state. Since little compensation techniques for nonlinear elongation of the cables can be found in the literature, this presents a relevant limitation on the maximum ratio between $f_{\min}$ and $f_{\max}$.

Vibration: Tensions in the cables can reduce the vibration of the cables by increasing the cable's eigenfrequency. This effect is well known from string instruments. It was also shown that pretension can be used to influence the vibration of the platform

of planar robots [15]. For an ideal string, the first eigenfrequency λ can be computed from basic physics

$$\lambda = \frac{1}{2l}\sqrt{\frac{f}{\rho A}}, \tag{4}$$

where ρ is the density of the cable and A is the cross section. We can easily compute the tension f for a desired minimum eigenfrequency λ

$$f(\lambda) = 4\lambda^2 l^2 \rho A. \tag{5}$$

To achieve higher eigenfrequencies for large robots, high pretension is required.

Summary: When using force sensors to measure the tension in the cables it might be necessary to maintain a minimum tension, since some force sensors provide low quality signals close to zero tension. For very small forces the measurement is also subject to errors caused by mechanical parts for including the sensors. Therefore, small cable forces are hard to distinguish from friction. Also for the lower cable forces $f_{\min}$ one has to consider a safety margin in order to prevent control errors to trigger one of the effects listed above.

Taking all these above mentioned effects into account, it becomes apparent that the ratio between minimum and maximum cable force may be significant but cannot be increased to arbitrary values.

4 Examples

The effective limits of the cable forces depend on a large number of physical parameters, design decisions, and application requirements. Since a number of components such as motors, cable material and diameter are subject to a long design procedure, we consider these parameters to be given. In the following the considerations on the cable force limits are exemplified based on case-studies performed for some prototypes, i.e. the IPAnema family prototypes [13]. In particular, we consider the use-cases described in Table 1 which are applied in the following robot setup:

- IPAnema 1 spatial system, medium size robot for a fast pick-and-place tasks.
- IPAnema 2 spatial system, medium size robot for handling of solar collectors.
- IPAnema 3 mini spatial system, small size robot, high-dynamics laboratory system for testing of kinematic codes, control algorithms, and calibration.
- IPAnema 3 spatial system, large size robot (>16 × 6 × 5 m), 5.0 kW for handling in a logistics scenario.

Beside the actual scenario we consider additionally the nominal design parameter of the IPAnema 3 winches for 2.5 mm cable and 6.0 mm cable.

In order to compute feasible tensions we have to fix the requirements. We checked the following criteria: The maximum sagging s shall be smaller that 0.1 % of the

Table 1 Overview of some use-cases of the IPAnema robot family

Robot	Frame			Winch	Cable			
	Length	Width	Height	Power	Material	Diameter	Weight/ length	Breaking load
	(m)	(m)	(m)	(kW)		(mm)	(kg/m)	(N)
IPAnema 1	4	3	2	1.8	Dyneema	2.5	0.0035	5,800
IPAnema 2	8	6	5.5	1.8	Dyneema	2.5	0.0035	5,800
IPAnema 3 mini	1	1	1	0.2	Dyneema	1.5	0.0013	2,300
IPAnema 3 planar	20	5	0	5.0	Dyneema	6	0.023	43,000
IPAnema 3 spatial	16	6	5	5.0	Steel	6	0.129	21,100
IPAnema 3 winch A	45	0	0	7.5	Dyneema	2.5	0.0035	5,800
IPAnema 3 winch B	23	0	0	7.5	Dyneema	6	0.023	43,000

Table 2 Tension limits for the IPAnema use-cases

Robot	Nominal (given) f_{max} (N)	Sagging $s < l/1000$ f_{min} (N)	Sagging $s < d$ f_{min} (N)	Length error $\Delta l < l/1000$ f_{min} (N)	Length error $\Delta l < 1\,\text{mm}$ f_{min} (N)	Eigenfrequency $f_E > 10\,\text{Hz}$ f_{min} (N)
IPAnema 1	180	2.6	5.1	1.2	2.8	40.6
IPAnema 2	720	5.0	22.8	2.5	8.5	182.4
IPAnema 3 mini	60	2.8	3.2	0.1	0.2	1.6
IPAnema 3 planar	3,000	59.3	203.7	26.3	136.4	3910.0
IPAnema 3 IZS setup	3,000	287.0	851.5	26.0	613.2	16357.2
IPAnema 3 winch A	580	192.9	3472.0	10.0	66.9	2835.0
IPAnema 3 winch B	4,000	648.6	2486.4	33.5	160.6	4866.8

length of the cable or smaller than the cable's diameter (columns 3 and 4). The length error caused by the sagging shall be smaller than 0.1 % of the cable length or smaller than 1 mm (columns 5 and 6) and the eigenfrequency of the cable should be at least 10 Hz (column 7). The results are given in absolute numbers in Table 2.

5 Conclusions

In this chapter we discussed a number of technical issues that influence the feasible minimum and maximum tensions for the cables of cable-driven parallel robots. Taking all these issues into account can lead to a surprisingly small intervals for the feasible cable forces in some applications. In the most optimistic use-case we

found a ratio between minimum and maximum tension of 600. Considering strict requirements on sagging, larger application typically allow for a smaller ratio and thus for less workspace compared to the theoretical limits. However, since cable robots are evolving towards commercial applications such limitation needs to be taken in account to allow for safe and reliable operation. Furthermore, some criteria discussed in the chapter are pose dependant. To the best of the authors knowledge, it was not yet considered in literature to adjust the cable force limits to the pose e.g. for workspace computation or for control. Relaxing and tightening these bounds may have either positive or negative influence on the robots workspace since it can present additional potentials at some poses as well as the need for higher pretension in other poses.

Acknowledgments This work has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 285404.

References

1. Borgstrom, P.H., Jordan, B.L., Sukhatme, G.S., Batalin, M.A., Kaiser, W.J.: Rapid computation of optimally safe tension distributions for parallel cable-driven robots. IEEE Trans. Robot. **25**(6), 1271–1281 (2009)
2. Bouchard, S., Moore, B., Gosselin, C.: On the ability of a cable-driven robot to generate a prescribed set of wrenches. J. Mech. Robot. **2**(1), 1–10 (2010)
3. Ebert-Uphoff, I., Voglewede, P.A.: On the connections between cable-driven parallel manipulators and grasping. In: IEEE International Conference on Robotics and Automation, pp. 4521–4526. New Orleans (2004)
4. Feyrer, K.: Drahtseile: Bemessung, Betrieb. Springer, Sicherheit (2000)
5. Gosselin, C.: On the determination of the force distribution in overconstrained cable-driven parallel mechanisms. In: Proceedings of the Second International Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators, pp. 9–17. Montpellier, France (2008)
6. Gouttefarde, M., Merlet, J.P., Daney, D.: Wrench-feasible workspace of parallel cable-driven mechanisms. In: ICRA, pp. 1492–1497. Rome, Italy (2007)
7. Hassan, M., Khajepour, A.: Minimum-norm solution for the actuator forces in cable-based parallel manipulators based on convex optimization. In: ICRA, pp. 1498–1503 (2007).
8. Kozak, C., Zhou, Q., Wang, J.: Static analysis of cable-driven manipulators with non-negligible cable mass. IEEE Trans. Robot. **22**(3), 425–433 (2006)
9. Lamaury, J., Gouttefarde, M.: Control of a large redundantly actuated cable-suspended parallel robot. In: IEEE International Conference on Robotics and Automation, pp. 4659–4664 (2013)
10. Li, H., Zhang, X., Yao, R., Sun, J., Pan, G., Zhu, W.: Optimal force distribution based on slack rope model in the incompletely constrained cable-driven parallel mechanism of FAST telescope. In: Cable-driven Parallel Robots, pp. 87–102. Springer, Berlin (2013)
11. Mikelsons, L., Bruckmann, T., Schramm, D., Hiller, M.: A Real-time capable force calculation algorithm for redundant tendon-based parallel manipulators. In: ICRA. Pasadena (2008)
12. Oh, S.R., Agrawal, S.K.: Cable suspended planar robots with redundant cables: controllers with positive tensions. In: IEEE Transactions on Robotics (2005)
13. Pott, A.: Influence of pulley kinematics on cable-driven parallel robots. In: Latest Advances in Robot Kinematics, pp. 197–204. Springer, Berlin (2012)
14. Pott, A., Bruckmann, T., Mikelsons, L.: Closed-form force distribution for parallel wire robots. In: Computational Kinematics, pp. 25–34. Springer, Berlin (2009)

15. Surdilovic, D., Radojicic, J., Krüger, J.: Geometric stiffness analysis of wire robots: a mechanical approach. In: Cable-driven Parallel Robots, pp. 389–404. Springer, Berlin (2013)
16. Verhoeven, R.: Analysis of the workspace of tendon-based Stewart platforms. Ph.D. thesis, University of Duisburg-Essen, Duisburg (2004)

Characterization of the Subsystems in the General Three-System of Screws

Marco Carricato and Dimiter Zlatanov

Abstract The chapter examines the subspaces of a space spanned by three twists (or wrenches) with finite pitches and linearly independent directions. The main result is the complete characterization of all cylindroids nested within a general three-system of screws. The findings are illustrated by means of the prestereographic model of three-dimensional projective space, which allows to faithfully represent the screws of a three-system, with their pitches and axis locations, by the surface of a sphere.

Keywords Screw systems · Projective space · Mechanism synthesis

1 Introduction

In robotic systems, the possible instantaneous motions of a rigid body, or the systems of forces acting on it, are described by a subspace of the six-dimensional vector space of twists, or wrenches. Such linear subspaces, or the underlying projective spaces, are referred to as screw systems. There is a great variety of screw systems, because subspaces with the same dimension, but spanned by different screws, cannot in general be transformed in one another by a rigid displacement. Screw systems were first studied in [1], but a comprehensive classification was obtained in [5] (re-derived in [4], see also [3, 8]). Hunt first described in detail how the locations and pitches of the screws vary within systems of different types [5] (see also [7]).

Two screw systems are *equivalent* if one is a rigid displacement of the other. This equivalence relation divides the space of linear subspaces of $se(3)$ into infinitely many

M. Carricato (✉)
Department of Industrial Engineering and Interdepartmental Center for Health Sciences and Technologies, University of Bologna, Bologna, Italy
e-mail: marco.carricato@unibo.it

D. Zlatanov
University of Genoa, Genoa, Italy
e-mail: zlatanov@dimec.unige.it

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_27,

(nonintersecting) *classes*. Geometrically similar classes are grouped into *types*: one general and a variety of special, according to [5]. Following Gibson and Hunt [4] a screw-system class can be labeled by: its dimension; I or II, indicating whether or not there are screws of more than one finite pitch; a letter (from A to D) denoting the number (from 0 to 3) of independent infinite-pitch screws; and additional parameters. In this chapter we study three-system A-types, including the general three-system class 3–IA(h_x, h_y, h_z) with principal pitches h_x, h_y, and h_z.

One aspect of the geometry of screw systems, which has not been explored in detail, is the nesting of lower-rank systems within spaces of higher dimension. The key issue is the characterization of all two-systems within a three-system. (Nesting in higher-rank systems reduces to lower dimensions via reciprocity.) One can ask: what classes of two systems exist in any given three-system class, and where are the representatives of each class located? Such questions are of theoretical importance and have practical relevance in the design of mechanisms as they relate to the type-synthesis of serial chains with a desired mobility space of the end link [2, 6].

The present chapter computes explicitly (Sect. 4) the 2-subspaces of a general 3-system, which is spanned by three screws with concurrent and perpendicular axes and different finite pitches. The characterizations of the nesting subspaces of two special cases, when two or all the principal pitches are equal, are also derived. Nesting in three-systems containing infinite-pitch screws is discussed in [10].

2 Screws and Projective Spaces

A twist (or a wrench) is given by a pair of 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 at the origin). An element, $\boldsymbol{\xi} = (\boldsymbol{\omega}, \mathbf{v})$, of $se(3)$ (or $se(3)^*$) is associated with a *screw* about which the body twists (or the wrench is applied)—a line in space, $\ell(\boldsymbol{\xi})$, the screw axis, with a metric scalar, the pitch h, given by

$$h = \frac{\boldsymbol{\omega} \cdot \mathbf{v}}{\boldsymbol{\omega} \cdot \boldsymbol{\omega}}, \quad \mathbf{r}_\perp = \frac{\boldsymbol{\omega} \times \mathbf{v}}{\boldsymbol{\omega} \cdot \boldsymbol{\omega}} \tag{1}$$

where $\mathbf{r}_\perp$ is the axis point closest to the origin. 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 (not encountered herein) is a pure direction of a translation $(0, \mathbf{v})$ (or a force couple $(0, \mathbf{m})$).

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 $se(3)$. 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. The equivalence classes can be thought of as lines through the origin in $\mathbb{R}^{n+1}$.

It is often desirable to visualize the elements of projective space as points of an n-dimensional manifold. The *prestereographic model* [9] uses 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 lines tangent to the sphere at the origin have no such second point, and so the image space needs to be augmented by "blowing-up" the origin and replacing it with a copy of $\mathbb{RP}^{n-1}$).

3 The General Screw-Systems and Their Prestereographic Models

The screw system, $\mathscr{P}^1 = P(\mathscr{A}^2)$, of a two-dimensional subspace $\mathscr{A}^2 \subset se(3)$, is classified as *general*, when it has a pair of generators, the *principal screws* of the system, with different finite pitches, $h_1 > h_2$, and intersecting perpendicular axes:

$$\mathscr{A}^2 = \text{Span}\,(\boldsymbol{\xi}_1, \boldsymbol{\xi}_2), \quad \boldsymbol{\xi}_i = (\mathbf{e}_i, h_i\mathbf{e}_i + \mathbf{r} \times \mathbf{e}_i), \quad i = 1, 2 \tag{2}$$

where $\mathbf{e}_1 \perp \mathbf{e}_2$ are unit and $\mathbf{r}$ is the radius-vector of the intersection point.

In the plane π of $\ell(\boldsymbol{\xi}_1)$ and $\ell(\boldsymbol{\xi}_2)$, consider a circle, $S^1(\mathbf{s})$, with antipodes $\mathbf{r}$ and $\mathbf{r}+\mathbf{s}$, $\mathbf{s} = s_1\mathbf{e}_1 + s_2\mathbf{e}_2$, $|\mathbf{s}| = h_1 - h_2$. We define the image, $\mathbf{w}_\mathbf{s}(\boldsymbol{\xi})$, as the intersection of the projection of the screw axis on π with $S^1(\mathbf{s})$,

$$\begin{array}{ccccc} \mathscr{P}^1 & \longrightarrow & \mathbb{RP}^1 & \longrightarrow & S^1(\mathbf{s}) \\ \big[\boldsymbol{\xi}\big] & \longmapsto & [\boldsymbol{\omega}] & \longmapsto & \mathbf{r} + (\mathbf{s}\cdot\boldsymbol{\omega})\boldsymbol{\omega} \end{array} \tag{3}$$

We have $\mathbf{w}_i = \mathbf{w}_\mathbf{s}(\boldsymbol{\xi}_i) = \mathbf{r} + s_i\mathbf{e}_i$, $i = 1, 2$. It can be shown that the line from $\mathbf{w}_2$ to $\mathbf{w}_1$ is a *pitch axis*: the projection on it of $\mathbf{w}_\mathbf{s}(\boldsymbol{\xi})$ measures the relative pitch, $h - h_2$, of $\boldsymbol{\xi} \in \mathscr{A}^2$. Moreover, the distance of $\mathbf{w}_\mathbf{s}(\boldsymbol{\xi})$ to the pitch axis gives the elevation of the screw axis above π. Thus, the generalized *Ball circle*, $S^1(\mathbf{s})$, models geometrically the relationship between screw-axis location and pitch in the cylindroid [1, 5, 9].

A *general three-system*, $\mathscr{P}^2 = P(\mathscr{A}^3)$, is one generated by screws with different finite pitches and mutually perpendicular concurrent axes. We assume these to be the axes of the frame $Oxyz$, the unit twists on the *principal screws* being $\boldsymbol{\xi}_x = (\mathbf{i}, h_x\mathbf{i})$, $\boldsymbol{\xi}_y = (\mathbf{j}, h_y\mathbf{j})$, $\boldsymbol{\xi}_z = (\mathbf{k}, h_y\mathbf{k})$, where $h_x > h_y > h_z$. For any $\boldsymbol{\xi} \in \mathscr{A}^3$,

$$\boldsymbol{\xi} = \omega_x\boldsymbol{\xi}_x + \omega_y\boldsymbol{\xi}_y + \omega_z\boldsymbol{\xi}_z = (\boldsymbol{\omega}, \mathbf{H}\boldsymbol{\omega}). \tag{4}$$

where $\boldsymbol{\omega} = \omega_x\mathbf{i} + \omega_y\mathbf{j} + \omega_z\mathbf{k}$ and $\mathbf{H} = \text{diag}\,(h_x, h_y, h_z)$.

We select $\mathbf{s} = s_x\mathbf{i} + s_z\mathbf{k}$, $s_x = \sqrt{HH_x}$, $s_z = \sqrt{HH_z}$, $|\mathbf{s}| = H$, where $H_x = h_x - h_y$, $H_z = h_y - h_z$, and $H = H_x + H_z = h_x - h_z$. We use the prestereographic map,[1] $\mathbf{w} = \mathbf{w}_\mathbf{s}$, on the sphere, $\mathscr{S} = S^2(\mathbf{s})$, with antipodal points O and $\mathbf{s}$, Fig. 1. First we translate all screw axes of $\mathscr{P}^2$ to the origin, then we intersect them with $\mathscr{S}$:

$$\begin{array}{ccccc} \mathscr{P}^2 & \longrightarrow & \mathbb{RP}^2 & \longrightarrow & \big\{S^2(\mathbf{s}) - \{O\}\big\} \sqcup \mathbb{RP}^1 \\ \big[\boldsymbol{\xi}\big] & \longmapsto & [\boldsymbol{\omega}] & \longmapsto & \begin{cases} (\mathbf{s}\cdot\boldsymbol{\omega})\boldsymbol{\omega} & \text{if } \ \boldsymbol{\omega}\cdot\mathbf{s} \neq 0 \\ [\boldsymbol{\omega}] & \text{if } \ \boldsymbol{\omega}\cdot\mathbf{s} = 0 \end{cases} \end{array} \tag{5}$$

[1] Classic stereographic projection from O maps $\mathscr{S}$ to a standard plane model of $\mathbb{RP}^2$.

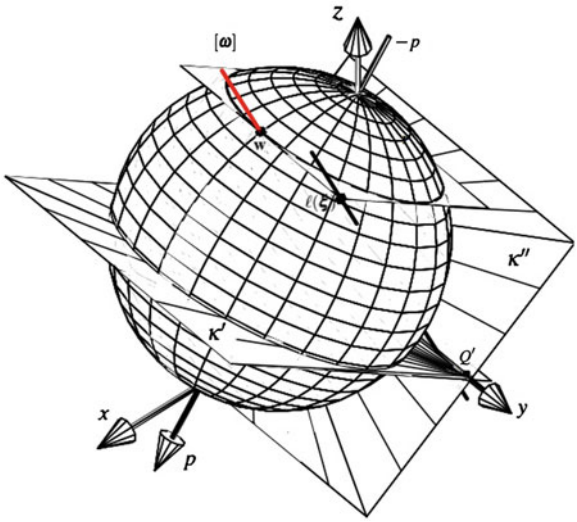


Fig. 1 The sphere, $\mathscr{S}$, with the two planes: κ', normal to the pitch axis; and κ'', tangent to $\mathscr{S}$ at O. The pencils, $\mathscr{B}'$ and $\mathscr{B}''$, of h_y-screws are visible in the planes. See [9] for more details

where $(\boldsymbol{\omega}, \mathbf{v})$ is a normalized twist in $[\boldsymbol{\xi}]$. 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''.

In the general three-system the h_y-screws form two planar pencils in two planes intersecting in $\ell(\boldsymbol{\xi}_y)$ [5]. The sphere is constructed so that it is tangent to one of these pencils, $\mathscr{B}''$ at Q'', Fig. 1. The second pencil, $\mathscr{B}'$, is in the plane $\kappa' \perp \mathbf{e}_p$ at Q'. (The coordinates of Q' and Q'' are $(0, \pm H/2, 0)$).

The line from $\mathbf{w}_z = s_z\mathbf{k}$ to $\mathbf{w}_x = s_x\mathbf{i}$, with unit vector $\mathbf{e}_p$, is the *pitch axis* of $\mathscr{S}$. The projection of the pre-stereographic image vector of any screw on the pitch axis is equal to its relative pitch, $p = h - h_y = \mathbf{w} \cdot \mathbf{e}_p$ [9].

As on the Ball circle, the position of the image point with respect to the pitch axis contains information about the location of the screw. A point on the screw axis is obtained by moving the image point, $\mathbf{w}$, along the "westward" tangent at a distance equal to that between $\mathbf{w}$ and the pitch axis [9], ($\mathbf{w}_x$ and $\mathbf{w}_y$ playing the roles of North and South poles, respectively).

4 General Two-Systems in the General Three-System

As the general three system contains no infinite-pitch screws, all 2-subsytems are either general (2–IA) or of the first special type (2–IIA, a concurrent planar pencil of same-pitch screws). It is known that the only intersecting same-pitch screws in a 3–IA system are the two concurrent pencils, $\mathscr{B}'$ and $\mathscr{B}''$. This means that there

are exactly two 2–IIA subsytems in any 3–IA system, and all other 2-subsystems are cylindroids of classes 2–IA(h_1, h_2) for some values of h_1 and h_2.

We proceed to find all the individual 2-subsystems of each general class, that is all concurrent perpendicular pairs $\boldsymbol{\xi}_1$, $\boldsymbol{\xi}_2$ within Span ($\boldsymbol{\xi}_x$, $\boldsymbol{\xi}_y$, $\boldsymbol{\xi}_z$). Thus,

$$\boldsymbol{\xi}_i = e_{ix}\boldsymbol{\xi}_x + e_{iy}\boldsymbol{\xi}_y + e_{iz}\boldsymbol{\xi}_z = (\mathbf{e}_i, \mathbf{H}\mathbf{e}_i), \quad h_i = \mathbf{e}_i \cdot \mathbf{H}\mathbf{e}_i, \qquad i = 1, 2 \tag{6}$$

where $\mathbf{H} = \text{diag}\,(h_x, h_y, h_z)$. Two perpendicular finite pitch screws are concurrent if and only if they are reciprocal, i.e., $\mathbf{e}_1 \cdot \mathbf{H}\mathbf{e}_2 = 0$. Therefore,

$$\mathbf{e}_i \cdot \mathbf{e}_j = \delta_{ij} \tag{7}$$

$$\mathbf{e}_i \cdot \mathbf{H}\mathbf{e}_j = \frac{1}{2}\delta_{ij}(h_i + h_j) \quad i = 1, 2 \tag{8}$$

where δ_{ij} is the Kronecker delta. For each given pair $h_1 > h_2$, these are six equations for the six scalar components of $\mathbf{e}_1$ and $\mathbf{e}_2$.

Taking each (7) with $i = j$, multiplying it with h_z (or h_y), and subtracting from the corresponding (8) yields (9) (or (10), respectively)

$$e_{ix}^2(h_x - h_z) + e_{iy}^2(h_y - h_z) = h_i - h_z \tag{9}$$

$$e_{ix}^2(h_x - h_y) + e_{iz}^2(h_z - h_y) = h_i - h_y \qquad i = 1, 2 \tag{10}$$

Taking now $i \neq j$ in (7) and (8):

$$e_{1x}e_{2x}(h_x - h_z) + e_{1y}e_{2y}(h_y - h_z) = 0 \tag{11}$$

$$e_{1x}e_{2x}(h_x - h_y) + e_{1z}e_{2z}(h_z - h_y) = 0 \qquad i = 1, 2 \tag{12}$$

We now express e_{iy}^2 and e_{iz}^2 with e_{ix}^2 via (9) and (10), and substitute in (11) and (12) (manipulated and squared). This results in a *linear* system for e_{1x}^2 and e_{2x}^2, yielding

$$e_{1x}^2 = \frac{(h_x - h_2)(h_y - h_1)(h_z - h_1)}{(h_1 - h_2)(h_y - h_x)(h_z - h_x)} \tag{13}$$

Similar formulas are obtained for the other 5 unknowns:

$$e_{i\sigma}^2 = -\frac{(h_\sigma - h_{i+1})(h_{\sigma+1} - h_i)(h_{\sigma+2} - h_i)}{(h_i - h_{i+1})(h_{\sigma+1} - h_\sigma)(h_{\sigma+2} - h_\sigma)} \tag{14}$$

$$i = 1, 2, 1, \ldots \quad \sigma = x, y, z, x, y, \ldots$$

where the cyclic indices i and σ scan the sets $\{1, 2\}$ and $\{x, y, z\}$, respectively (and so, for example, $i + 1 = 1$ when $i = 2$ and $\sigma + 1 = x$ when $\sigma = z$).

It can be observed that if $h_2 > h_y$ or $h_y > h_1$ there is ***no solution*** (some $e_{i\sigma}^2$ would be negative). Therefore, $h_x \geq h_1 \geq h_y \geq h_2 \geq h_z$, with $h_x > h_y > h_z$

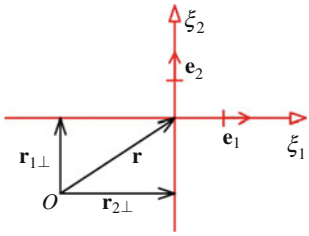


Fig. 2 The view in the central plane of the nested cylindroid

and $h_1 > h_2$. Obviously, when *both* h_1 and h_2 are among the principal pitches, $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_2$ are principal screws of the 3-system and there is ***only one*** nested 2-system. When *only one* h_i is a principal pitch, it is easy to see from (14) that $\boldsymbol{\xi}_i$ must be the corresponding principal screw and there are ***exactly two*** different solutions for $\boldsymbol{\xi}_{i+1}$. It should be noted that we identify solutions obtained from each other by reversing the direction of one or both screws, because the spanned two-system is the same.

Focusing on the remaining case $h_x > h_1 > h_y > h_2 > h_z$, we first note that, from (11) and (12), if one $e_{i\sigma} = 0$ then at least two other unknowns are zero. But this would mean that either $\boldsymbol{\xi}_1$ or $\boldsymbol{\xi}_2$ is a principal screw with a principal pitch, a contradiction. Therefore, $\mathbf{e}_1$ and $\mathbf{e}_2$ have no zero components.

Because we can reverse the directions $\mathbf{e}_1$ and $\mathbf{e}_2$, the choice of sign of, say, $e_{1x} > 0$ and $e_{2x} > 0$, is arbitrary. From (11) and (12) we see that there are then two independent choices for the signs of e_{1y} and e_{2x} and, for each of those, two possibilities for the signs of e_{1z} and e_{2z}. This gives ***four different solutions*** for the nested 2-system.

We now proceed to find the intersection point, $\mathbf{r}$, of $\ell(\boldsymbol{\xi}_1)$ and $\ell(\boldsymbol{\xi}_2)$. The point on $\ell(\boldsymbol{\xi}_i)$ closest to O is $\mathbf{r}_{i\perp} = \mathbf{e}_i \times \mathbf{He}_i$. It is known that O, $\ell(\boldsymbol{\xi}_1)$, and $\ell(\boldsymbol{\xi}_2)$ are coplanar [5]. (Indeed, $\mathbf{e}_1 \times \mathbf{e}_2 \cdot \mathbf{e}_i \times \mathbf{He}_i = 0$ can be shown easily implying that all $\mathbf{r}_{i\perp}$ and $\mathbf{e}_i$ are coplanar and $\mathbf{r}_{i\perp} \parallel \mathbf{e}_{i+1}$.) Therefore, $\mathbf{r} = \mathbf{r}_{1\perp} + \mathbf{r}_{2\perp}$. It can be shown that if the signs of $e_{i\sigma}$ in the four nested systems are chosen so that $e_{1x}e_{2x} < 0$ and e_{1x} is positive twice, the directions of $\mathbf{r}_{i\perp}$ and $\mathbf{e}_{i+1}$ always coincide, Fig. 2, and we have

$$\mathbf{r}_{i\perp} = \sqrt{\frac{(h_\alpha - h_i)(h_\beta - h_i)(h_\gamma - h_i)}{h_i - h_{i+1}}}\mathbf{e}_{i+1} \qquad i = 1, 2, 1, \ldots \tag{15}$$

5 The Nested Two-Systems on the Prestereographic Sphere

We analyze the problem geometrically using the sphere, $\mathscr{S}$. The screws of any two-system are parallel to a plane and so their images on $\mathscr{S}$ are coplanar. There is a one-to-one correspondence between the nested subsystems and the planes through the origin. Each 2-subsystem, $\mathscr{P}^1(\pi)$, of plane π, maps to a (distinct) circle, $\sigma(\pi) = \mathscr{S} \cap \pi$, through O (except the tangent pencil $\mathscr{B}''$). The principal screws of $\mathscr{P}^1(\pi)$ map to

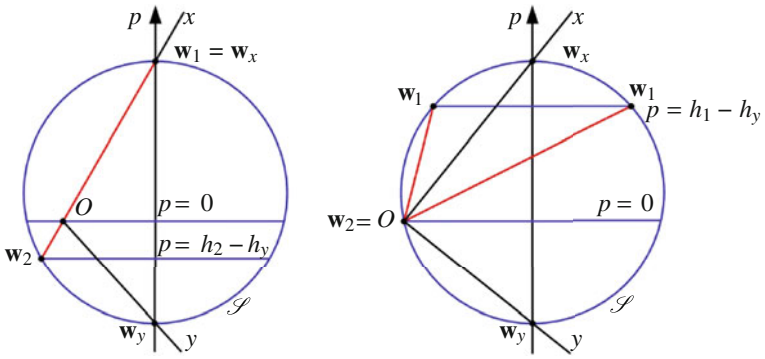


Fig. 3 Side views (as segments $\mathbf{w}_1\mathbf{w}_2$) of the image circles on $\mathscr{S}$ when $h_x = h_1$ (*left*) or $h_y = h_2$

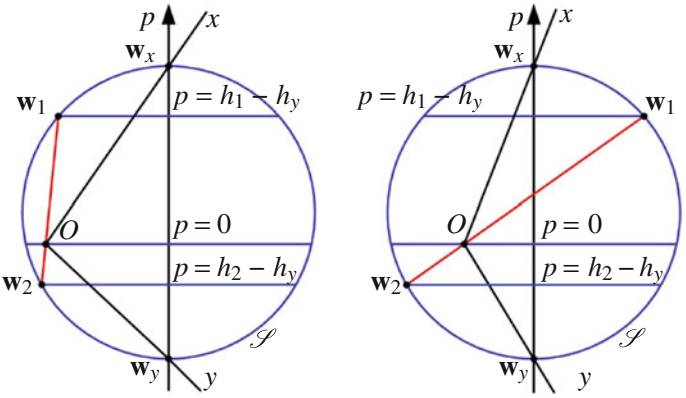


Fig. 4 Side views (as segments $\mathbf{w}_1\mathbf{w}_2$) of the image circles on $\mathscr{S}$ when $h_x > h_1 > h_y > h_2 > h_z$

the points, $\mathbf{w}_1$ and $\mathbf{w}_2$, with highest and lowest pitch (i.e., "latitude" on $\mathscr{S}$) on $\sigma(\pi)$. Such points are antipodal on the circle (not necessarily on the sphere) and the line connecting them will intersect the pitch axis (i.e., they are on the same "meridian circle"). (Figs. 3 and 4 are projections on these meridian planes).

Since every circular image of a nested 2-subsystem passes through O, every subsystem includes a screw of pitch h_y. Hence ***no solution*** exists unless $h_1 \geq h_y \geq h_2$. The case $h_1 = h_y = h_2$ gives two possible special-subsystem images: the 0-pitch parallel circle and the tangent pencil $\mathscr{B}''$. When $h_x = h_1 > h_2 = h_y$ (or $h_y = h_1 > h_2 = h_z$) we have a ***unique solution***, $\mathscr{S} \cap Oxy$ (or $\mathscr{S} \cap Oyz$). When only one h_i is principal in $\mathscr{P}^2$, there are ***two possible sections***, Fig. 3. When the principal pitch is h_x or h_y the two circles are congruent, while when $h_i = h_y$ they have different diameters.

Finally, when $h_x > h_1 > h_y > h_2 > h_z$, there are ***exactly four circles*** through O on $\mathscr{S}$ tangent to both the h_1- and h_2-pitch parallels. Indeed, there are two different possible diameters, corresponding to the cases when the plane π intersects the pitch

axis inside and outside the sphere. When looking at the meridian plane containing $\mathbf{w}_1$ and $\mathbf{w}_2$, the two segments appear as the side and the diagonal of an equilateral trapezoid. For each diameter there are two distinct sections through O: on the side view in Fig. 4, point O may be either on the front or on the back of the sphere.[2]

It is easy to construct geometrically the 2-system origin, $\mathbf{r}$, in the plane π. First, we find two points on the axes by translating $\mathbf{w}_1$ and $\mathbf{w}_2$ along the westward tangents at distances equal to the radii of the h_1- and h_2-parallels. From these points, $\ell(\boldsymbol{\xi}_1)$ and $\ell(\boldsymbol{\xi}_2)$ can be drawn parallel to $\mathbf{w}_1$ and $\mathbf{w}_2$, respectively, to their intersection, $\mathbf{r}$.

6 The Special A-Type Three-Systems

When exactly two of the three principal pitches are equal, we no longer have a general but a (so-named in [5]) first special three-system. If $h_y = h_z$, on $\mathscr{S}$ we have $\mathbf{w}_x = O$ and the pitch axis coincides with Ox. As the subsystem image circles pass through O, every 2-subsystem will have a principal pitch $h_2 = h_y = h_z$.

There is a unique special 2-subsystem of concurrent (at O) screws with pitch $h_y = h_z$ in the Oyz plane. When $h_1 = h_x$ the section can be any meridian circle $\mathscr{S}$: there are ***infinitely many*** 2-subsystems of class 2–IA(h_x, h_y), with the same $\boldsymbol{\xi}_1 = \boldsymbol{\xi}_x$ and $\boldsymbol{\xi}_2$ any h_y-pitch screw through O in Oyz. If $h_x > h_1 > h_2$, any point on the h_1-pitch parallel can be chosen as $\mathbf{w}_1$. There are ***infinitely many*** 2-subsystems of class 2–IA(h_1, h_y), but note that they are "twice as many" as when $h_1 = h_x$. The origin of all subsystems is in Oyz, at a distance equal to the radius of the h_1-pitch parallel.

The case when $h_x = h_y > h_z$ is analogous to $h_x > h_y = h_z$. Finally, when $h_x = h_y = h_z$, we have a system consisting of all h_x-pitch screws with axes through O. Trivially, the 2-subsystems are all of the same class: concurrent planar pencils at O.

7 Conclusions

The chapter analyzes the nesting of two-subsystems within screw systems of rank three with no infinite-pitch screws. In the important case of the general three-system, it is shown that at most four screw cylindroids of every class are nested, and all solutions are computed explicitly. The subcases with zero, one, two, and exactly four subsystems of a given class are identified precisely and the possibilities are illustrated geometrically with the help of a prestereographic model.

[2] It should be noted that the varying multiplicity of solutions for the 2-system can also be visualized in elegant ways using plane models (like the ones in [1, 4]) of the 3-system.

References

1. Ball, R.: A Treatise on the Theory of Screws. Cambridge University Press, Cambridge (1900)
2. Carricato, M., Zlatanov, D.: Persistent screw systems. Mech. Mach. Theory **73**, 296–313 (2014)
3. Donelan, P., Gibson, C.: On the hierarchy of screw systems. Acta Appl. Mathe. **32**(3), 267–296 (1993)
4. Gibson, C., Hunt, K.: Geometry of screw systems. Mech. Mach. Theory **25**(1), 1–27 (1990)
5. Hunt, K.: Kinematic Geometry of Mechanisms. Clarendon Press, Oxford (1978)
6. Kong, X., Gosselin, C.: Type Synthesis of Parallel Mechanisms. Springer, Berlin (2007)
7. Phillips, J.: Freedom in Machinery. Cambridge University Press, Cambridge (1984–1990)
8. Selig, J.: Geometric Fundamentals of Robotics. Springer, Berlin (2005)
9. Zlatanov, D.: The representation of the general three-system of screws by a sphere. Mech. Mach. Theory **49**, 315–331 (2012)
10. Zlatanov, D., Carricato, M.: Characterization of the subsystems in the special three-systems of screws. In: Glazunov, V., Ceccarelli, M. (eds.) ROMANSY 20—Robot Design, Dynamics and Control. Springer (2014)

Geometrical Patterns for Measurement Pose Selection in Calibration of Serial Manipulators

Alexandr Klimchik, David Daney, Stephane Caro and Anatol Pashkevich

Abstract The chapter is devoted to the accuracy improvement in geometric calibration of serial manipulators. Particular attention is paid to the optimal selection of measurement poses, which reduce measurement noise impact on the parameters identification precision. In contrast to previous works, the proposed approach yields simple geometrical patterns that allow user to take into account the joint and workspace constraints and to find measurement configurations without tedious computations. The advantages and practical significance of the proposed approach are illustrated by an example that deals with 6-dof serial manipulator.

Keywords Calibration · Design of experiments · Geometrical patterns · Serial robot

1 Introduction

Since most of industrial serial robots operate without position feedback,

the precision of a geometric model used in the control algorithm should be high enough. In order to meet this requirement in practice, the calibration technique is

A. Klimchik (✉) · A. Pashkevich
IRCCyN, Ecole des Mines de Nantes, Nantes, France
e-mail: alexandr.klimchik@emn.fr

A. Pashkevich
e-mail: anatol.pashkevich@emn.fr

D. Daney
Institut National de Recherche en Informatique et en Automatique (INRIA),
Le Chesnay, France
e-mail: david.daney@inria.fr

S. Caro
IRCCyN, Centre National de la Recherche Scientifique, Paris, France
e-mail: stephane.caro@irccyn.ec-nantes.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_28,

usually applied. For this reason, the problem of robot calibration has always been in the focus of the robotic community. There exist open-loop and closed-loop techniques that are suitable for serial and parallel manipulators [3, 4]. However, to apply them in a real industrial environment, it is necessary to reduce the impact of the measurement noise that corrupts the input data and is evaluated via the terms of the covariance matrix [6].

In order to reduce the impact of measurement noise, two approaches can be generally applied: (i) increasing the number of experiments and (ii) applying design of experiments theory for selection of measurement configurations. The first approach is quite a simple, it does not require any computation, but increases a lot the measurement time and related cost. In contrast, the second approach allows us to increase identification accuracy without increasing the number of measurements [2]. However this approach requires additional knowledge and some computational efforts. Nevertheless, because of its advantages the second approach is more attractive and is in the focus of this chapter.

Although the design of experiment theory has been used for more than a century in different areas, the obtained results cannot be applied directly in robotics since they are mostly suitable for linear regression models. Therefore, to obtain similar results for robot calibration that rely on highly non-linear models, the existing techniques should be essentially revisited. In addition, to make the technique attractive for practicing engineers, it is reasonable to obtain some geometrical patterns allowing the user to generate optimal measurement configurations without any computational efforts.

2 Problem of Measurement Pose Selection in Robot Calibration

To show the importance of the problem of interest, let us present first a simple motivation example. For the purpose of simplicity, let us limit our study by a 2-dof planar manipulator with two actuated revolute joints and link lengths $l_1 = 1.0\,\text{m}$, $l_2 = 1.0\,\text{m}$. This manipulator has four geometrical parameters to be identified: (i) two link length deviations and (ii) two joint encoder offsets.

For comparison purposes, let us examine two sets of measurement configurations assuming that the measurement noise is Gaussian with zero mean and the standard deviation $\sigma = 0.1\,\text{mm}$. Simulation results are summarized in Fig. 1, where the root-mean-square errors ρ of the end-effector position after calibration throughout the robot workspace are presented. As follows from Fig. 1, in the worst manipulator configuration, the Set #1 provides a positioning error equal to 2.29 mm while the Set #2 reduces the worst positioning error down to 0.14 mm, i.e. by a factor of 16. Hence, this simple example clearly shows that the selection of measurement configurations is a very important issue in robot calibration. In fact, poorly chosen measurement configurations may have a negative effect and even reduce the robot accuracy after calibration. This motivates careful planning of the calibration experiments in order to increase calibration efficiency, which is in the focus of the chapter.

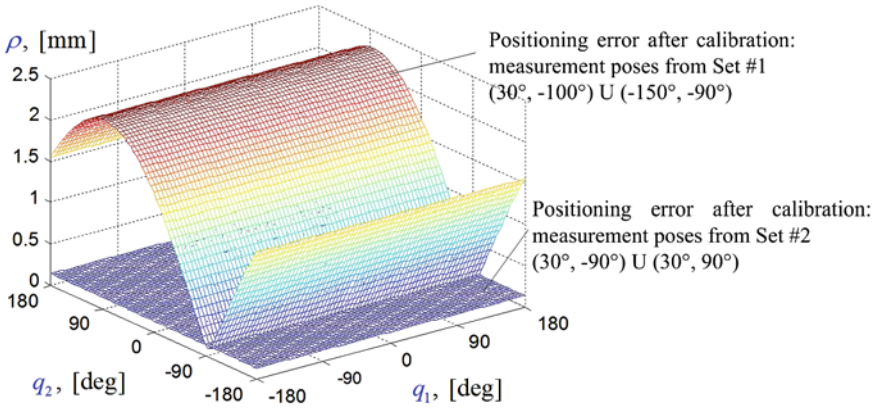


Fig. 1 Manipulator accuracy after calibration for two different sets of measurement poses

In the frame of this chapter, a general planar manipulator with revolute joints is considered, whose the end-effector position is computed using the following expressions

$$x = \sum_{i=1}^{n} \left(l_i^0 + \Delta l_i \right) \cdot \cos(\theta_i^0 + \Delta\theta_i); \quad y = \sum_{i=1}^{n} \left(l_i^0 + \Delta l_i \right) \cdot \sin(\theta_i^0 + \Delta\theta_i) \quad (1)$$

where l_i^0 are the nominal link lengths, Δl_i are their deviations, q_j^0 are nominal joint coordinates, the variables θ_i^0 are defined as $\sum_{j=1}^{i} q_j^0$, $\Delta\theta_i$ are the joint offsets and n is the number of links. Collecting the unknown parameters (Δl_i and $\Delta\theta_i$) into the vector $\Delta\pi$ and the measurements (x^k, y^k) into the vector $\Delta\mathbf{P}^k$, expression (1) can be rewritten as $\Delta\mathbf{P} = \mathbf{J}\,\Delta\pi$, where $\mathbf{J}$ is a Jacobian matrix. Then, one can get unknown parameters using the least-square technique that leads to

$$\Delta\pi = \left(\sum_{k=1}^{m} \mathbf{J}^{k\,T} \mathbf{J}^k \right)^{-1} \cdot \sum_{k=1}^{m} \mathbf{J}^{k\,T} \Delta\mathbf{P}^k \quad (2)$$

where the superscript 'k' indicates the experiment number, and m is the number of measurements.

Taking into account that each measurement is corrupted by the unbiased random Gaussian noise with standard deviation (std) σ, the identification accuracy of the parameters $\Delta\pi$ can be evaluated via the covariance matrix, which is computed as follows:

$$\mathrm{cov}(\Delta\pi) = \sigma^2 \left(\sum_{k=1}^{m} \mathbf{J}^{k\,T} \mathbf{J}^k \right)^{-1} \quad (3)$$

The latter allows us to evaluate the impact of measurement noise on the parameters identification accuracy and can be used to estimate the quality of measurement configurations. Based on the covariance matrix, it is possible to choose the measurement configurations that yield parameters less sensitive to measurement noise. In engineering practice, this procedure is referred to as the design of calibration experiments. In order to compare different plans of experiments (i.e. their efficiency), different performance measures have been proposed that deal with the norm, trace, etc. of the covariance matrix or information matrix (its inverse) [1]. In [5] the authors proposed an optimality condition of the calibration plan where the main idea is to ensure that the information matrix is diagonal. This condition corresponds to the D-optimal plan of experiments and is satisfied when

$$\sum_{k=1}^{m} \cos\left(\sum_{s=j}^{i} q_s^k\right) = 0, \quad \sum_{k=1}^{m} \sin\left(\sum_{s=j}^{i} q_s^k\right) = 0, \qquad \forall\, i,\, j = \overline{2,n},\ i \geq j \tag{4}$$

i.e. the sums of sines and cosines of links orientation q_s^k with respect to any joint s should be equal to zero.

It has been proved that the above presented equations define the desired set of the optimal measurement configurations, but in general case solution of this system requires essential efforts. In fact, for $n = 2$, $m \geq 2$ expressions (4) provide two simple constraints only $\sum_{k=1}^{m} \cos q_2^k = 0$, $\sum_{k=1}^{m} \sin q_2^k = 0$, which are imposed on the $2m$ design variables $\left\{q_1^k,\ q_2^k \mid k = \overline{1,m}\right\}$. So, relevant configurations can be easily found geometrically [5]. However, even in the case of $n = 3$, $m \geq 3$ there are 6 rather complicated trigonometric constraints imposed on $3m$ design variables $\left\{q_1^k,\ q_2^k,\ q_3^k \mid k = \overline{1,m}\right\}$. Therefore, the solution of the corresponding underconstrained system of algebraic equations becomes non-trivial. It is apparent that the complexity of the problem essentially increases with n (the number of manipulator links). Additional difficulties arise when the joint limits should be taken into account (i.e. $\left\{q_i^{\min} \leq q_i^k \leq q_i^{\max} \mid\ i = \overline{1,n};\ \ k = \overline{1,m}\right\}$). Hence, it looks reasonable to develop relevantly simple and efficient technique allowing practical engineers to select desired measurement configurations in accordance with Eq. (4) without any tedious computations. This problem is in the focus of this chapter.

3 Geometrical Patterns for Measurement Pose Selection

Before defining patterns, let us obtain some important properties of the optimality conditions (4) that allow us to reduce the problem complexity.

Property 1: *Superposition of optimal plans gives also an optimal plan for this.*

Proof of this property is obvious and is based on the additivity of the operations included in (4). Using this property it is possible to generate optimal plans with a large number of measurement configurations using simple sub-sets. In a trivial case,

it is possible to repeat experiments using the same set of optimal configurations. The latter is very important in practice, since it is much less time consuming to repeat experiments without changing manipulator configurations.

Property 2: *The angles $q_2, \ldots, q_n$ can be rearranged in the optimal plans in an arbitrary manner without losing the optimality conditions* (4).

Proof Let us assume that for an n-link manipulator optimal plan consists of m measurement configurations. Let us now change in the optimal plan column a and b $(a, b > 1)$ and analyze the sets of angles for which sums of sines and cosines should be satisfied in the optimal plan $\sum_{s=j}^{i} q_s^k$. In this case conditions (4) will be satisfied for $i, j = \overline{2, a}$ and $i, j = \overline{b, n}$ since all arguments of sines and cosines remain the same. For case $a \le j \le i \le b$ conditions also do not change. However, in the cases $a \le j \le b \le i$ and $j \le a \le i \le b$ optimality conditions $\sum_{s=j}^{i} q_s^k$ will be replaced $\sum_{s=j}^{i} q'^k_s = \sum_{s=j}^{i} q_s^k + q_a^k - q_b^k$ by and $\sum_{s=j}^{i} q'^k_s = \sum_{s=j}^{i} q_s^k + q_b^k - q_a^k$ respectively, where prime indicates the angle corresponding to the rearranged new set of angles. Assuming that plan $\sum_{s=j}^{i} q_s^k$ is optimal, plans $\sum_{s=j}^{i} q'^k_s - q_a^k + q_b^k$ and $\sum_{s=j}^{i} q'^k_s + q_a^k - q_b^k$ will be also optimal. Further, taking into account that the superposition of optimal plans gives optimal plan and applying it to the last two plans gives plan $2\sum_{s=j}^{i} q'^k_s$ that should be also optimal. This plan can be split into two sub-plans $\sum_{s=j}^{i} q'^k_s$ for which the sums of sines and cosines should be equal. Taking into account that double sum is equal to zero, the plan $\sum_{s=j}^{i} q'^k_s$ will be also optimal.

This property can be useful when the joint limits are narrower than the minimum range required for the optimal plan. In this case, it is reasonable to change joint coordinates with the ones where joint limits are not so critical (more than 240°) and where the required range of joint variations is lower.

Property 3: *Optimal plan for n-link manipulator can be obtained using two lower-order optimal plans for n_1- and n_2-link manipulators, where $n_1 + n_2 = n + 1$.*

This property gives us an elegant technique to generate optimal plans of calibration experiments without straightforward solution of system (4). In this case the number of measurement configurations is defined by $n_1 \times n_2$. In fact, by a sequential splitting of the original n-link kinematic chain, it is possible to reduce the problem complexity and to replace the problem of generating optimal plan for n-link manipulator by several sub-problems for 2-, 3- and 4-link manipulators. The latter motivates us to develop typical geometrical patterns that can be used directly to generate optimal plans for complex manipulators.

Now let us introduce some geometrical patterns for typical serial manipulators that can be used to generate optimal plans. In the frame of these patterns, all variables α_i, β_i and γ, δ are treated as arbitrary angles.

For $n = 2, m = 2$, the complete set of solutions can be expressed as:

$$\begin{array}{ll} q_1^1 = \alpha_1; & q_2^1 = \beta \\ q_1^2 = \alpha_2; & q_2^2 = \beta + \pi \end{array} \tag{5}$$

For $n = 2$, $m = 3$, the desired set of solutions is

$$\begin{array}{ll} q_1^1 = \alpha_1; & q_2^1 = \beta \\ q_1^2 = \alpha_2; & q_2^2 = \beta + 2\pi/3 \\ q_1^3 = \alpha_3; & q_2^3 = \beta - 2\pi/3 \end{array} \tag{6}$$

For $n = 3$, $m = 3$, the geometrical pattern can be presented as

$$\begin{array}{lll} q_1^1 = \alpha_1; & q_2^1 = \beta; & q_3^1 = \gamma \\ q_1^2 = \alpha_2; & q_2^2 = \beta + 2\pi/3; & q_3^2 = \gamma + 2\pi/3 \\ q_1^3 = \alpha_3; & q_2^3 = \beta - 2\pi/3; & q_3^3 = \gamma - 2\pi/3 \end{array} \tag{7}$$

For $n = 3$, $m = 4$, an optimal solution can be expressed as

$$\begin{array}{lll} q_1^1 = \alpha_1; & q_2^1 = \beta_1; & q_3^1 = \gamma \\ q_1^2 = \alpha_2; & q_2^2 = \beta_1 + \pi; & q_3^2 = \gamma \\ q_1^3 = \alpha_3; & q_2^3 = \beta_2; & q_3^3 = \gamma + \pi \\ q_1^4 = \alpha_4; & q_2^4 = \beta_2 + \pi; & q_3^4 = \gamma + \pi \end{array} \tag{8}$$

For $n = 4$, $m = 4$, the desired geometrical pattern can be written as

$$\begin{array}{llll} q_1^1 = \alpha_1; & q_2^1 = \beta_1; & q_3^1 = \gamma; & q_4^1 = \delta \\ q_1^2 = \alpha_2; & q_2^2 = \beta_1 + \pi; & q_3^2 = \gamma; & q_4^3 = \delta + \pi \\ q_1^3 = \alpha_3; & q_2^3 = \beta_2; & q_3^3 = \gamma + \pi_2; & q_4^3 = \delta + \beta - \beta_2 \\ q_1^4 = \alpha_4; & q_2^4 = \beta_2 + \pi; & q_3^4 = \gamma + \pi_2; & q_4^4 = \delta + \beta - \beta_2 + \pi \end{array} \tag{9}$$

When the optimal plan for the entire n-link manipulator is defined, it is required to fix all arbitrary variables. In case there is no joint limit, they can be set to any value. In contrast, in case the joint angles are limited, these variables should be properly selected using the following rules:

Rule 1. If the joint limits for the angle are narrower than $2\pi/3$ while plan of experiments requires higher width for joint variations, the joint coordinates of this joint should be changed with the one for which the joint coordinates can be set within the interval of the length π.

Rule 2. If the joint limits for the angle are narrower than π, some optimality conditions in (4) cannot be satisfied and the plan of experiments should be modified. In this case, the best results can be achieved when the joint coordinates are set to the joint limits.

Rule 3. For the joint with a range of variation equal to π, an arbitrary parameter can be fixed in the interval $[q_{\min};\ q_{\max} - \pi]$, where $q_{\min}$ and $q_{\max}$ are the corresponding joint limits.

Rule 4. For the joint with a range of variation equal to $2\pi/3$, an arbitrary parameter can be fixed in the interval $[q_{\min} + \pi/3;\ q_{\max} - \pi/3]$.

Hence, the above defined properties and rules allow us to generate optimal plan of experiments for the complex manipulator using simple patterns without any tedious

computation. In the next section the obtained results will be applied for the generation of a plan of experiments for a 6-dof manipulator.

4 Case Study: Optimal Measurement Poses for a 6-Dof Robot

To show the efficiency of the developed optimal measurement pose selection technique, let us consider a 6-dof serial manipulator with six revolute joints and six links ($l_1 = 1.4\,\text{m}$, $l_2 = 1.1\,\text{m}$, $l_3 = 1.0\,\text{m}$, $l_4 = 0.8\,\text{m}$, $l_5 = 0.6\,\text{m}$, $l_6 = 0.4\,\text{m}$), whose joint limits are $-140^\circ \le q1 \le 0^\circ$, $-120^\circ \le q2 \le 90^\circ$, $-120^\circ \le q3 \le 120^\circ$, $-180^\circ \le q4 \le 180^\circ$, $0^\circ \le q5 \le 210^\circ$, $0^\circ \le q6 \le 360^\circ$. In order to simplify the determination of measurement configurations, the manipulator can be split into two sub-chains: (i) 3-link chain corresponding to links #1 − #3, and (ii) 4-link chain composed of a virtual link/joint and links #4 − #6 of entire manipulator.

According to the geometrical patterns presented above, the optimal plans for the first and second sub-chains can be generated using Eqs. (7) and (9), respectively. For the entire manipulator, the optimal plan can be obtained using Property 3 (it consists of 12 measurement configurations). However, such a plan of experiments requires a large range of variations for q_2 (equal to $2\pi/3$), which cannot be included in the joint limits. This difficulty can be overcome using Property 2 allowing us permutation of q_2 and q_4. This provides us with the following plan of calibration experiments, which satisfies Eq. (4) and, consequently, insures the covariance matrix diagonality:

$$
\begin{array}{cccccc}
\alpha_1; & \delta_1; & \gamma & \beta; & \chi; & \varphi \\
\alpha_1; & \delta_1 + \pi; & \gamma & \beta; & \chi; & \varphi + \pi \\
\alpha_1; & \delta_2; & \gamma & \beta; & \chi + \pi_2; & \varphi + \delta_1 - \delta_2 \\
\alpha_1; & \delta_2 + \pi; & \gamma & \beta; & \chi + \pi_2; & \varphi + \delta_1 - \delta_2 + \pi \\
\alpha_2; & \delta_1; & \gamma + 2\pi/3 & \beta + 2\pi/3; & \chi; & \varphi \\
\alpha_2; & \delta_1 + \pi; & \gamma + 2\pi/3 & \beta + 2\pi/3; & \chi; & \varphi + \pi \\
\alpha_2; & \delta_2; & \gamma + 2\pi/3 & \beta + 2\pi/3; & \chi + \pi_2; & \varphi + \delta_1 - \delta_2 \\
\alpha_2; & \delta_2 + \pi; & \gamma + 2\pi/3 & \beta + 2\pi/3; & \chi + \pi_2; & \varphi + \delta_1 - \delta_2 + \pi \\
\alpha_3; & \delta_1; & \gamma - 2\pi/3 & \beta - 2\pi/3; & \chi; & \varphi \\
\alpha_3; & \delta_1 + \pi; & \gamma - 2\pi/3 & \beta - 2\pi/3; & \chi; & \varphi + \pi \\
\alpha_3; & \delta_2; & \gamma - 2\pi/3 & \beta - 2\pi/3; & \chi + \pi_2; & \varphi + \delta_1 - \delta_2 \\
\alpha_3; & \delta_2 + \pi; & \gamma - 2\pi/3 & \beta - 2\pi/3; & \chi + \pi_2; & \varphi + \delta_1 - \delta_2 + \pi
\end{array}
\tag{10}
$$

Taking into account given joint limits, the arbitrary angles in the above expression have been set as

$$
\begin{array}{ccccc}
\alpha_1 = 0; & \alpha_2 = -\pi/3; & \alpha_3 = -2\pi/3; & \beta = 0; & \\
\delta_1 = -2\pi/3; & \delta_2 = -\pi/2; & \gamma = 0; & \chi = \pi/6; & \varphi = \pi/2
\end{array}
\tag{11}
$$

Using this plan of experiments, the desired geometrical parameters can be identified with a precision of 0.029 mm for the link lengths and 4.13 mdeg for the joint

Table 1 Identification accuracy for different plans of calibration experiments

Joint	Proposed plan		Regular plan		Random plan	
	δq_i (mdeg)	δL_i (mm)	δq_i (mdeg)	δL_i (mm)	δq_i (mdeg)	δL_i (mm)
1	1.18	0.029	1.70	0.041	1.90	0.046
2	1.50	0.029	2.21	0.042	2.58	0.050
3	1.65	0.029	2.32	0.041	2.46	0.043
4	2.07	0.029	3.04	0.042	3.32	0.046
5	2.76	0.029	4.04	0.042	4.39	0.046
6	4.13	0.029	5.73	0.040	5.55	0.039
$\rho_{\max}$	0.61		0.88		0.96	

offsets (assuming that the measurement system precision is equal to 0.1 mm). In more details, relevant results are presented in Table 1. For comparison purposes, the identification accuracy has been also evaluated for regular and random plans of experiments that contain the same number of measurement configurations. It should be noted that for a 6-dof manipulator it is not possible to obtain a regular plan with 12 measurement configurations. For this reason the plan has been generated using five measurement configurations within the joint limits from which 12 configurations have been selected randomly. In order to reduce the factor of particular set of measurement configurations, simulations for regular and random plans have been repeated 1,000 times and the results have been averaged.

Hence, simulation results confirm advantages of the proposed approach. In this study, the identification accuracy for the optimal plan is better by 34–72 % comparing with the random plan and by 38–48 % compared with the regular plan of experiments. In addition, maximum positioning errors have been reduced by 44 % and 57 %, respectively. It should be stressed that the proposed approach does not require any computation for optimal measurement pose selection and is able to improve the identification accuracy using a small number of measurement configurations.

5 Conclusions

The chapter presented a new approach for the design of calibration experiments for robotic manipulators that essentially simplifies the optimal pose selection procedure. The main theoretical results are expressed as a set of several properties and rules, which allow user to obtain optimal measurement configurations without any computation, simply using superpositions and permutations of the proposed geometrical patterns describing optimal measurement configurations for 2-, 3- and 4-link manipulators. The efficiency of the developed approach has been confirmed by an illustrative example that deals with the calibration of a 6-dof manipulator.

Acknowledgments The work presented in this chapter was partially funded by the project ANR (Project ANR-2010-SEGI-003-02-COROUSSO), France and FEDER ROBOTEX, France.

References

1. Atkinson, A.C., Donev, A.N., Tobias, R.D.: Optimum experimental designs, with SAS. Oxford University Press, Oxford (2007)
2. Borm, J.H., Meng, C.H.: Determination of optimal measurement configurations for robot calibration based on observability measure. Int. J. Robot. Res. **10**(1), 51–63 (1991)
3. Elatta, A., Gen, L.P., Zhi, F.L., Daoyuan, Y., Fei, L.: An overview of robot calibration. Inf. Technol. J. **3**(1), 74–78 (2004)
4. Hollerbach, J., Khalil, W., Gautier, M.: Model identification. In: Siciliano, B., Khatib, O. (eds.) Springer Handbook of Robotics, pp. 321–344. Springer, Berlin (2008)
5. Klimchik, A., Wu, Y., Caro, S., Pashkevich, A.: Design of experiments for calibration of planar anthropomorphic manipulators. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), pp. 576–581. IEEE (2011)
6. Mooring, B.W., Roth, Z.S., Driels, M.R.: Fundamentals of manipulator calibration. Wiley, New York (1991)

Stiffness Analysis of a Fully Compliant Spherical Chain with Two Degrees of Freedom

Farid Parvari Rad, Giovanni Berselli, Rocco Vertechy and Vincenzo Parenti-Castelli

Abstract This chapter introduces and investigates a fully compliant spherical chain that is obtained by the in-series connection of two identical primitive spherical flexures with coincident center of spherical motion. The compliance matrix of the proposed chain is obtained via an analytical procedure and validated via finite element analysis. Comparison with an equivalent fully compliant chain employing straight beam hinges is also provided to highlight the added benefits when using primitive spherical flexures.

Keywords Spherical flexures · Compliance matrix · Finite element analysis · Parasitic motions

1 Introduction

Spherical mechanisms are an important class of spatial kinematic chains that find vast applications in pointing/orientation systems and mechanical transmissions [3]. These mechanisms are characterized by having all points of their links moving on

F. Parvari Rad (✉) · V. Parenti-Castelli
Department of Mechanical Engineering, University of Bologna, Bologna, Italy
e-mail: farid.parvarirad2@unibo.it

V. Parenti-Castelli
e-mail: vincenzo.parenti@unibo.it

G. Berselli
Department of Mechanical Engineering, University of Modena and Reggio Emilia, Modena and Reggio Emilia, Italy
e-mail: giovanni.berselli@unimore.it

R. Vertechy
Percro Lab., Scuola Superiore Sant'Anna, Pisa, Italy
e-mail: r.vertechy@sssup.it

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_29,

concentric spherical surfaces; the center of these spheres being denoted as center of spherical motion. In its simplest form, a spherical chain features the in-series connection of two or more revolute pairs with axes intersecting in the said center of spherical motion.

As with any planar or spatial linkage, spherical mechanisms based on conventional kinematic pairs suffer from backlash, friction, slip-stick and wear in the joints. This poses a major challenge to the development of spherical mechanisms with high repeatability and resolution, and that are, at the same time, easy to manufacture and assemble, maintenance free and cost effective.

In order to overcome this limitation, compliant joints can be used [6, 10]. Compliant joints are solid-state connections in which the relative motion of joined links is gained via material deformations rather than by the sliding or rolling of mating surfaces. Compliant joints can be categorized in three classes [15]: small-length flexure pivots (including notch type hinges), long flexible segments (including straight beams, curved beams, leaf springs and tape springs) and complex flexures that are obtained by combining primitive compliant joints belonging to the former two classes. Although compliant joints provide monolithic connections with no backlash and hysteresis, they usually bring the following disadvantages: limited range of motion, parasitic motions and stress concentrations. For this reason, to be effective, compliant joints need to be conceived and optimized for the specific application at hand. To date, only a limited number of works have investigated compliant joints specifically designed for spherical motion as well as fully compliant spherical mechanisms. Smith [16] proposed compliant universal joints fabricated from circular leaf springs, which also provided axial translation for self-alignment applications. However, the proposed joints are affected by significant stress concentrations that limit their ranges of motion. Lobontiu et al. [11, 12] investigated the two- and three-axis flexure hinges. The former consists of two collocated notches that are cut perpendicular to each other; the latter consists of an axial-symmetric notch. In both cases, the resulting hinge features a small cross-sectional area and is prone to unintentional rotations or buckling even when loaded with small forces. Moon et al. [17] developed a compliant revolute hinge based on torsion beams of cross or segmented-cross type, and employed two of them, connected in series with orthogonal axes, to conceive a fully compliant universal joint. Later on, the ensemble of two universal joints of this kind has been proposed by Machekposhti et al. [4] to obtain a compliant constant velocity Double-Hooke's universal joint. Different authors [5, 13] employed two in-series connected flexure notch hinges with orthogonal axes to conceive a fully compliant universal joint. Jacobsen et al. [7] employed three in-series connected lamina emergent torsional joints with axes intersecting in a single point to make spherical chains with three degrees of freedom (for compliant joints or mechanisms, the number of degrees of freedom is intended as the number of independent prevalent directions of motion). These spherical chains were then used to build a 3-RRR spherical parallel mechanism (R being a revelote joint). Callegari et al. [1] addressed the analysis and design of a 3-CRU spherical parallel mechanism with flexure hinges (C and U being cylindrical and universal joints respectively).

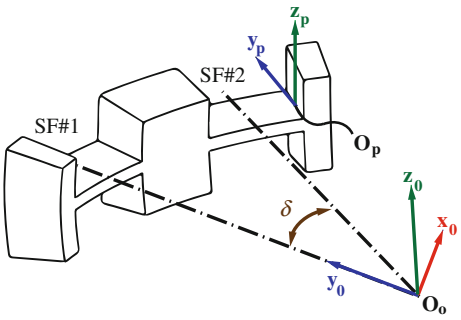


Fig. 1 SF-based compliant chain

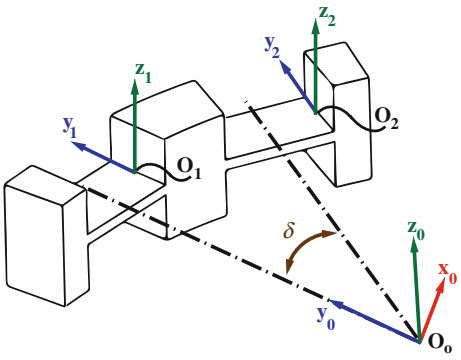


Fig. 2 SBF-based compliant chain

Li and Chen [9] employed two circularly-curved beam flexures with rectangular cross-section to devise a spherical Young parallel mechanism. Apart the works by Lobontiu et al. and that by Li and Chen, all the aforementioned studies are based on the use and proper combination of primitive flexures that are specifically conceived for prevalent planar motions only. Recently, a primitive Spherical Flexure (SF) specifically designed for prevalent spherical motion has been proposed and analyzed [14]. The proposed SF features an arc of a circle as centroidal axis and an annulus sector as cross-section; circle and annulus have a common center coinciding to that of the desired prevalent spherical motion; the axis of the smaller SF central moment of inertia points towards the desired center of prevalent spherical motion.

In this context, this chapter presents a fully compliant spherical chain that is made by the in-series connection of two identical primitive SFs with coinciding centers of spherical motion (see Fig. 1). In particular, the stiffness analysis of the proposed fully compliant spherical chain is addressed. Simulation results are compared to those of a similar chain (see Fig. 2) employing Straight Beam Flexures (SBFs). Overall, the stiffness analysis highlights that the use of two primitive SFs makes it possible to conceive fully compliant spherical chains with two independent prevalent directions of rotation and with reduced parasitic translational motions.

2 Background Theory

Within the validity limits of the superposition principle (which assumes linear elastic materials and small deflections), the kinetostatic behavior of a flexural hinge in the 3D space can be deduced by the analysis of its compliance matrix [18]. With reference to Fig. 1, given an external perturbation wrench, ${}^{p}\mathbf{w}$, acting on one point O_p of the end link and whose components are expressed with respect to the end link coordinate frame S_p (with axis x_p, y_p, z_p and origin in the point O_p), the incremental displacement vector, ${}^{p}\mathbf{s}$, can be expressed as:

$$ {}^{p}\mathbf{s} = \begin{bmatrix} {}^{p}\mathbf{u} \\ \hdashline {}^{p}\boldsymbol{\theta} \end{bmatrix} = {}^{p}\mathbf{C} \cdot \begin{bmatrix} {}^{p}\mathbf{f} \\ \hdashline {}^{p}\mathbf{m} \end{bmatrix} = {}^{p}\mathbf{C} \cdot {}^{p}\mathbf{w} \tag{1} $$

where ${}^{p}\mathbf{s}$ is composed of an incremental translation ${}^{p}\mathbf{u} = \begin{bmatrix} u & v & w \end{bmatrix}^{\mathrm{T}}$ and an incremental rotation ${}^{p}\theta = \begin{bmatrix} \alpha & \phi & \psi \end{bmatrix}^{\mathrm{T}}$, whereas ${}^{p}\mathbf{w}$ is composed of a force vector ${}^{p}\mathbf{f} = \begin{bmatrix} f_x & f_y & f_z \end{bmatrix}^{\mathrm{T}}$ and a torque vector ${}^{p}\mathbf{m} = \begin{bmatrix} m_x & m_y & m_z \end{bmatrix}^{\mathrm{T}}$. As a consequence, the compliance matrix ${}^{p}\mathbf{C}$ is a 6×6 matrix with frame-dependent entries of non homogenous physical dimensions, which relates the external wrench to the resulting translations and rotations. Concerning the frame dependency, as explained in [18], compliance matrices at different reference frames (e.g. from the frame S_p to a generic frame S_0) can be related resorting to the 6×6 *adjoint matix* ${}^{p}\mathbf{T}_0$:

$$ {}^{p}\mathbf{T}_0 = \begin{bmatrix} {}^{p}\mathbf{R}_0 & \mathbf{0} \\ \hdashline {}^{p}\widetilde{\mathbf{r}}_0 \cdot {}^{p}\mathbf{R}_0 & {}^{p}\mathbf{R}_0 \end{bmatrix} = \begin{bmatrix} {}^{0}\mathbf{R}_p^{\mathrm{T}} & \mathbf{0} \\ \hdashline \left({}^{0}\widetilde{\mathbf{r}}_p \cdot {}^{0}\mathbf{R}_p\right)^{\mathrm{T}} & {}^{0}\mathbf{R}_p^{T} \end{bmatrix} \tag{2} $$

where ${}^{p}\mathbf{R}_0$ denotes the rotation matrix of frame S_0 with respect to frame S_p (i.e. the columns of ${}^{p}\mathbf{R}_0$ are the unit vectors of frame S_0 expressed in the coordinate frame S_p), and ${}^{0}\widetilde{\mathbf{r}}_p$ denotes the skew symmetric matrix of the position vector ${}^{p}\mathbf{r}_0$, which locates the origin of frame S_0 with respect to frame S_p. Specifically, once the compliance matrix ${}^{p}\mathbf{C}$ at frame S_p is known (Fig. 1), the compliance matrix related to the frame S_0 can be simply calculated as:

$$ {}^{0}\mathbf{C} = {}^{0}\mathbf{T}_p^{-\mathrm{T}} \cdot {}^{p}\mathbf{C} \cdot {}^{0}\mathbf{T}_p^{-1} = {}^{p}\mathbf{T}_0^{\mathrm{T}} \cdot {}^{p}\mathbf{C} \cdot {}^{p}\mathbf{T}_0 \tag{3} $$

where T denotes the transpose of a matrix. As reported in [2], the 6×6 *adjoint matrix* is also useful for characterizing a collection of n in-series connected compliant flexures. In this case, the overall system compliance can be simply obtained by summing up the compliance matrices of each flexure, once all matrices are related to a common reference frame S_0. Therefore, resorting to Eq. (3), the following relation holds:

$$ {}^{0}\mathbf{C} = \sum_{k}^{n} {}^{0}\mathbf{T}_k^{-\mathrm{T}} \cdot {}^{k}\mathbf{C} \cdot {}^{0}\mathbf{T}_k^{-1} = \sum_{k}^{n} {}^{k}\mathbf{T}_0^{\mathrm{T}} \cdot {}^{k}\mathbf{C} \cdot {}^{k}\mathbf{T}_0 \tag{4} $$

3 Closed-Form Compliance Equations for the Spherical 2-SF Chain

Owing to the general remarks reported in the previous section, the procedure for deriving the closed-form compliance equations in the case of a single SF is firstly analyzed. Similarly to [8] and referring to Fig. 3, let one consider a cantilever curved beam with a uniform cross section fixed at one end (Node 1) and generically loaded at the other end (Node 2, the free end). Node 1 and node 2 are located on the beam fixed and free end respectively. In addition, let one define a global coordinate system S_k located at Node 2, and a local coordinate system S_l located on the centroid of a generic beam cross section. Let the external load, ${}^k\mathbf{w}$, and the corresponding deformation, ${}^k\mathbf{s}$, be expressed with respect to S_k. Considering the centroidal axis (i.e. curve C in Fig. 3), the relative position and orientation of the local, S_l, and global, S_k, coordinate systems can be expressed as a function of the curvilinear coordinate s, by means of vector ${}^k\mathbf{r}(s)$ and matrix ${}^l\mathbf{R}_k(s)$. With reference to Fig. 3, the generic load ${}^k\mathbf{w}$ acting at the free end is balanced by a load ${}^l\mathbf{w}'$ acting on the element ds. This load ${}^l\mathbf{w}'$ produces a deformation $d^l\mathbf{s}'$ of the same element, ${}^l\mathbf{E}$ being the corresponding deformation per unit length. The analytical relations between vectors $d^l\mathbf{s}'$, ${}^l\mathbf{w}'$ and ${}^l\mathbf{E}$ can be expressed as follows:

$$d^l\mathbf{s}' = {}^l\mathbf{E} \cdot ds; \qquad {}^l\mathbf{w}' = \mathbf{K} \cdot {}^l\mathbf{E} \tag{5}$$

The matrix $\mathbf{K}$ is the stiffness matrix of the element ds that can be written as:

$$\mathbf{K} = \mathrm{Diag}\left[EA,\ b_m GA,\ b_n GA,\ GJ,\ EI_m,\ EI_n\right] \tag{6}$$

where A, b_m, b_n, I_m, I_n, J, E and G are, respectively, cross section area, shear coefficients, principal moments of inertia and polar moment of inertia of the beam's cross section, Young's modulus and shear modulus of the employed material. Resorting to Eq. (2), the load ${}^l\mathbf{w}'$, acting on ds and due to the presence of a load ${}^k\mathbf{w}$ at the free end, can be computed via the adjoint transformation matrix ${}^l\mathbf{T}_k$, relating global and local coordinate systems. In particular, the following relation holds:

$${}^l\mathbf{w}' = {}^l\mathbf{T}_k \cdot {}^k\mathbf{w} \tag{7}$$

having written the matrix ${}^l\mathbf{T}_k$ as a function of the curvilinear coordinate s, such that:

$${}^k\mathbf{T}_l(s) = \left[\begin{array}{c:c} {}^k\mathbf{R}_l(s) & \mathbf{0} \\ \hdashline {}^k\mathbf{R}_l(s) \cdot {}^k\tilde{\mathbf{r}}(s) & {}^k\mathbf{R}_l(s) \end{array}\right]. \tag{8}$$

In addition, the deformation of the element ds, denoted as $d^l\mathbf{s}'$, causes a deformation at the free end, $d^k\mathbf{s}$, that can be calculated as:

$$d^k\mathbf{s} = {}^l\mathbf{T}_k^T \cdot d^l\mathbf{s}' \tag{9}$$

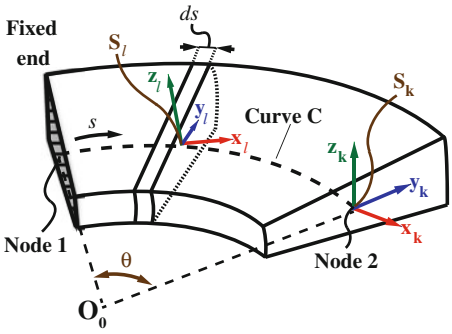


Fig. 3 SF loaded at the free end

By merging Eqs. (5), (7) and (9) one can obtain:

$$d^k\mathbf{s} = {}^l\mathbf{T}_k^T \cdot \mathbf{K}^{-1} \cdot {}^l\mathbf{T}_k \cdot {}^k\mathbf{w} \cdot ds \tag{10}$$

The relation between the load ${}^k\mathbf{w}$ and the deformation ${}^k\mathbf{s}$ of the free node can then be found by integrating Eq. (10), as follows:

$${}^k\mathbf{s} = {}^k\mathbf{C} \cdot {}^k\mathbf{w} \tag{11}$$

where

$${}^k\mathbf{C} = \int_C {}^l\mathbf{T}_k^T \cdot \mathbf{K}^{-1} \cdot {}^l\mathbf{T}_k \cdot ds \tag{12}$$

Matrix ${}^k\mathbf{C}$ is the compliance matrix for a single cantilever curved beam loaded at the free end, as referred to the coordinate system S_k located on Node 2. For multiple cantilever curved beams connected in series, the overall compliance can be obtained resorting to Eq. (4), after a common reference frame S_0 is chosen.

In the following, the procedure is outlined for the fully compliant spherical chain depicted in Fig. 1. The chain is composed by the in-series connection of two identical spherical flexures (hereafter referred to as SF#1 and SF#2) with centroidal axis lying on the same circumference. Let one denote as ${}^1\mathbf{C}$ and ${}^2\mathbf{C}$ the compliance matrices of each SF as referred to the hinge end (similarly to Eq. (12)). The center of the spherical motion (i.e. point O_0 in Fig. 1) is then taken as the origin of S_0, the frame axis being oriented such that y_0 axis passes through the centroid of the SF#1 mid cross section, whereas z_0 axis is orthogonal to the symmetry plane containing the centroidal axis (see Fig. 4).

As a further step, the adjoint matrices ${}^1\mathbf{T}_0$ and ${}^2\mathbf{T}_0$, respectively relating ${}^1\mathbf{C}$ and ${}^2\mathbf{C}$ to S_0, should be computed. Recalling Eq. (2), the rotation matrices, ${}^1\mathbf{R}_0$ and ${}^2\mathbf{R}_0$, and the translation vectors, ${}^1\mathbf{r}_0$ and ${}^2\mathbf{r}_0$, composing ${}^1\mathbf{T}_0$ and ${}^2\mathbf{T}_0$ are defined as follows:

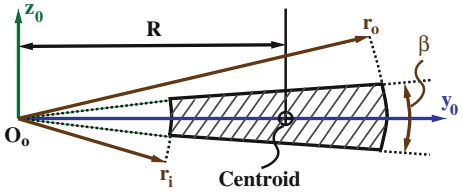


Fig. 4 Cross section properties of SFs

$$
{}^{1}\mathbf{R}_0 = \begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) & 0 \\ \sin(\theta/2) & \cos(\theta/2) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{13}
$$

$$
{}^{2}\mathbf{R}_0 = \begin{bmatrix} \cos(\theta/2+\delta) & -\sin(\theta/2+\delta) & 0 \\ \sin(\theta/2+\delta) & \cos(\theta/2+\delta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{14}
$$

$$
{}^{1}\mathbf{r}_0 = {}^{2}\mathbf{r}_0 = \begin{bmatrix} 0, & -R, & 0 \end{bmatrix}^{\mathrm{T}} \tag{15}
$$

where R and θ represent the radius and subtended angle of SF#1 and SF#2 centroidal axis, whereas δ is the angle between the y_0 axis and an axis connecting point O_0 and the centroid of the SF#2 mid cross section. The overall chain compliance matrix can then be computed resorting to Eq. (4), where $n = 2$. In particular, the matrix ${}^{0}\mathbf{C}$ can be expressed as follows:

$$
{}^{0}\mathbf{C} = \begin{bmatrix} C_{x,f_x} & C_{x,f_y} & 0 & 0 & 0 & C_{x,m_z} \\ C_{y,f_x} & C_{y,f_y} & 0 & 0 & 0 & C_{y,m_z} \\ 0 & 0 & C_{z,f_z} & C_{z,m_x} & C_{z,m_y} & 0 \\ 0 & 0 & C_{\theta_x,f_z} & C_{\theta_x,m_x} & C_{\theta_x,m_y} & 0 \\ 0 & 0 & C_{\theta_y,f_z} & C_{\theta_y,m_x} & C_{\theta_y,m_y} & 0 \\ C_{\theta_z,f_x} & C_{\theta_z,f_y} & 0 & 0 & 0 & C_{\theta_z,m_z} \end{bmatrix} \tag{16}
$$

Matrix ${}^{0}\mathbf{C}$ relates the wrench ${}^{0}\mathbf{w}$ acting on ${}^{0}\mathbf{C}$ to the corresponding generalized displacement ${}^{0}\mathbf{s}$, (namely ${}^{0}\mathbf{s} = {}^{0}\mathbf{C} \cdot {}^{0}\mathbf{w}$). The analytical expression of the matrix entries are reported in Table 1.

It can be noticed that each compliance element is written in terms of the cross section area, principal and polar moments of inertia (namely A, I_m, I_n, and J). In particular, the SF cross section is an annular sector (Fig. 4), that can be considered as the common section of two concentric circular sectors with different radius. Therefore, denoting θ as the subtended angle of the SF centroidal axis (see Fig. 3), β as the subtended angle of the circular sector in the $y_0 - z_0$ plane, and r_i and r_o as the radius of the inner and the outer circular sectors respectively (see Fig. 4), the cross section properties can be written as a function of the hinge geometric parameters, the analytical expression being reported in Table 2.

Table 1 Compliance elements of the mechanism

$C_{x,f_x} = -\frac{R\left(-2\left(AR^2+I_n\right)\left(\cos^2(\delta)\sin(\theta)+\theta\right)\right)}{2EAI_n}$	$C_{x,f_y} = C_{y,f_x} = -\frac{R\left(\sin(2\delta)\sin(\theta)\left(AR^2+I_n\right)\right)}{2EAI_n}$
$C_{y,f_y} = \frac{R\left(2\left(AR^2+I_n\right)\left(\theta-\cos^2(\delta)\sin(\theta)\right)\right)}{2EAI_n}$	$C_{x,m_z} = C_{\theta_z,f_x} = -\frac{R^2\left(-4\cos^2\left(\frac{\delta}{2}\right)\sin\left(\frac{\theta}{2}\right)\right)}{EI_n}$
$C_{y,m_z} = C_{\theta_z,f_z} = -\frac{R^2\left(2\sin(\delta)\sin\left(\frac{\theta}{2}\right)\right)}{EI_n}$	$C_{z,m_x} = C_{\theta_x,f_z} = \frac{R^2\left(-4\cos^2\left(\frac{\delta}{2}\right)\sin\left(\frac{\theta}{2}\right)\right)}{\mathrm{GJ}}$
$C_{\theta_x,m_x} = -\frac{R\left(-2GJ\left(\theta-\cos^2(\delta)\sin(\theta)\right)-2EI_m\left(\cos^2(\delta)\sin(\theta)+\theta\right)\right)}{2EGJI_m}$	$C_{z,f_z} = 2\,\frac{R^3\theta}{GJ}$
$C_{\theta_x,m_y} = C_{\theta_y,m_x} = R\sin(\delta)\cos(\delta)\sin(\theta)\left(\frac{1}{EI_m}-\frac{1}{GJ}\right)$	$C_{z,m_y} = C_{\theta_y,f_z} = \frac{R^2\left(2\sin(\delta)\sin\left(\frac{\theta}{2}\right)\right)}{\mathrm{GJ}}$
$C_{\theta_y,m_y} = \frac{R\left(2GJ\left(\cos^2(\delta)\sin(\theta)+\theta\right)+2EI_m\left(\theta-\cos^2(\delta)\sin(\theta)\right)\right)}{2EGJI_m}$	$C_{\theta_z,m_z} = 2\,\frac{R\theta}{EI_n}$

Table 2 Spherical flexure areas and moments of inertia

$A = \frac{(r_o^2-r_i^2)\beta}{2}$	$I_m = \frac{1}{8}(r_o^4-r_i^4)(\beta-\sin\beta)$
$I_n = \frac{1}{8}(r_o^4-r_i^4)(\beta+\sin\beta) - \frac{8}{9}\frac{(r_o^3-r_i^3)^2\sin^2(\beta/2)}{(r_o^2-r_i^2)\beta}$	$J = \frac{1}{4}(r_o^4-r_i^4)\beta - \frac{8}{9}\frac{(r_o^3-r_i^3)^2\sin^2(\beta/2)}{(r_o^2-r_i^2)\beta}$

4 Numerical Example and Model Validation

A fully-compliant spherical chain featuring two identical in-series SFs is considered as a case study. The SF geometric parameters employed in the simulations are $R = 55.2\,\mathrm{mm}$, $r_o = 60\,\mathrm{mm}$, $r_i = 50\,\mathrm{mm}$, $\theta = \pi/3$ and $\beta = \pi/180$. The hinge material is Acrylic Plastic with Young's modulus $E = 3{,}000\,\mathrm{MPa}$ and Poisson's ratio $\nu = 0.33$. Shear induced deformations are neglected, both SFs being slender beam hinges. For a generic angle δ, the aforementioned theoretical procedure is adopted to estimate the overall compliance matrix.The dependency of the principal compliance elements, C_{θ_x,m_x} and C_{θ_y,m_y}, on the angle δ is shown in Fig. 5, which highlights that $C_{\theta_x,m_x} = C_{\theta_y,m_y}$ when $\delta = 90°$. Results concerning this particular geometry (namely $\delta = 90°$) are then validated through FEA performed with the commercial software COMSOL. FEA simulations are executed by individually loading the chain along the axes x_0, y_0 and z_0 (see Fig. 1). The compliance elements are simply computed as the ratios between each load component and the corresponding deformations. Table 3 compares the results obtained via analytical model and FEA. The comparison shows a close agreement between the two methods.

As a further step, an SF-based chain has been compared with a similar chain featuring SBF flexures and equal values of the principal compliance elements. Defining w and l as the SBF widht and length, the latter design constraint is achieved whenever both SBFs and SFs are carachterized by equal width (i.e. $w = r_o - r_i$), and same centroidal axis lenght (i.e. $l = R\theta$). By applying the method described in Sect. 3, the compliance matrix of the SBF-based chain has been computed and numerical data are shown in Table 4. The quantitative comparison between the two design solutions is then performed by defining three compliance ratios as follows:

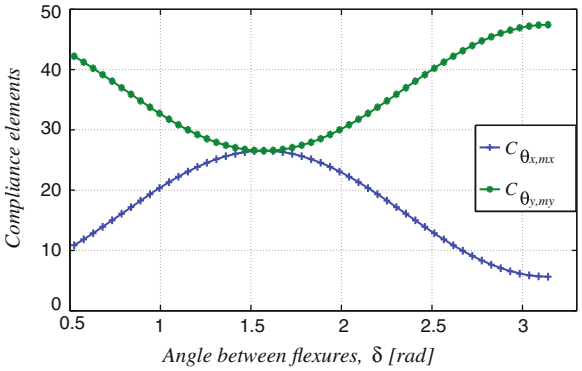


Fig. 5 Influence of varying δ on principal compliances

Table 3 Compliance elements of the 2-SF spherical chain and comparison between analytical and FEA results

Compliance elements	C_{x,f_x}	$C_{x,f_y} = C_{y,f_x}$	$C_{x,m_z} = C_{\theta_z,f_x}$	C_{y,f_y}	$C_{y,m_z} = C_{\theta_z,f_y}$	C_{z,f_z}
Analytic	7.3603e-4	−5.9730e-20	0.0127	7.3603e-4	−0.0127	0.0039
FEA	7.3631e-4	−5.9000e-20	0.0127	7.3635e-4	−0.0127	0.0038
Error (%)	0.038	1.22	0	0.038	0	2.56
Compliance elements	C_{θ_x,m_x}	$C_{z,m_x} = C_{\theta_x,f_z}$	$C_{z,m_y} = C_{\theta_y,f_z}$	C_{θ_y,m_y}	$C_{\theta_x,m_y} = C_{\theta_y,m_x}$	C_{θ_z,m_z}
Analytic	26.5379	−0.0334	0.0334	26.5379	−4.9013e-15	0.4827
FEA	26.5455	−0.0334	0.0334	26.5455	−4.9000e-15	0.4842
Error (%)	0.028	0	0	0.028	0.026	0.31

Table 4 Compliance elements of the 2-SBF spherical chain

Compliance elements	C_{x,f_x}	$C_{x,f_y} = C_{y,f_x}$	$C_{x,m_z} = C_{\theta_z,f_x}$	C_{y,f_y}	$C_{y,m_z} = C_{\theta_z,f_y}$	C_{z,f_z}
Value	8.0311e-4	−4.0926e-20	0.0133	8.0311e-4	−0.0133	0.0183
Compliance elements	C_{θ_x,m_x}	$C_{z,m_x} = C_{\theta_x,f_z}$	$C_{x,m_y} = C_{\theta_y,f_z}$	C_{θ_y,m_y}	$C_{\theta_x,m_y} = C_{\theta_y,m_x}$	C_{θ_z,m_z}
Value	26.5379	−0.0350	0.0350	26.5379	−1.5472e-15	0.4827

$$r_1 = \frac{|C^{SF}_{x,f_x}|}{|C^{SBF}_{x,f_x}|}; \quad r_2 = \frac{|C^{SF}_{x,m_z}|}{|C^{SBF}_{x,m_z}|} = \frac{|C^{SF}_{\theta_z,f_x}|}{|C^{SBF}_{\theta_z,f_x}|} = \frac{|C^{SF}_{z,m_x}|}{|C^{SBF}_{z,m_x}|} = \frac{|C^{SF}_{\theta_x,f_z}|}{|C^{SBF}_{\theta_x,f_z}|}; \quad r_3 = \frac{|C^{SF}_{z,f_z}|}{|C^{SBF}_{z,f_z}|}; \tag{17}$$

where each compliance element concerning the SF-based chain (referred to with SF superscript) is divided by the corresponding compliance element concerning the SBF-based chain (referred to with SBF superscript). In particular, the variables r_1, r_2, and r_3 represent ratios between undesired (secondary) compliances in all

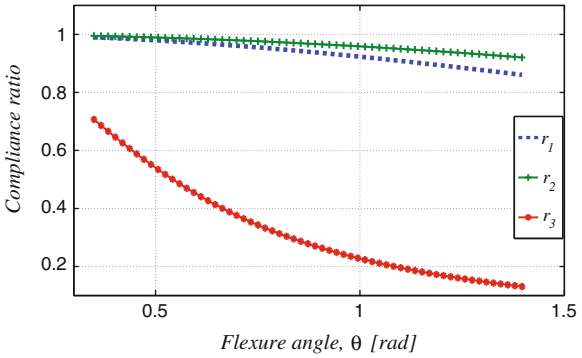


Fig. 6 Influence of varying θ on compliance ratios

Fig. 7 Finite element model of the 2-SF chain

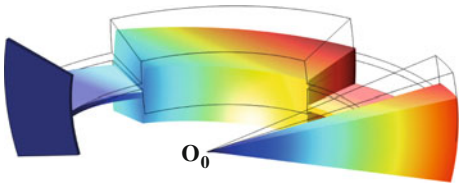
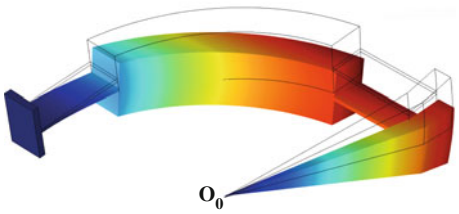


Fig. 8 Finite element model of the 2-SBF chain



those applications requiring a spherical motion. Therefore, compliance ratios whose value is less than unity simply indicates that the SF-based chain outperformes the SBF-based chain in terms of parasitic motions. As an example, for a given radius $R = 55.2\,\text{mm}$, the values of r_1, r_2 and r_3 as a function of θ are reported in Fig. 6. For a given angle $\theta = \pi/3$, the compliance ratios are constant (independent of R), namely $r_1 = 0.9191$, $r_2 = 0.9549$ and $r_3 = 0.2115$. In conclusion, for what concerns this particular case study, numerical simulations confirm the benefits when using the primitive SF as compared to the traditional SBF. Finally, pictures of the two considered design solutions are reported in Figs. 7 and 8 which also show the chain deformed shapes (contour plot of the total displacement) when a generic torque lying in $x_0 - y_0$ plane is applied to the end-link.

5 Conclusions

A fully compliant spherical chain featuring two in-series connected identical primitive spherical flexures with coincident centers of spherical motion is introduced and analyzed. The closed form compliance equations of the proposed chains are presented as a function of hinge dimensions, relative location and employed material. Comparison with an equivalent chain featuring two straight beam hinges with axes intersecting in the sought center of spherical motion is also performed in terms of secondary compliance elements, together with their finite element verification. The results highlight that the use of primitive spherical flexures makes it possible to conceive fully compliant spherical chains with reduced parasitic motions.

References

1. Callegari, M., Cammarata, A., Gabrielli, A., Ruggiu, M., Sinatra, R.: Analysis and design of a spherical micromechanism with flexure hinges. ASME J. Mech. Des. **131**, 051003 (2009)
2. Carter Hale, L.: Principles and techniques for designing precision machines. Ph.D. thesis, Departtment of Mechanical Enggineering, MIT, Cambridge (1999)
3. Chiang, C.: Kinematics of Spherical Mechanisms. Krieger Publishing Company, Florida (2000)
4. Farhadi Machekposhti, D., Tolou, N., Herder, J.L.: The scope for a compliant homokinetic coupling based on review of compliant joints and rigid-body constant velocity universal joints. In: ASME IDETC, International Design Engineering Technical Conferences vol. 4, pp. 379–392 (2012)
5. Hesselbach, J., Wrege, J., Raatz, A., Becker, O.: Aspects on design of high precision parallel robots. Assembly Autom. **24**(1), 49–57 (2004)
6. Howell, L.L.: Compliant Mechanisms. Wiley, New York (2001)
7. Jacobsen, J.O., Chen, G., Howell, L.L., Magleby, S.P.: Lamina emergent torsional (LET) joint. Mechanims and Machine Theory **44**(11), 2098–2109 (2009)
8. Jafari, M., Mahjoob, M.: An exact three-dimensional beam element with nonuniform cross section. ASME J. Appl. Mech. **77**(6) (2010)
9. Li, G., Chen, G.: Achieving compliant spherical linkage designs from compliant planar linkages based on prbm: a spherical young mechanism case study. In: IEEE ROBIO, International Conference on Robotics and Biomimetics, pp. 193–197. IEEE (2012)
10. Lobontiu, N.: Compliant Mechanisms: Design of Flexure Hinges. CRC Press, Boca Raton (2002)
11. Lobontiu, N., Garcia, E.: Two-axis flexure hinges with axially-collocated and symmetric notches. Comput. Struct. **81**(13), 1329–1341 (2003)
12. Lobontiu, N., Paine, J.: Design of circular cross-section corner-filleted flexure hinges for three-dimensional compliant mechanisms. ASME J. Mech. Des. **124**(3), 479–484 (2002)
13. Palmieri, G., Palpacelli, M.C., Callegari, M.: Study of a fully compliant u-joint designed for minirobotics applications. ASME J. Mech. Des. **134**(11), 111003(9) (2012)
14. Parvari Rad, F., Berselli, G., Vertechy, R., Parenti-Castelli, V.: Compliance based characterization of spherical flexure hinges for spatial compliant mechanisms. In: 19th CISM-IFToMM Symposium on Robot Design, Dynamics, and Control, pp. 1–8 (2014) (Submitted to Romansy)
15. Pei, X., Yu, J., Zong, G., Bi, S., Su, H.: The modeling of cartwheel flexural hinges. Mech. Mach. Theory **44**(10), 1900–1909 (2009)
16. Smith, S.: Flexures, Elements of Elastic Mechanisms. Gordon and Breach Science Pub, New York (2000)

17. Trease, B., Moon, Y., Kota, S.: Design of large-displacement compliant joints. ASME J. Mech. Des. **127**(4), 788–798 (2005)
18. Zhang, S., Fasse, E.D.: A finite-element-based method to determine the spatial stiffness properties of a notch hinge. ASME J. Mech. Des. **123**(1), 141–147 (2001)

Points, Lines, Screws and Planes in Dual Quaternions Kinematics

Luiz Alberto Radavelli, Edson Roberto De Pieri, Daniel Martins and Roberto Simoni

Abstract Quaternions and dual quaternions are interesting elements which are being used to robot kinematics over five decades. They arise from Clifford algebras as many isomorphisms. In this chapter we offer representations to points, vectors, lines, screws and planes in dual quaternions coordinates, allowing a huge possibilities to solve problems, especially robot kinematics. No Clifford algebra is necessary, we will use only quaternions units. The displacement of the given elements are found in terms of dual quaternions algebra. For all these elements we must define the right dual quaternions conjugation and operations to handle with. Also, the principle of transference now is not sufficient, as we will explain into the chapter. Examples are presented to show the applicability of our results.

Keywords Quaternions · Dual quaternions · Point · Line · Screw · Plane · Kinematics

1 Introduction

Robot positional kinematics studies positions and orientations of robots. It relates the actuated-joint parameters with the end-effector position and orientation, called posture of end-effector. The actuated-joint parameters are the angles of rotation and

L. A. Radavelli (✉) · E. R. De Pieri · D. Martins
Federal University of Santa Catarina, Campus Florianópolis, Florianópolis, Brazil
e-mail: luiz.radavelli@ufsc.br

E. R. De Pieri
e-mail: edson@das.ufsc.br

D. Martins
e-mail: danielemc@gmail.com

R. Simoni
Federal University of Santa Catarina, Campus Joinville, Joinville, Brazil
e-mail: roberto.simoni@ufsc.br

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_30,

the distance of translations of the joints. Positional kinematics is commonly divided into two ways: *direct kinematics*, in which we have the actuated-joint variables and we study the posture of the end-effector; and in the opposite way, the *inverse kinematics*, in which the posture of the end-effector is given and the problem is to calculate the actuated-joint variables.

Traditionally, the robot positional kinematics is obtained by Denavit-Hartenberg method, but the dual numbers theory have being used as an alternatively tool to kinematics over five decades. The definitions, operations and properties of dual quaternions arise from a more general algebra: The *Clifford* algebras [9]. There are many ways to get the dual quaternions algebra from Clifford algebras. So, firstly it is necessary to establish a convenient notation to work with them.

Dual quaternions are elements composed by scalar, vector and dual numbers. The first work applying dual numbers into kinematics is due to Yang and Freudenstein [19]. After, several authors have shown advantages using dual quaternions algebra in robot kinematics. Some of them have more interest on computational analysis [8, 10], others on singularities [13]. Recent works using dual quaternions appear in control theory and dynamic [15–17], but for the geometrical [2, 3, 11], elements treatment the most interesting papers are.

2 Quaternions and Dual Quaternions Algebras

There are several ways to define a Clifford Algebra and the definition will depend on its purpose [6]. A Clifford Space is an extension of an Euclidean vector space that works with more general concepts—multivectors. A Clifford Space becomes a Clifford Algebra when a product between multivectors, called *geometric product* or *Clifford product*, is defined.

Basically, considering an orthonormal basis $\{e_1, \ldots, e_n\}$ of Euclidean vector space $\mathbb{R}^n$, the corresponding 2^n-dimensional Clifford Algebra $\mathrm{C}l(p, q, r)$, $n = p + q + r$, is composed by combinations of k-degree elements $e_1 e_2 \ldots e_k$ with a geometrical product satisfying

1. $e_i e_j + e_j e_i = 0, \quad \text{if } i \neq j;$
2. $e_i^2 = \varepsilon_i$

where $\varepsilon_i = +1, -1, 0$ represents the signature of a generator e_i, such that p, q, r are the number of generators with signatures $+1, -1, 0$, respectively.

The operations between Clifford elements may be reduced to axioms 1. and 2. above, and the conjugate operation on generations

1. $e_i^* = -e_i;$
2. $(e_i e_j)^* = e_j^* e_i^*.$

A Clifford algebra has some decompositions and isomorphisms which allow many representations for quaternions and dual quaternions. Indeed, an important decomposition is $\mathrm{C}l(p, q, r) = \mathrm{C}l^+(p, q, r) \oplus \mathrm{C}l^-(p, q, r)$ where $\mathrm{C}l^+(p, q, r)$ is a sub-

algebra of even degree elements, named *Spinors*. The isomorphism $C\ell^+(p, q, r) \simeq C\ell(p, q-1, r)$ produces $C\ell^+(0, 3, 0) \simeq C\ell(0, 2, 0) = \mathbb{H}$—the Quaternions space—and $C\ell(0, 2, 1) \simeq C\ell^+(0, 3, 1) = \mathbb{H}_2$—the dual quaternions space. Therefore, the Clifford generators, quaternions and dual quaternions units are related by $e_2e_3 \leftrightarrow i$, $e_3e_1 \leftrightarrow j$, $e_1e_2 \leftrightarrow k$, $ee_1e_2e_3 \leftrightarrow \varepsilon$, $e_1e \leftrightarrow \varepsilon i$, $e_2e \leftrightarrow \varepsilon j$, $e_3e \leftrightarrow \varepsilon k$.

If $q \in C\ell^+(0, 3, 0)$, then $q = w + xe_2e_3 + ye_3e_1 + ze_1e_2$, $e_1^2 = e_2^2 = e_3^2 = -1$, correspond to the quaternion $q = w + xi + yj + zk$, where $i^2 = j^2 = k^2 = ijk = -1$. If $w = 0$, then the quaternion q correspond to $3D$ vectors and it is called *pure quaternion*. Thus, it is natural to think quaternions as the sum of a scalar and a vector:

$$q = w + \mathbf{v} = Sc(q) + Ve(q).$$

Let $q = w + xi + yj + zk = w + \mathbf{v}$ and $q_2 = w_2 + x_2i + y_2j + z_2k = w_2 + \mathbf{v}_2$ quaternions, the usual operations are:

$$q^* := w - \mathbf{v}, \quad Sc(q) := \frac{q + q^*}{2}, \quad Ve(q) := \frac{q - q^*}{2}, \quad \|q\|^2 := qq^* = q^*q,$$
$$q \pm q_2 := (w \pm w_2) + (\mathbf{v} \pm \mathbf{v}_2)q \qquad q_2 := (ww_2 - \mathbf{v} \cdot \mathbf{v}_2) + (w\mathbf{v}_2 + w_2\mathbf{v} + \mathbf{v} \times \mathbf{v}_2).$$

To pure quaternions we have interesting properties like $qq_2 = \mathbf{q}\mathbf{q}_2 = -\mathbf{q}\cdot\mathbf{q}_2 + \mathbf{q}\times\mathbf{q}_2$ which establishes the relation among quaternions and $\mathbb{R}^3$ inner and outer products. From the pure quaternions product follows more two identities:

$$\mathbf{q}\mathbf{q}_2 + \mathbf{q}_2\mathbf{q} = -2(\mathbf{q} \cdot \mathbf{q}_2) \tag{1}$$
$$\mathbf{q}\mathbf{q}_2 - \mathbf{q}_2\mathbf{q} = 2(\mathbf{q} \times \mathbf{q}_2). \tag{2}$$

A rotation of a quaternion $\xi = 0 + \mathbf{r}$,$\mathbf{r} \in \mathbb{R}^3$, is given by $\xi' = q\xi q^*$, where $q = \cos\frac{\theta}{2} + \mathbf{s}\left(\sin\frac{\theta}{2}\right)$ is the rotation quaternion operator which encodes a rotation by θ around the axis represented by the unit direction vector $\mathbf{s}$ [9, 12]. A more general form of rotations is given by

$$\xi' = q(\xi - \mathbf{s}_0)q^* + \mathbf{s}_0, \tag{3}$$

where $\mathbf{s}_0$ is the rotational axis position vector [9]. Another important result states that if ξ is a pure quaternion then $\xi' = q\xi q^*$ is also a pure quaternion. Indeed, the scalar part of ξ' is

$$Sc(q\xi q^*) = \frac{q\xi q^* + (q\xi q^*)^*}{2} = \frac{q\xi q^* + q\xi^* q^*}{2} = \frac{q(\xi + \xi^*)q^*}{2} = 0. \tag{4}$$

Now, if we consider $h \in C\ell^+(0, 3, 1)$, then $h = w + xe_2e_3 + ye_3e_1 + ze_1e_2 + w_2ee_1e_2e_3 + x_2e_1e + y_2e_2e + z_2e_3e$, $e_1^2 = e_2^2 = e_3^2 = -1$ and $e^2 = 0$, correspond to dual quaternions. Using the algebra isomorphism $C\ell^+(0, 3, 1) \simeq \mathbb{H}_2$, the dual quaternion main representation is

$$h = w + xi + yj + zk + w_2\varepsilon + x_2 i\varepsilon + y_2 j\varepsilon + z_2 k\varepsilon,$$

but we also have the *generalized dual number* and the *generalized quaternion* representations, respectively,

$$h = (w + xi + yj + zk) + \varepsilon(w_2 + x_2 i + y_2 j + z_2 k),$$
$$h = (w + w_2) + (x + x_2\varepsilon)i + (y + y_2\varepsilon)j + (z + z_2\varepsilon)k.$$

Let $h = q + \varepsilon p$ and $h_2 = q_2 + \varepsilon p_2$ dual quaternions, the usual operations are:

$$h^* = q^* + \varepsilon p^*, \quad \overline{h} = q - \varepsilon p, \quad \overline{h^*} = q^* - \varepsilon p^*,$$
$$h \pm h_2 = (q \pm q_2) + \varepsilon(p \pm p_2), \quad hh_2 = (qq_2) + \varepsilon(qp_2 + pq_2).$$

A rotation by θ with axis represented by the unit vector $\mathbf{s}$ and a translation by vector $\mathbf{t}$ are performed from the dual quaternions by $h_R := \cos\frac{\theta}{2} + \mathbf{s}\left(\sin\frac{\theta}{2}\right) + \varepsilon 0 + \varepsilon\mathbf{0} = q + \varepsilon 0 + \varepsilon\mathbf{0}$ and $h_T := 1 + \mathbf{0} + \varepsilon 0 + \varepsilon\frac{\mathbf{t}}{2}$. Thus, the general displacement operator (a rotation followed by a translation) is

$$h := h_T\, h_R = q + \varepsilon\frac{tq}{2}, \tag{5}$$

where $q = \cos\frac{\theta}{2} + \mathbf{s}(\sin\frac{\theta}{2})$ and $t = 0 + \mathbf{t}$ are its rotational and translational components, respectively [9, 12].

A convenient way to represent quaternions q and dual quaternions h is to think $q \in \mathbb{R}^4$ and $h \in \mathbb{R}^8$, therefore $q = [w\, x\, y\, z]^T = q(1:4)$, $h = [w\, x\, y\, z\, w_2\, x_2\, y_2\, z_2]^T = h(1:8)$ and their operations can be easily implemented.

3 Points, Vectors, Lines, Screws and Planes in DQ Coordinates

In this section we will provide our definition of some geometrical elements in dual quaternions coordinates. There is a Clifford algebra representation of point, line and plane elements in *Selig* [11], but our representations are free from Clifford generators, which are very confused to handle with and have much tendencies to errors.

In $3D$ space, the representation of points and vectors is confused because a point $\mathbf{p}$ and a vector $\mathbf{v}$ are given by their components: $[x\ y\ z]^T$. In the projective space $\mathbb{R}P^3 \subset \mathbb{R}^4$ points and vectors differs each other, once we have $p = 1 + \mathbf{p} = [1\, x\, y\, z]^T$ and $r = 0 + \mathbf{v} = [0\ x\ y\ z]^T$. In dual quaternions coordinates, a point and a vector have the following representations:

$$P = 1 + \mathbf{0} + \varepsilon(0 + \mathbf{p}) = \left[1\ 0\ 0\ 0\ 0\ x\ y\ z\right]^T,$$
$$V = 0 + \mathbf{0} + \varepsilon(0 + \mathbf{v}) = \left[0\ 0\ 0\ 0\ 0\ x\ y\ z\right]^T.$$

The representation of a line L is given *Plücker* coordinates into a dual quaternions, that is, if $\mathbf{l}$ and $\mathbf{l}_0$ are the direction and position vectors of L, then $\mathbf{m} = \mathbf{l}_0 \times \mathbf{l}$ is the moment vector of L about origin of coordinate system and the dual quaternions representation of L is

$$L = 0 + \mathbf{l} + \varepsilon(0 + \mathbf{m}) = \begin{bmatrix} 0 & l_x & l_y & l_z & 0 & m_x & m_y & m_z \end{bmatrix}^T.$$

The representation of screws is similar to lines but we have an additional term to the moment vector:

$$\$ = 0 + \mathbf{s} + \varepsilon(0 + \mathbf{m} + \lambda\mathbf{s}) = \begin{bmatrix} 0 & s_x & s_y & s_z & 0 & m_x + \lambda s_x & m_y + \lambda s_y & m_z + \lambda s_z \end{bmatrix}^T,$$

where $\mathbf{s}$, $\mathbf{s}_0$ and $\mathbf{m} = \mathbf{s}_0 \times \mathbf{s}$ are the direction, position and moment vector of the screw \$, respectively, λ is the pitch of \$. The pitch is the distance parallel the axis $\mathbf{s}$ to complete one rotation around $\mathbf{s}$. So, a rotational angle θ and translational distance d are related by $\lambda = 2\pi d/\theta$.

The representation of planes follows from the classical calculus. One way to define a plane Π with normal vector $\mathbf{n}$ and support vector $\mathbf{r}_0$ is given by $\mathbf{n} \cdot (\mathbf{r} - \mathbf{r}_0) = 0 \iff \mathbf{n} \cdot \mathbf{r} = d$, where d is the distance of the plane from origin of coordinate system. In dual quaternions algebra, the inner product returns an opposite of an usual inner product. Therefore we have $\mathbf{n} \cdot \mathbf{r_0} = -d$ and the plane representation is

$$\Pi = 0 + \mathbf{n} + \varepsilon(-d + \mathbf{0}) = \begin{bmatrix} 0 & n_x & n_y & n_z & -d & 0 & 0 & 0 \end{bmatrix}^T.$$

4 Geometrical Element Displacements

In this section we will present the displacement equation for points, lines, screws and planes, which arise from geometrical analysis and dual quaternions representations. We must remember the general displacement operator (5), in which we have $h^* = q^* - \varepsilon \frac{q^* t}{2}$ and $\overline{h^*} = q^* + \varepsilon \frac{q^* t}{2}$.

Let $P = 1 + \varepsilon\mathbf{p}$ be a point in dual quaternions coordinate. A rotation of P cause a reorientation of the vector $\mathbf{p}$ to $\mathbf{p}'' = q\mathbf{p}q^*$. The translation of P moves $\mathbf{p}''$ to the new position $\mathbf{p}' = \mathbf{p}'' + \mathbf{t}$. Thus, the composition of rotation and translation of P must be

$$P' = 1 + \varepsilon(q\mathbf{p}q^* + \mathbf{t}) = 1 + \varepsilon(q\mathbf{p}q^*) + \varepsilon\frac{\mathbf{t}}{2} - \varepsilon\frac{\mathbf{t}^*}{2} = hP\overline{h^*}.$$

A vector $V = 0 + \varepsilon\mathbf{p}$ has similar equations which arise considering no translation.

Let L a line and \$ a screw in dual quaternions coordinate. A rotation of L cause a reorientation of the direction vector $\mathbf{l}$ and of the moment vector $\mathbf{m}$. A translation has no effect on $\mathbf{l}$, then: $\mathbf{l} \longmapsto \mathbf{l}' = \mathbf{q}\mathbf{l}\mathbf{q}^*$ but the moment vector has the *shifting law*: $\mathbf{m} \longmapsto \mathbf{m}' = q\mathbf{m}q^* + \mathbf{t} \times (q\mathbf{l}q^*)$—see Fig. 1(left). Once $\mathbf{l}$ and $\mathbf{t}$ are pure quaternions,

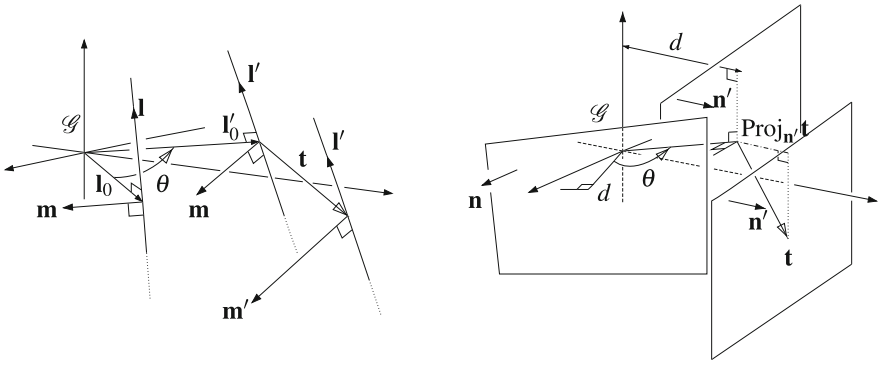


Fig. 1 General displacement of the line L (*left*) and plane Π (*right*) from two steps: a rotation from θ *rad* followed by a translation from $\mathbf{t}$

(4) assures $q\mathbf{l}q^*$ is a pure quaternion and using (2) we get:

$$L' = q\mathbf{l}q^* + \varepsilon\big[q\mathbf{m}q^* + \mathbf{t} \times (q\mathbf{l}q^*)\big] = \Big(q + \varepsilon\frac{\mathbf{t}}{2}q\Big)(\mathbf{l} + \varepsilon\mathbf{m})\Big(q + \varepsilon\frac{\mathbf{t}}{2}q\Big)^* = hLh^*.$$

Some authors use the *Principle of Transference* [7] to handle with dual quaternion displacements, saying that the dual quaternions transformations arise from quaternions rotation like $\xi' = q\xi q^* \longmapsto \xi' = h\xi h^*$ [1, 5]. Basically, the principle of transference states that the real numbers are cropped to dual ones, transforming the quaternion equation $\xi' = q\xi q^*$ into the dual quaternions equation $\breve{\xi}' = \breve{q}\breve{\xi}\breve{q}^*$, where $\breve{\xi}$, must be a dual quaternion and $\breve{\xi}$ must be a dual vector.

Let $\Pi = \mathbf{n} - \varepsilon d$ be a plane. A rotation of Π cause a reorientation of vector $\mathbf{n}$, while the translation changes the distance d from the origin. The addition to d is the projection of translation vector $\mathbf{t}$ into normal vector $\mathbf{n}$—see Fig. 1. We must have $\mathbf{n} \longmapsto \mathbf{n}' = q\mathbf{n}q^*$ and $d \longmapsto d' = d + \mathbf{t}\cdot(q\mathbf{n}q^*)$. Once $\mathbf{t}$ and $\mathbf{n}$ are pure quaternions, (4) assures $q\mathbf{n}q^*$ is a pure quaternion and using (1) we have:

$$\Pi' = q\mathbf{n}q^* - \varepsilon\big[d + \mathbf{t}\cdot(q\mathbf{n}q^*)\big] = q\mathbf{n}q^* - \varepsilon d + \varepsilon\Big[\frac{\mathbf{t}(q\mathbf{n}q^*) + (q\mathbf{n}q^*)\mathbf{t}}{2}\Big] = h\Pi\overline{h^*}.$$

In order to unified the general displacement in dual quaternions coordinate, we define a *general conjugate*:

$$\widetilde{h} = \begin{cases} \overline{h^*}, & \text{if } \xi \leftarrow P, V, \text{ or } \Pi. \\ h^*, & \text{if } \xi \leftarrow L, \text{ or \$}. \end{cases}$$

Therefore, a general displacement of any geometrical element is given by $\xi' = h\xi\widetilde{h}$.

5 Application to Robot Kinematics

Once we know how to make points, vectors, lines, screws and planes displacements, we can use them to perform robot kinematics. Firstly, the end-effector information must be codified into the dual quaternion ξ. Examples will be illustrate some situations. In all cases, h is the general displacement operator, composed by the joints transformations, $\mathscr{B} = \{\mathbf{u}, \mathbf{v}, \mathbf{w}\}$ and $\mathscr{B}' = \{\mathbf{u}', \mathbf{v}', \mathbf{w}'\}$ are the end-effector orthogonal coordinate systems associated to Ox, Oy and Oz axis, before and after transformations, respectively all represented in global orthogonal coordinate system $\mathscr{G}$. The end-effector position is represented by vector $\mathbf{r}$.

Case 1—posture from three vectors: Let the vectors ${}^{x}V \leftarrow \mathbf{u}$, ${}^{z}V \leftarrow \mathbf{w}$ and ${}^{r}V \leftarrow \mathbf{r}$ in dual quaternions coordinates, namely, ${}^{x}V = 0 + \mathbf{0} + \varepsilon(0 + \mathbf{u})$, ${}^{z}V = 0 + \mathbf{0} + \varepsilon(0 + \mathbf{w})$ and ${}^{r}V = 0 + \mathbf{0} + \varepsilon(0 + \mathbf{r})$. Using the vector transformations we get

$$\mathbf{u}' \leftarrow {}^{x}V'(6:8), \quad \mathbf{w}' \leftarrow {}^{z}V'(6:8), \quad \mathbf{v}' = \mathbf{w}' \times \mathbf{u}', \quad \mathbf{r}' \leftarrow {}^{r}V'(6:8).$$

Case 2—posture from two lines: Let ${}^{x}L \leftarrow ({}^{x}\mathbf{l}, {}^{x}\mathbf{m})$ and ${}^{z}L \leftarrow ({}^{z}\mathbf{l}, {}^{z}\mathbf{m})$ two lines with directions $\mathbf{u}$, $\mathbf{w}$ and moment vectors ${}^{x}\mathbf{m}$, ${}^{z}\mathbf{m}$, respectively. Namely, ${}^{x}L = 0 + \mathbf{u} + \varepsilon(0 + {}^{x}\mathbf{m})$, ${}^{z}L = 0 + \mathbf{w} + \varepsilon(0 + {}^{z}\mathbf{m})$, the line transformations and line intersection [18] assure that

$$\mathbf{u}' \leftarrow {}^{x}L'(2:4), \ \ \mathbf{w}' \leftarrow {}^{z}L'(2:4), \ \ \mathbf{v}' = \mathbf{w}' \times \mathbf{u}', \ \ \mathbf{r}' = {}^{x}\mathbf{u}' \times {}^{x}\mathbf{m}' + \left[\left({}^{z}\mathbf{w}' \times {}^{z}\mathbf{m}'\right) \cdot {}^{x}\mathbf{u}'\right] {}^{x}\mathbf{u}',$$

where ${}^{x}\mathbf{m}' = {}^{x}L'(6:8)$ and ${}^{z}\mathbf{m}' = {}^{z}L'(6:8)$. Figure 2 (left) relates the robot kinematics from line displacements.

Case 3—posture from line, vector and plane: Let ${}^{x}L \leftarrow ({}^{x}\mathbf{l}, {}^{x}\mathbf{m}) = (\mathbf{u}, \mathbf{m})$ be a line with direction $\mathbf{u}$ and moment vector $\mathbf{m}$, ${}^{x}\Pi \leftarrow ({}^{x}\mathbf{n}, d) = (\mathbf{n}, d)$ be a plane with normal vector $\mathbf{u}$ and distance from origin d and ${}^{z}V \leftarrow \mathbf{w}$ a vector. Namely, ${}^{x}L = 0 + \mathbf{u} + \varepsilon(0 + \mathbf{m})$, ${}^{x}\Pi = 0 + \mathbf{n} + \varepsilon(-d + \mathbf{0})$ and ${}^{z}V = 0 + \mathbf{0} + \varepsilon(0 + \mathbf{w})$. The posture of end-effector is given by the transformations

$${}^{x}L' = h\,{}^{x}L\widetilde{h} = h\,{}^{x}Lh^{*}, \quad {}^{x}\Pi' = h\,{}^{x}\Pi\widetilde{h} = h\,{}^{x}\Pi\overline{h^{*}}, \quad {}^{z}V' = h\,{}^{z}V\widetilde{h} = h\,{}^{z}V\overline{h^{*}}.$$

Indeed, from these equations we have $\mathbf{u} = \mathbf{l} = \mathbf{n}$, then

$$\mathbf{u}' \leftarrow {}^{x}L'(2:4), \ \ \mathbf{w}' \leftarrow {}^{z}V'(6:8), \ \ \mathbf{v}' = \mathbf{w}' \times \mathbf{u}', \ \ \mathbf{r}' = \mathbf{l}'_0 + d'\mathbf{u}' = \mathbf{u}' \times \mathbf{m}' + d'\mathbf{u}',$$

where $d' = {}^{x}\Pi'(5)$, and $\mathbf{m}' = {}^{x}L'(6:8)$. The Fig. 2 (right) relates the robot kinematics using vector, line and plane displacements.

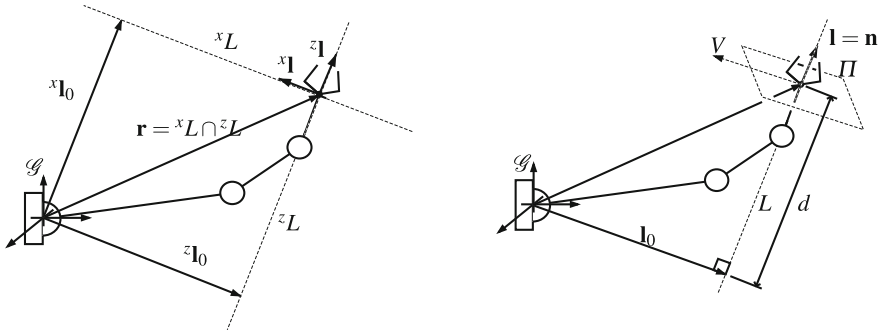


Fig. 2 Robot kinematics from line intersection (*left*) and plane (*right*) displacements in dual quaternions coordinate

6 Conclusion

In this chapter we presented a dual quaternions representation for points, vectors, lines, screws and planes, how we perform their displacements and some applications in positional kinematics. No Clifford algebra was used. All of it was developed from dual quaternions point of view, using no confuse Clifford generators.

On the bibliography review, dual quaternion transformations appears from principle of transference, but now it is not a sufficient trick anymore, once we have many kind of geometrical elements to handle with, not only dual vectors.

Quaternions and dual quaternions offer an alternatively algebra for modeling problems, therefore we can rethink those one which use rigid boby displacements in order to avoid some undesire effect. Three examples in robot positional kinematics enlighten the applicability of our dual quaternion approach. Unfortunately we do not have space to analyse a more challenging problems. That will be our future investigation.

References

1. Agrawal, O.P.: Hamilton operators and dual-number-quaternions in spatial kinematics. Mech. Mach. Theor. **22**(6), 569–575 (1987)
2. Bayro-Corrochano, E., Falcón, L.E.: Geometric algebra of points, lines, planes and spheres for computer vision and robotics. Robotica **23**(6), 1469–8668 (2005). doi: 10.1017/S0263574705001657
3. Christoph, M.H., Yang, w.: In: LI, Z., Sit, W. (eds.) Compliant motion constraints. Singapore, World Scientific (2003)
4. Dooley, J.R., McCarthy, J.M.: On the geometric analysis of optimum trajectories for cooperating robots using dual quaternion coordinates. In: IEEE Transactions on Robotics and Automation, pp. 1031–1036 (1993)
5. Funda, J., Paul, R.P.: A computational analysis os screw transformations in robotics. IEEE Trans. Robot. Autom. **6**(3), 382–388 (1990)

6. Lounesto, P.: Clifford Algebras and Spinors, 2nd edn. Cambridge University Press, Cambridge (2001)
7. Martfnez, J.M.R.: The principle of transference: history, statement and proof. Mech. Mach. Theor. **28**(1), 165–177 (1993)
8. Radavelli, L., Simoni, R., Pieri, E.R.D., Martins, D.: A comparative study of the kinematics of robot manipulators by Denavit- Hartenberg and dual quaternion. Mecnica Computacional XXXI, 2833–2848 (2012)
9. Radavelli, L.: Análise cinemática direta de robôs manipuladores via álgebra de clifford e quatérnios. Dissertação, UFSC (2013)
10. Sahul, S., Biswall, B.B., Subudhi, B.: A novel method for representing robot kinematics using quaternion theory. In: IEEE Sponsored Conference on Computational Intelligence, Control and Computer Vision in Robotics and Automation (2008)
11. Selig, J.M.: Clifford algebra of points, lines and planes. Robotica **18**, 545–546 (2000a)
12. Selig, J.M.: Geometric Fundamentals of Robotics. Springer, New York (2000b)
13. Shoham, M., Ben-Horin, P.: Application of grassmann-cayley algebra to geometrical interpretation of parallel robot singularities. Int. J. Rob. Res. **28**, 127–141 (2009)
14. Vince, J.: Geometric Algebra for Computer Graphics. Springer, London (2008)
15. Wang, J-Y., Liang, H-Z., Sun, Z-W., Wu, S-N., Zhang, S-H.: Relative motion coupled control based on dual quaternion. Aerosp. Sci. Technol. **25**(1), 102–113 (2013)
16. Wang, X., Han, D., Yu, C., Zheng, Z.: The geometric structure of unit dual quaternion with application in kinematic control. J. Math. Anal. Appl. **389**(2), 1352–1364 (2012)
17. Wang, X., Yu, C.: Unit dual quaternion-based feedback linearization tracking problem for attitude and position dynamics. Syst. Control Lett. **62**(3), 225–233 (2013). doi: http://dx.doi.org/10.1016/j.sysconle.2012.11.019
18. Woo, L., Freudenstein, F.: Application of line geometry to theoretical kinematics and the kinematic analysis of mechanical systems. Mechanisms **5**, 417–460 (1970)
19. Yang, A.T., Freudenstein, F.: Application of dual-number quaternions to the analysis of the spatial mechanism. AMSE Trans. J. Appl. Mech. **86**, 300–308 (1964)

Recovering Dual Euler Parameters From Feature-Based Representation of Motion

Daniel Condurache and Adrian Burlacu

Abstract The parameterization of a rigid-body motion can be done using multiple algebraic entities. A very important criterion when choosing a parameterization methods is the number of algebraic equations and variables. Recently, orthogonal dual tensors proved to be a complete tool for computing rigid body displacement and motion parameters. The present research is focused on developing new methods for recovering kinematic data when the state of features attached to a body during a rigid displacement is available. The proof of concept is sustained by computational solutions both for the construction of orthogonal dual tensors and for the recovery algorithms of the dual quaternion and the dual Rodrigues vector.

Keywords Dual rigid basis · Dual quaternion · Dual Rodrigues vector · Rigid motion

1 Introduction

Different techniques for rigid body motion representation can be obtained if dual numbers, dual vectors or dual matrices [1, 12] are combined with elements of screw theory [10]. First reported as biquaternions, dual quaternions are associated with linear algebra methods to represent a general displacement of a rigid body and to model the group of rigid body displacements [14]. A series of interesting applications

D. Condurache
Department of Theoretical Mechanics, Gheorghe Asachi Technical University of Iasi, Iasi, Romania
e-mail: daniel.condurache@gmail.com

A. Burlacu (✉)
Department of Automatic Control and Applied Informatics, Gheorghe Asachi Technical University of Iasi, Iasi, Romania
e-mail: aburlacu@ac.tuiasi.ro

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_31,

based on dual quaternions were considered and multiple algorithms were developed for: rigid motion analysis and planning [13]; hand-eye calibration [9]; kinematics equations evaluation of manipulator robots [11, 14]; relative orbital motion [6, 17].

The desire to create a complete framework for parametrization of rotation and rigid motion lead to closer inspection of linear invariants of the dual rotation matrix and the dual Euler-Rodriques parameters of rigid motion [1, 16]. A general screw displacement, that is a finite displacement, can be represented by an analogous to Rodrigues original formula but in a dual form. In the last years a new entity was considered for parameterization of rotation and motion: tensors. Tensor analysis expresses the invariance of the laws of physics with respect to the change of basis and change of frame operations [3, 4, 7]. In [7], we have presented an algebraic method based on orthogonal dual tensors for spatial rigid body motion parametrization. In this paper we present how rigid basis [5] of dual vectors can be used to provide direct computational solutions for the dual Euler-Rodrigues parameters, the dual quaternion and the dual Rodrigues vector when information regarding the state of different type of features attached to a rigid body are available. Our study shows how the dual rigid bases can be used to obtain free of coordinates computational methods when a rigid body is characterized by points and lines. This approach does not need the actual structure of the dual orthogonal tensor to compute the Euler parameters, fact which represents the novelty of our research.

2 Dual Tensor Based Representation of Motion

Let the set of real **dual numbers** be denoted by $\underline{\mathbb{R}} = \{\underline{a} = a + \varepsilon a_0 |\ a, a_0 \in \mathbb{R},\ \varepsilon^2 = 0\}$, where $a = Re(\underline{a})$ is the real part and $a_0 = Du(\underline{a})$ the dual part. In the Euclidean space, the linear space of free vectors with dimension 3 will be denoted by V_3. The ensemble of **dual vectors** is defined as $\underline{V_3} = \{\underline{\mathbf{a}} = \mathbf{a} + \varepsilon \mathbf{a}_0; \mathbf{a}, \mathbf{a}_0 \in V_3,\ \varepsilon^2 = 0\}$, where $\mathbf{a} = Re(\underline{\mathbf{a}})$ is the real part and $\mathbf{a}_0 = Du(\underline{\mathbf{a}})$ the dual part. For any three dual vectors $\underline{\mathbf{a}}, \underline{\mathbf{b}}, \underline{\mathbf{c}}$ the following notations will be used for the basic products: $\underline{\mathbf{a}} \cdot \underline{\mathbf{b}}$—scalar product, $\underline{\mathbf{a}} \times \underline{\mathbf{b}}$—cross product, $< \underline{\mathbf{a}}, \underline{\mathbf{b}}, \underline{\mathbf{c}} >$—scalar triple product. The magnitude of $\underline{\mathbf{a}}$, denoted by $|\underline{\mathbf{a}}|$, is the dual number computed from

$$|\underline{\mathbf{a}}| = \begin{cases} \|\mathbf{a}\| + \varepsilon \dfrac{\mathbf{a}_0 \cdot \mathbf{a}}{||\mathbf{a}||}, & Re(\underline{\mathbf{a}}) \neq \mathbf{0} \\ \varepsilon ||\mathbf{a}_0||, & Re(\underline{\mathbf{a}}) = \mathbf{0} \end{cases}, \tag{1}$$

where $||.||$ is the Euclidean norm. If $|\underline{\mathbf{a}}| = 1$ then $\underline{\mathbf{a}}$ is called unit dual vector.

For any $\underline{\mathbf{a}} \in \underline{V_3}$ a dual number $\underline{\lambda} \in \underline{\mathbb{R}}$ and a unit dual vector $\underline{\mathbf{u}} \in \underline{V_3}$ exist in order to have $\underline{\mathbf{a}} = \underline{\lambda}\, \underline{\mathbf{u}}$. Consider $||.||$ to denote the Euclidean norm then

$$\pm \underline{\lambda} = |\underline{\mathbf{a}}|; \quad \pm \underline{\mathbf{u}} = \begin{cases} \dfrac{\mathbf{a}}{||\mathbf{a}||} + \varepsilon \dfrac{\mathbf{a} \times (\mathbf{a}_0 \times \mathbf{a})}{||\mathbf{a}||^3} & Re(\underline{\mathbf{a}}) \neq \mathbf{0} \\ \dfrac{\mathbf{a}_0}{||\mathbf{a}_0||} + \varepsilon \mathbf{v} \times \dfrac{\mathbf{a}_0}{||\mathbf{a}_0||}, \ \forall \mathbf{v} \in V_3 & Re(\underline{\mathbf{a}}) = \mathbf{0} \end{cases}. \tag{2}$$

The previous result allows to geometrically describe any dual vector from the 3D Euclidean space, as detailed in [7].

An $\mathbb{R}$-linear application of $\underline{V}_3$ into $\underline{V}_3$ is called an Euclidean **dual tensor**:

$$\boldsymbol{T}(\underline{\lambda}_1\underline{\mathbf{v}}_1 + \underline{\lambda}_2\underline{\mathbf{v}}_2) = \underline{\lambda}_1\boldsymbol{T}(\underline{\mathbf{v}}_1) + \underline{\lambda}_2\boldsymbol{T}(\underline{\mathbf{v}}_2), \quad \forall\underline{\lambda}_1, \underline{\lambda}_2 \in \underline{\mathbb{R}}, \forall\underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2 \in \underline{\mathbf{V}}_3. \tag{3}$$

Let $\mathbf{L}(\underline{V}_3, \underline{V}_3)$ be the set of dual tensors, then any $\boldsymbol{T} \in \mathbf{L}(\underline{V}_3, \underline{V}_3)$ can be decomposed as $\boldsymbol{T} = T + \varepsilon T_0$, with $T, T_0 \in \mathbf{L}(V_3, V_3)$. Also, the dual transposed tensor denoted by $\boldsymbol{T}^{\mathrm{T}}$ is defined by

$$\underline{\mathbf{v}}_1 \cdot (\boldsymbol{T}\underline{\mathbf{v}}_2) = \underline{\mathbf{v}}_2 \cdot (\boldsymbol{T}^{\mathrm{T}}\underline{\mathbf{v}}_1), \quad \forall\underline{\mathbf{v}}_1, \underline{\mathbf{v}}_2 \in \underline{V}_3, \tag{4}$$

An important class of invariants that is used to describe the dual tensor are called **linear invariants** and are denoted by $\mathrm{vect}\boldsymbol{T} = \mathrm{vect}\frac{1}{2}[\boldsymbol{T} - \boldsymbol{T}^{\mathrm{T}}]$ and $\mathrm{trace}\boldsymbol{T}$ [2]. For an arbitrary dual tensor $\boldsymbol{T}$ the following entities can be computed

$$\mathrm{sym}\boldsymbol{T} = \frac{1}{2}[\boldsymbol{T} + \boldsymbol{T}^{\mathrm{T}}], \quad \mathrm{skew}\boldsymbol{T} = \frac{1}{2}[\boldsymbol{T} - \boldsymbol{T}^{\mathrm{T}}], \tag{5}$$

where "sym" is the symmetric part of the dual tensor and "skew" is its skew-symmetric part. The dual tensors set is an $\mathbb{R}$-module of rank 9, while the skew-symmetric dual tensors is structured as a rank 3 $\underline{\mathbb{R}}$ sub-module of $\mathbf{L}(\underline{V}_3, \underline{V}_3)$ and is isomorph with $\underline{V}_3$. For any skew-symmetric dual tensor $\boldsymbol{A}$, a dual vector $\underline{\mathbf{a}} \in \underline{V}_3$ exists so that:

$$\boldsymbol{A}\underline{\mathbf{v}} = \underline{\mathbf{a}} \times \underline{\mathbf{v}}, \forall\underline{\mathbf{v}} \in \underline{V}_3. \tag{6}$$

From now on $\widetilde{\underline{\mathbf{a}}}$ will denote the skew-symmetric dual tensor attached to a dual vector $\underline{\mathbf{a}} = \mathrm{vect}\,\widetilde{\underline{\mathbf{a}}}$.

In order to completely describe our solutions, we need to consider the following notations

$$\begin{gathered}\underline{S\mathbb{O}}_3 = \{\boldsymbol{R} \in \mathbf{L}(\underline{\mathbf{V}}_3, \underline{\mathbf{V}}_3) \mid \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}, \ \det\boldsymbol{R} = 1\}; \\ \{f : \mathbb{R} \to V_3\} = V_3^{\mathbb{R}}, \{f : \mathbb{R} \to S\mathbb{O}_3\} = S\mathbb{O}_3^{\mathbb{R}}, \{f : \mathbb{R} \to \underline{S\mathbb{O}}_3\} = \underline{S\mathbb{O}}_3^{\mathbb{R}},\end{gathered} \tag{7}$$

where $S\mathbb{O}_3$ is the set of real special orthogonal tensors and $f = f(t)$ (t being the time variable). The internal structure of any dual tensor function $\boldsymbol{R} \in \underline{S\mathbb{O}}_3^{\mathbb{R}}$ is illustrated by the following three results [7]:

Remark 1 For any $\boldsymbol{R} \in \underline{S\mathbb{O}}_3^{\mathbb{R}}$, an unique decomposition is viable

$$\boldsymbol{R} = Q + \varepsilon\widetilde{\boldsymbol{\rho}}Q, \tag{8}$$

where $Q = Q(t) \in S\mathbb{O}_3^{\mathbb{R}}$ and $\boldsymbol{\rho} = \boldsymbol{\rho}(t) \in V_3^{\mathbb{R}}$.

Remark 2 Based on the construction of $\underline{S\mathbb{O}_3}$ and the multiplication of dual tensors, a direct conclusion is the Lie group structure of (7). This Lie group globally parameterizes all rigid motions. Thus, a rigid body motion [7, 16] can be modeled by (8).

Remark 3 For any $\boldsymbol{R} \in \underline{S\mathbb{O}_3}$ a dual number $\underline{\alpha} = \alpha + \varepsilon d$ and a dual unit vector $\underline{\mathbf{u}} = \mathbf{u} + \varepsilon \mathbf{u}_0$ exists in order to have the following expression:

$$\boldsymbol{R} = \boldsymbol{I} + \sin\underline{\alpha}\,\underline{\tilde{\mathbf{u}}} + (1 - \cos\underline{\alpha})\underline{\tilde{\mathbf{u}}}^2 = \exp(\underline{\alpha}\,\underline{\tilde{\mathbf{u}}}). \tag{9}$$

The parameters $\underline{\alpha}$ and $\underline{\mathbf{u}}$ are called the **natural invariants** of $\boldsymbol{R}$ and can be recovered from the linear invariants [2] using (9):

$$\underline{\mathbf{u}}\sin\underline{\alpha} = \text{vect}\boldsymbol{R}, \quad \cos\underline{\alpha} = \frac{1}{2}[\text{trace}\boldsymbol{R} - 1]. \tag{10}$$

The unit dual vector $\underline{\mathbf{u}}$ is the Plucker representation of the Mozzi-Chalses axis, while the dual angle $\underline{\alpha} = \alpha + \varepsilon d$ embeds the rotation angle α and the translation d. In literature there are presented different methods [7, 8] which can be used to recover $\underline{\mathbf{u}}$ and $\underline{\alpha}$ from (10).

3 Dual Euler-Rodrigues, Dual Quaternions and Dual Rodrigues Vector Recovery Solutions

3.1 Dual Quaternions

A dual quaternion can be defined as an associated pair of a dual scalar quantity and a free dual vector:

$$\underline{\widehat{\mathbf{q}}} = (\underline{q}, \underline{\mathbf{q}}), \quad \underline{q} \in \underline{\mathbb{R}}, \underline{\mathbf{q}} \in \underline{V_3}. \tag{11}$$

The set of dual quaternions will be denoted $\underline{\mathbf{Q}}$ and is organized as a $\underline{\mathbb{R}}$-module of rank 4, if dual quaternion addition and multiplication with dual numbers are considered. The product of two dual quaternions $\underline{\widehat{\mathbf{q}}}_1 = (\underline{q}_1, \underline{\mathbf{q}}_1)$ and $\underline{\widehat{\mathbf{q}}}_2 = (\underline{q}_2, \underline{\mathbf{q}}_2)$ is defined by

$$\underline{\widehat{\mathbf{q}}}_1\underline{\widehat{\mathbf{q}}}_2 = (\underline{q}_1 \cdot \underline{q}_2 - \underline{\mathbf{q}}_1 \cdot \underline{\mathbf{q}}_2, \underline{q}_1\underline{\mathbf{q}}_2 + \underline{q}_2\underline{\mathbf{q}}_1 + \underline{\mathbf{q}}_1 \times \underline{\mathbf{q}}_2). \tag{12}$$

Taking into account the above properties results that the $\underline{\mathbb{R}}$-module $\underline{\mathbf{Q}}$ becomes an associative, non-commutative linear dual algebra of order 4 over the ring of dual numbers . For any dual quaternion defined by (11) the followings can be computed: the norm denoted by $|\underline{\widehat{\mathbf{q}}}|$ and the conjugate denoted by $\underline{\widehat{\mathbf{q}}}^*$. Regarded solely as a free $\underline{\mathbb{R}}$-module, $\underline{\mathbf{Q}}$ contains two remarkable sub-modules: $\underline{\mathbf{Q}}_{\underline{\mathbb{R}}}$ and $\underline{\mathbf{Q}}_{\underline{V_3}}$. The first one composed from pairs $(\underline{q}, \underline{\mathbf{0}}), \underline{q} \in \underline{\mathbb{R}}$, isomorphic with $\underline{\mathbb{R}}$, and the second one, containing the pairs $(\underline{0}, \underline{\mathbf{q}}), \underline{\mathbf{q}} \in \underline{V_3}$, isomorphic with $\underline{V_3}$. Also, any dual quaternion

can be written as $\widehat{\underline{\mathbf{q}}} = \underline{q} + \underline{\mathbf{q}}$, where $\underline{q} = (q, \mathbf{0})$ and $\underline{\mathbf{q}} = (0, \mathbf{q})$, or $\widehat{\underline{\mathbf{q}}} = \widehat{\mathbf{q}} + \varepsilon\widehat{\mathbf{q}}_0$, where $\widehat{\mathbf{q}}, \widehat{\mathbf{q}}_0$ are real quaternions.

Let $\mathbb{U}$ denote the set of unit quaternions and $\underline{\mathbb{U}}$ denotes the set dual unit quaternions. For any $\widehat{\underline{\mathbf{q}}} \in \underline{\mathbb{U}}$, the following representation is valid [15]

$$\widehat{\underline{\mathbf{q}}} = (1 + \varepsilon\frac{1}{2}\boldsymbol{\rho})\widehat{\mathbf{q}}\,, \tag{13}$$

where $\boldsymbol{\rho} \in V_3$ and $\widehat{\mathbf{q}} \in \mathbb{U}$. This representation is the quaternionic counterpart to (8) and allows that a dual number $\underline{\alpha}$ and a dual vector $\underline{\mathbf{u}}$ exist in order to have

$$\widehat{\underline{\mathbf{q}}} = \cos\frac{\underline{\alpha}}{2} + \underline{\mathbf{u}}\sin\frac{\underline{\alpha}}{2}, \tag{14}$$

which gives a quaternionic counterpart to (9). For $\underline{\alpha} = \alpha + \varepsilon d$ and $\underline{\mathbf{u}} = \mathbf{u} + \varepsilon\mathbf{u}_0$, the pair $(\alpha, \mathbf{u})$ is recovered from $\widehat{\underline{\mathbf{q}}} = \cos\frac{\alpha}{2} + \mathbf{u}\sin\frac{\alpha}{2}$ and $d = \boldsymbol{\rho}\cdot\mathbf{u}$, $\mathbf{u}_0 = \frac{1}{2}\boldsymbol{\rho}\times\mathbf{u} + \frac{1}{2}\cot\frac{\alpha}{2}\mathbf{u}\times(\boldsymbol{\rho}\times\mathbf{u})$ if $\alpha \neq 0$ or $\mathbf{u}_0 = \frac{1}{2}\boldsymbol{\rho}\times\mathbf{u}$ if $\alpha = 0$.

Remark 4 The mapping $\exp : \underline{V}_3 \to \underline{\mathbb{U}}$, $\widehat{\underline{\mathbf{q}}} = \exp\frac{\boldsymbol{\varphi}}{2}$, is well defined and surjective.

Proof Consider $\underline{\mathbf{v}}$ a dual vector from $\underline{V}_3$. Based on (2), $\underline{\mathbf{v}}$ can be structured as $\underline{\mathbf{v}} = \underline{\lambda}\,\underline{\mathbf{u}}$. Taking into account that $\underline{\mathbf{u}}$ is a unitary dual vector and the definition of the quaternionic product (12) results that $\widehat{\underline{\mathbf{u}}}\widehat{\underline{\mathbf{u}}} = -1$. This proves the correct construction of the mapping $\exp(\underline{\lambda}\,\underline{\mathbf{u}}) = \cos\underline{\lambda} + \underline{\mathbf{u}}\sin\underline{\lambda}$. Considering $\underline{\lambda} = \frac{\underline{\alpha}}{2}$ together with (14), the proof of surjectivity is complete.

Remark 5 Based on the construction of $\underline{\mathbb{U}}$ and the multiplication of dual quaternions, a direct conclusion is its Lie group structure ($\underline{V}_3$ being the associated Lie algebra, where the cross-product between dual vectors is the internal mapping), which can be used to global parameterize all rigid motions.

3.2 Orthogonal Dual Tensors Construction

Given two dual vectors $\underline{\mathbf{a}}$ and $\underline{\mathbf{b}} \in \underline{V}_3$, $\underline{\mathbf{a}} \otimes \underline{\mathbf{b}}$ denotes ***dyadic product***, which is a dual tensor defined by:

$$\underline{\mathbf{a}} \otimes \underline{\mathbf{b}} : \underline{V}_3 \to \underline{V}_3, \quad (\underline{\mathbf{a}} \otimes \underline{\mathbf{b}})\underline{\mathbf{v}} = (\underline{\mathbf{v}} \cdot \underline{\mathbf{b}})\underline{\mathbf{a}}, \ \forall\underline{\mathbf{v}} \in \underline{V}_3. \tag{15}$$

An important property of (15) is that $(\underline{\mathbf{a}} \otimes \underline{\mathbf{b}})(\underline{\mathbf{c}} \otimes \underline{\mathbf{d}}) = (\underline{\mathbf{b}} \cdot \underline{\mathbf{c}})\underline{\mathbf{a}} \otimes \underline{\mathbf{d}}$. If the dual tensor defined by (15) is analyzed, the following results emerge: $(\underline{\mathbf{a}} \otimes \underline{\mathbf{b}})^{\mathrm{T}} = \underline{\mathbf{b}} \otimes \underline{\mathbf{a}}$, $\mathrm{vect}(\underline{\mathbf{a}} \otimes \underline{\mathbf{b}}) = \frac{1}{2}(\underline{\mathbf{b}} \times \underline{\mathbf{a}})$ and $\mathrm{trace}(\underline{\mathbf{a}} \otimes \underline{\mathbf{b}}) = \underline{\mathbf{a}} \cdot \underline{\mathbf{b}}$. From this point on we uncover how the dyadic product can be used to construct a dual tensor.

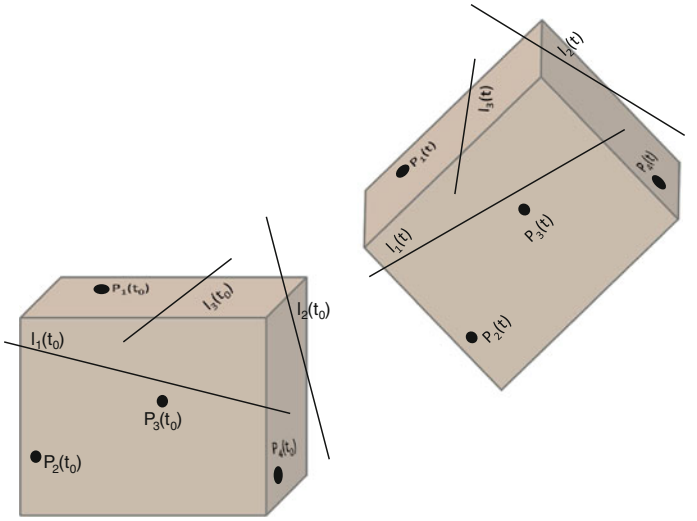


Fig. 1 Rigid body features, points and lines, represented at two time stamps

A basis $\mathbf{B} = \{\underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3\}$ from $\underline{V}_3^{\mathbb{R}}$, which has the values $\underline{\mathbf{e}}_{0i} = \underline{\mathbf{e}}_i(t_0)$ at initial time t_0 , is named dual rigid basis if:

$$\begin{cases} \underline{\mathbf{e}}_i \cdot \underline{\mathbf{e}}_j = \underline{\mathbf{e}}_{0i} \cdot \underline{\mathbf{e}}_{0j}, \ i, j = \overline{1,3}, \\ < \underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3 > = < \underline{\mathbf{e}}_{01}, \underline{\mathbf{e}}_{02}, \underline{\mathbf{e}}_{03} > \end{cases}. \tag{16}$$

Remark 6 For a dual basis $\mathbf{B} = \{\underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3\}$, the set $\mathbf{B}^* = \{\underline{\mathbf{e}}^1, \underline{\mathbf{e}}^2, \underline{\mathbf{e}}^3\}$ represents its reciprocal

$$\underline{\mathbf{e}}^1 = \frac{\underline{\mathbf{e}}_2 \times \underline{\mathbf{e}}_3}{< \underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3 >}, \ \underline{\mathbf{e}}^2 = \frac{\underline{\mathbf{e}}_3 \times \underline{\mathbf{e}}_1}{< \underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3 >}, \ \underline{\mathbf{e}}^3 = \frac{\underline{\mathbf{e}}_1 \times \underline{\mathbf{e}}_2}{< \underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3 >}. \tag{17}$$

The dual vectors $\left\{\underline{\mathbf{e}}^1, \underline{\mathbf{e}}^2, \underline{\mathbf{e}}^3\right\}$ are uniquely determined using the conditions $\underline{\mathbf{e}}_i \cdot \underline{\mathbf{e}}^j = \delta_i^j, i, j = \overline{1,3}$, δ_i^j being the Kronecker symbol. If $\mathbf{B} = \{\underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3\}$ is a dual rigid basis then $\mathbf{B}^* = \{\underline{\mathbf{e}}^1, \underline{\mathbf{e}}^2, \underline{\mathbf{e}}^3\}$, its reciprocal dual basis, is also rigid.

Using points and lines as rigid body features (as shown in Fig. 1), dual rigid bases can be constructed using the following five combinations [7]:

1. *Three non-coplanar directed lines*
   Consider that three non-coplanar lines $\mathbf{l}_i(t_0), i = \overline{1,3}$ are part of a rigid body. These lines can be characterized by normalized Plucker coordinates, which are staked in dual vectors $\underline{\mathbf{u}}_i(t_0) = \mathbf{m}_i(t_0) + \varepsilon \mathbf{n}_i(t_0), i = \overline{1,3}$. For each line the time t correspondent is denoted $\mathbf{l}_i(t), i = \overline{1,3}$ while the attached dual vectors are $\underline{\mathbf{u}}_i(t) = \mathbf{m}_i(t) + \varepsilon \mathbf{n}_i(t), i = \overline{1,3}$. In this case, the dual rigid bases are $\{\underline{\mathbf{e}}_{01}, \underline{\mathbf{e}}_{02}, \underline{\mathbf{e}}_{03}\} = \{\underline{\mathbf{u}}_1(t_0), \underline{\mathbf{u}}_2(t_0), \underline{\mathbf{u}}_3(t_0)\}$ and $\{\underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3\} = \{\underline{\mathbf{u}}_1(t), \underline{\mathbf{u}}_2(t), \underline{\mathbf{u}}_3(t)\}$.

2. *Two non-parallel directed lines*
   If two lines are available, then the non-parallel condition must be fulfilled. Let $\mathbf{l}_1(t_0)$ and $\mathbf{l}_2(t_0)$ be the two lines that describe the rigid body at time $t = t_0$. The rigid basis $\{\underline{\mathbf{e}}_{01}, \underline{\mathbf{e}}_{02}, \underline{\mathbf{e}}_{03}\}$ is computed as $\{\underline{\mathbf{u}}_1(t_0), \underline{\mathbf{u}}_2(t_0), \underline{\mathbf{u}}_1(t_0) \times \underline{\mathbf{u}}_2(t_0)\}$ The same method is applied at any time t when, for example, $\mathbf{l}_1(t)$ and $\mathbf{l}_2(t)$ are valid.
3. *Four non-planar points*
   Let $P_i(t_0), i = \overline{1,4}$ be four non-planar points attached to a rigid body. The dual rigid basis $\{\underline{\mathbf{e}}_{01}, \underline{\mathbf{e}}_{02}, \underline{\mathbf{e}}_{03}\}$ is computed from

$$\begin{aligned} \underline{\mathbf{e}}_{01} &= \rho_{02} - \rho_{01} + \varepsilon\rho_{01} \times \rho_{02} \\ \underline{\mathbf{e}}_{02} &= \rho_{03} - \rho_{01} + \varepsilon\rho_{01} \times \rho_{03} \\ \underline{\mathbf{e}}_{03} &= \rho_{04} - \rho_{01} + \varepsilon\rho_{01} \times \rho_{04} \end{aligned} \tag{18}$$

   where $\rho_{0i},\ i = \overline{1,4}$ is the position vector of $P_i(t_0)$. Using a similar procedure for $P_i(t), i = \overline{1,4}$ the dual rigid basis $\{\underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3\}$ is obtained.
4. *Three non-collinear points*
   In case of non-availability of one point from the four points set, if the remaining three points are non-collinear (e.g $P_1(t_0)$, $P_2(t_0)$, $P_3(t_0)$ are known) then $\underline{\mathbf{e}}_{01}$ and $\underline{\mathbf{e}}_{02}$ are computed using (18). For $\underline{\mathbf{e}}_{03}$ the following equation is used:

$$\underline{\mathbf{e}}_{03} = \underline{\mathbf{e}}_{01} \times \underline{\mathbf{e}}_{02}. \tag{19}$$

   The same approach is valid for $\{\underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3\}$ generated by $\{P_1(t), P_2(t), P_3(t)\}$.
5. *One point and one directed line*
   Let the $P_1(t_0)$ and $\mathbf{l}_1(t_0)$ be the available measurements on the rigid body. First, the constraint $P_1(t_0) \notin \mathbf{l}_1(t_0)$ must be fulfilled. If $\underline{\mathbf{u}} = \mathbf{u} + \varepsilon\mathbf{u}_0$ is the unit dual vector of $\mathbf{l}_1(t_0)$ then $\underline{\mathbf{e}}_{01} = \underline{\mathbf{u}}$, $\underline{\mathbf{e}}_{02} = \mathbf{u} \times \mathbf{w}_0 + \varepsilon\boldsymbol{\rho}_{P_1(t_0)} \times (\mathbf{u} \times \mathbf{w}_0)$ and $\underline{\mathbf{e}}_{03} = \underline{\mathbf{e}}_{01} \times \underline{\mathbf{e}}_{02}$. The vector $\mathbf{w}_0$ is defined as $\mathbf{w}_0 = \mathbf{u}_0 - \boldsymbol{\rho}_{P_1(t_0)} \times \mathbf{u}$. The same approach is used to compute $\{\underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3\}$ when $P_1(t)$ and $\mathbf{l}_1(t)$ are known.

Theorem 1 *If* $\mathbf{B} = \{\underline{\mathbf{e}}_1, \underline{\mathbf{e}}_2, \underline{\mathbf{e}}_3\}$ *is a dual rigid basis in* $\underline{V}_3^{\mathbb{R}}$ *and* $\underline{\mathbf{e}}_i$ *are continuous functions, then the dual tensor*

$$\boldsymbol{R} = \underline{\mathbf{e}}_i \otimes \underline{\mathbf{e}}_0^i \tag{20}$$

is proper orthogonal and uniquely defined by $\boldsymbol{R}(\underline{\mathbf{e}}_{0i}) = \underline{\mathbf{e}}_i,\ i = \overline{1,3}$.

Remark 7 Based on the linear invariants of (20) results:

$$\underline{\mathbf{u}} \sin\underline{\alpha} = \frac{1}{2}\underline{\mathbf{e}}_0^i \times \underline{\mathbf{e}}_i, \quad \cos\underline{\alpha} = \frac{1}{2}[\underline{\mathbf{e}}_0^i \cdot \underline{\mathbf{e}}_i - 1]. \tag{21}$$

Previous, the Einstein's rule for mute indexes summation was considered. The proofs for Theorem 1 and Remark 7 are presented in [7]. Next the computational solutions for both dual quaternions and dual Rodrigues vector are revealed.

3.3 Computational Solutions

Our solutions consider as inputs information about the features attached to a rigid body at different time stamps and generates the dual Euler-Rodrigues parameters, the dual quaternion and the dual Rodrigues vector which parameterizes the rigid displacement.

Theorem 2 *Let*

$$\underline{\gamma} = \cos\frac{\underline{\alpha}}{2}, \quad \underline{\boldsymbol{\gamma}} = \underline{\mathbf{u}}\sin\frac{\underline{\alpha}}{2}, \tag{22}$$

be the dual Euler-Rodrigues parameters. Using some properties for functions of dual number variable [16] *together with* (21) *results that if* $Re(\underline{\mathbf{e}}_0^i \cdot \underline{\mathbf{e}}_i) \neq -1$ *then*

$$\underline{\gamma} = \pm\frac{1}{2}\sqrt{1 + \underline{\mathbf{e}}_0^i \cdot \underline{\mathbf{e}}_i}, \quad \underline{\boldsymbol{\gamma}} = \frac{\pm\underline{\mathbf{e}}_0^i \times \underline{\mathbf{e}}_i}{2\sqrt{1 + \underline{\mathbf{e}}_0^i \cdot \underline{\mathbf{e}}_i}}. \tag{23}$$

Theorem 3 *Given the dual rigid bases* $\{\underline{\mathbf{e}}_1(t), \underline{\mathbf{e}}_2(t), \underline{\mathbf{e}}_3(t)\}$ *and* $\{\underline{\mathbf{e}}_{01}, \underline{\mathbf{e}}_{02}, \underline{\mathbf{e}}_{03}\}$, *as presented for any of the five combinations of features, the structure of the dual quaternion that parametrizes the rigid displacement is*

$$\pm\widehat{\underline{\mathbf{q}}} = \begin{cases} \dfrac{1 - \widehat{\underline{\mathbf{e}}}_i\widehat{\underline{\mathbf{e}}}_0^i}{|1 - \widehat{\underline{\mathbf{e}}}_i\widehat{\underline{\mathbf{e}}}_0^i|}; \ \textit{if } Re(\widehat{\underline{\mathbf{e}}}_i\widehat{\underline{\mathbf{e}}}_0^i) \neq 1 \\ \\ \left(1 + \varepsilon\dfrac{1}{2}\widehat{\boldsymbol{\rho}}\right)\widehat{\mathbf{u}}; \ \textit{if } Re(\widehat{\underline{\mathbf{e}}}_i\widehat{\underline{\mathbf{e}}}_0^i) = 1 \end{cases}, \tag{24}$$

where $\widehat{\underline{\mathbf{e}}}_i, \widehat{\underline{\mathbf{e}}}_0^i, \widehat{\boldsymbol{\rho}}, \widehat{\mathbf{u}} \in \underline{\mathbf{Q}}_{\underline{V}_3}$.

Proof For $Re(\widehat{\underline{\mathbf{e}}}_i\widehat{\underline{\mathbf{e}}}_0^i) \neq 1$ the solution is $\pm\widehat{\underline{\mathbf{q}}} = \underline{\gamma} + \underline{\boldsymbol{\gamma}}$, which combined with (23) leads to (24) . The proof for the case $Re(\widehat{\underline{\mathbf{e}}}_i\widehat{\underline{\mathbf{e}}}_0^i) = 1$, which embeds a rotation with an angle equal to π, is based on choosing $\mathbf{e}_0 + \mathbf{e}$ as the pair $\mathbf{e}_{0i} + \mathbf{e}_i$, $i = \overline{1,3}$ which has the maximum norm. The recovering procedure presented in [7] leads to

$$\mathbf{u} = \frac{\mathbf{e}_0 + \mathbf{e}}{||\mathbf{e}_0 + \mathbf{e}||}, \quad \boldsymbol{\rho} = \frac{1}{2}\mathbf{e}_i \times Q_0\mathbf{e}_0^i, \tag{25}$$

where $Q_0 = \mathbf{e}_i \otimes \mathbf{e}_0^{i*} + \mathbf{e}_i^* \otimes \mathbf{e}_0^i$, $\underline{\mathbf{e}}_i = \mathbf{e}_i + \varepsilon\mathbf{e}_i^*$, $\underline{\mathbf{e}}_0^i = \mathbf{e}_0^i + \varepsilon\mathbf{e}_0^{i*}$.

Remark 8 Using the second equation from (24), the solution for (23) when $Re(\underline{\mathbf{e}}_0^i \cdot \underline{\mathbf{e}}_i) = -1$ is:

$$\underline{\gamma} = -\varepsilon\frac{\boldsymbol{\rho} \cdot \mathbf{u}}{2}, \quad \underline{\boldsymbol{\gamma}} = \mathbf{u} + \varepsilon\frac{\boldsymbol{\rho} \times \mathbf{u}}{2}. \tag{26}$$

Theorem 4 *A rigid motion can also be described by the dual Rodrigues vector, which embeds the natural invariants of* $\boldsymbol{R}$ *in*

$$\underline{\mathbf{b}} = \tan \frac{\underline{\alpha}}{2} \underline{\mathbf{u}}. \tag{27}$$

Using (22) *and* (23)*, the dual Rodrigues vector can be recovered from*

$$\underline{\mathbf{b}} = \frac{1}{\underline{\gamma}} \underline{\boldsymbol{\gamma}} = \frac{\underline{\mathbf{e}}_0^i \times \underline{\mathbf{e}}_i}{1 + \underline{\mathbf{e}}_0^i \cdot \underline{\mathbf{e}}_i}, \tag{28}$$

when $Re(\underline{\mathbf{e}}_0^i \cdot \underline{\mathbf{e}}_i) \neq -1$.

3.4 Numerical Results

The solutions presented in the previous subsection can be easily transposed into simulation algorithms. An implementation algorithm start by acknowledging the type of features combination chose for the description of the rigid body. The next step, computation of the dual rigid bases and their reciprocals, is accomplished using the solutions presented in Sect. 3.2. Once this step is finalized, Eqs. (23), (24) and (28) are employed to recover the dual Euler-Rodrigues parameters, the dual quaternion and the dual Rodrigues vector which parameterizes the rigid displacement.

Inspired by [13], lets consider a rigid body to be described at time t_0 by three points: $P_{01} = [1\ 0\ 0]^{\mathrm{T}}$, $P_{02} = [1\ 2\ 0]^{\mathrm{T}}$, $P_{03} = [1\ 2\ 1]^{\mathrm{T}}$, while at time t_1 the same configuration of points has the following data $P_1 = [2\ 3\ 1]^{\mathrm{T}}$, $P_2 = [0\ 3\ 1]^{\mathrm{T}}$, $P_3 = [0\ 3\ 2]^{\mathrm{T}}$. Without computing the dual tensor, our algorithm first constructs the dual rigid bases

$$\begin{array}{ll} \underline{\mathbf{e}}_{01} = \left[0\ 2\ 0\right]^{\mathrm{T}} + \varepsilon[0\ \ 0\ \ 2]^{\mathrm{T}}; & \underline{\mathbf{e}}_1 = \left[-2\ 0\ 0\right]^{\mathrm{T}} + \varepsilon[0\ -2\ 6]^{\mathrm{T}} \\ \underline{\mathbf{e}}_{02} = \left[0\ 2\ 1\right]^{\mathrm{T}} + \varepsilon[0\ -1\ 2]^{\mathrm{T}}; & \underline{\mathbf{e}}_2 = \left[-2\ 0\ 1\right]^{\mathrm{T}} + \varepsilon[3\ -4\ 6]^{\mathrm{T}}, \\ \underline{\mathbf{e}}_{03} = \left[2\ 0\ 0\right]^{\mathrm{T}} + \varepsilon[0\ \ 0\ \ 0]^{\mathrm{T}}; & \underline{\mathbf{e}}_3 = \left[0\ \ 2\ \ 0\right]^{\mathrm{T}} + \varepsilon[-2\ 0\ 4]^{\mathrm{T}} \end{array} \tag{29}$$

the reciprocal

$$\begin{array}{l} \underline{\mathbf{e}}_0^1 = \left[\ 0\ \ 0.5\ \ -1\right]^{\mathrm{T}} + \varepsilon[0\ \ \ 1\ \ 0.5]^{\mathrm{T}} \\ \underline{\mathbf{e}}_0^2 = \left[\ 0\ \ \ 0\ \ \ \ 1\right]^{\mathrm{T}} + \varepsilon[0\ -1\ \ \ 0]^{\mathrm{T}}. \\ \underline{\mathbf{e}}_0^3 = \left[0.5\ 0\ \ \ 0\right]^{\mathrm{T}} + \varepsilon[0\ \ \ 0\ \ \ 0]^{\mathrm{T}} \end{array} \tag{30}$$

After validating that $Re(\underline{\mathbf{e}}_0^i \cdot \underline{\mathbf{e}}_i) \neq -1$, the algorithm employs (23), (24) and (28) to recover the dual Euler-Rodrigues parameters, the dual quaternion and the dual Rodrigues vector:

$$\pm\underline{\gamma} = 0.70711 - \varepsilon 0.35355, \quad \pm\underline{\boldsymbol{\gamma}} = [0 \;\; 0 \;\; 0.7071]^{\mathrm{T}} + \varepsilon \left[1.4142 \;\; 0 \;\; 0.3536\right]^{\mathrm{T}}, \tag{31}$$

$$\pm\underline{\widehat{\mathbf{q}}} = [0.70711 \;\; 0 \;\; 0 \;\; 0.7071]^{\mathrm{T}} + \varepsilon \left[-0.35355 \;\; 1.4142 \;\; 0 \;\; 0.3536\right]^{\mathrm{T}}, \tag{32}$$

$$\underline{\mathbf{b}} = [0 \;\; 0 \;\; 1]^{\mathrm{T}} + \varepsilon [2 \;\; 0 \;\; 1]^{\mathrm{T}}. \tag{33}$$

This is a first step in proving our concept, future applications will be developed and will make use of direct measurements from different type of sensors.

4 Conclusions

The research presented in this paper is focused on developing new solutions to recover dual Euler parameters from feature based representation of rigid motion. Dual rigid bases can be constructed using different combinations of points and lines which represent rigid body features. The combination between a dual rigid basis and its reciprocal provides a natural computational instrument that can be used to solve many problems in the kinematics of rigid bodies. Our approach does not necessitate the computation of the dual tensor when the Euler parameters need to be recovered. This may imply an advantage over existing solutions (which usually iterate the computation of the dual matrix associated with the dual tensor) when point and lines features of the rigid body are available. The theoretical solutions have a form suitable for a direct implementation into numerical codes and should provide useful tools for the development of future applications.

References

1. Angeles, J.: The application of dual algebra to kinematic analysis. Comput. Methods Mech. Syst. **161**, 3–31 (1998)
2. Angeles, J.: Fundamentals of Robotic Mechanical Systems Theory, Methods, and Algorithms, 4th edn. Springer, New York (2014)
3. Bauchau, O., Choi, J.Y.: The vector parameterization of motion. Nonlinear Dyn. **33**, 165–188 (2003)
4. Bauchau, O., Li, L.: Tensorial parameterization of rotation and motion. J. Comput. Nonlinear Dyn. **6**, 031,007.1–031,007.8 (2011)
5. Condurache, D.: A New General Investigation of the Kinematics of Rigid Bodies. Polirom, Iasi-Romania (2010)
6. Condurache, D., Burlacu, A.: On six d.o.f relative orbital motion parametrization using rigid bases of dual vectors. In: Proceedings of AAS/AIAA Astrodynamics Specialist Conference, pp. 2293–2312. Hilton Head, South Carolina (2013)
7. Condurache, D., Burlacu, A.: Dual tensors based solutions for rigid body motion parameterization. Mech. Mach. Theory **74**, 390–412 (2014)
8. Dai, J.: Finite displacement screw operators with embedded Chasles motion. J. Mech. Rob. **4**(4), 1–9 (2012)

9. Daniilidis, K.: Hand-eye calibration using dual quaternions. Int. J. Robot. Res. **18**, 286–298 (1999)
10. Davidson, J., Hunt, K.: Robots and Screw Theory Applications of Kinematics and Statistics to Robotics. Oxford Press, Oxford (2004)
11. Han, D., Wei, Q., Lei, Z.: Kinematic control of free rigid bodies using dual quaternions. Int. J. Autom. Comput. **5**, 319–324 (2008)
12. Pennestri, E., Stefanelli, P.: Linear algebra and numerical algorithms using dual numbers. Multibody Syst. Dyn. **18**, 323–344 (2007)
13. Pennestri, E., Valentini, P.: Dual quaternions as a tool for rigid body motion analysis: A tutorial with an application to biomechanics. The Archive of Mechanical Engineering **LVII**, 187–205 (2010)
14. Perez-Gracia, A., McCarthy, J.M.: Dual quaternion synthesis of constrained robotic systems. J. Mech. Des. **126**, 425–435 (2004)
15. Perez-Gracia, A., McCarthy, J.M.: Kinematic synthesis of spatial serial chains using Clifford algebra exponentials. Proc. IMechE Part C J. Mech. Eng. Sci. **220**, 953–968 (2006)
16. Trainelli, L.: An attempt at a systematic framework for the parameterization of rotation and rigid motion. In: Proceedings of the 6th World Congress on Computational Mechanics (2004)
17. Wang, J., Liang, H., Sun, Z., Zhang, S., Liu, M.: Finite-time control for spacecraft formation with dual-number-based description. J. Guidance Control Dyn. **35**(3), 950–962 (2012)

Kinematics and Dynamics of a 3-RPSR Parallel Robot Used as a Pipe-Bending Machine

Mario Alberto Garcia-Murillo, Yukio Takeda, Eduardo Castillo-Casataneda, Daisuke Matsuura, Syohei Kawasumi and Jaime Gallardo-Alvarado

Abstract In this work the kinematics and dynamics of a six degrees of freedom 3-RPSR parallel manipulator, used as a pipe-bender machine, are solved. First the forward position analysis of the mechanism under study is solved by using mechanical constraints equations. Afterwards, the velocity and acceleration analysis are addressed by taking advantage of the properties of reciprocal screws. Then, the generalized forces of the mechanism are determined by combining the screw theory with the principle of virtual work. Finally, a numerical example is provided.

Keywords Parallel robot · Pipe-bender · Screw theory · Dynamics · Virtual work

1 Introduction

The dynamic analyses of a manipulator are a crucial stage in the process design and control. The calculation of the forces or torques of active joints allows selecting the more appropriate actuators and performing a suitable mechanical design to meet requirements such as stiffness of the mechanism [8]. The methods of Newton-Euler and the Lagrangian are the most used to solve the dynamic analysis of parallel mechanisms [1, 8]. Other authors use the formulations of Kapel, Routh, and Apell equations [3]. The Newton-Euler equations consider reactions of kinematic pairs, which do not to exclude the explicit formulation between movements and forces, creating a large number of equations. Moreover, the Lagrangian, Routh and Apell methods, involve

M. A. Garcia-Murillo (✉) · E. Castillo-Casataneda
CICATA-Qro, National Polytechnic Institute, Querétaro, Mexico
e-mail: mgarciam1114@alumno.ipn.mx

Y. Takeda · D. Matsuura · S. Kawasumi
Department of Mechanical Sciences and Engineering, Tokyo Institute of Technology, Tokyo, Japan

J. Gallardo-Alvarado
Department of Mechanical Engineering, Celaya Institute of Technology, Celaya, Mexico

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_32,

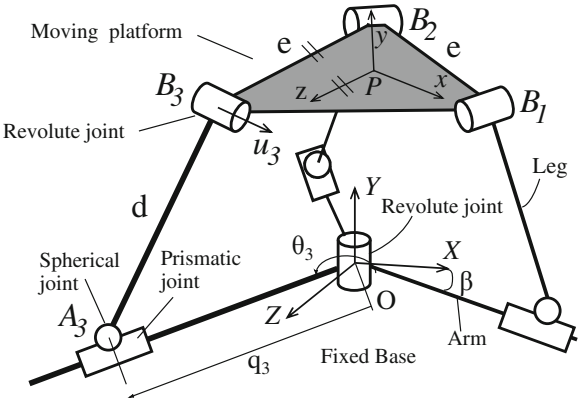


Fig. 1 3-RPSR parallel manipulator

systems of nonlinear differential equations that are computationally less efficient [3]. Other mechanical principles have been used to solve the dynamic analysis of robotic systems [1], such as the generalized momentum approach [4] and the Hamilton's principle [5]. One of the more popular approach to get the dynamic model of parallel robots is the use of the virtual work principle [1, 2, 8], which avoids the calculation of the internal reactions in the mechanism. In addition, the combination of this principle and the screw theory significantly simplifies the model.

Takeda [6, 7] proposed a 6 DOF spatial parallel manipulator as the movable-die drive mechanism, which can achieve a wide range of output orientation angles, for bending pipes into three- dimensional complex shapes. This chapter develops a model to determine the generalized forces through the principle of virtual work and the screw theory for that robot.

2 Kinematic Analysis of 3-RPSR Parallel Robot

The robot under study consists of a mobile platform and a fixed base connected by means of three RPSR kinematic chains (Fig. 1). The three actuated revolute joints share a common rotation axis, normal to the plane of the base. The arrangement of the kinematic chains is such that the axes of prismatic joints form a constant angle β with the plane of the base link. The centres of spherical joints are denoted as A_i and the connected locations of a leg and the moving platform are denoted by B_i, unless otherwise indicated, $i = 1, 2, 3$ (Fig. 1). The global coordinate system is O_{XYZ}, where point O is located at the intersection of the lines that pass through points A_i and are parallel to its corresponding axis of the prismatic joints. In addition, Y axis is collinear to the triple revolute joint. On the other hand, moving reference frame P_{xyz} is located at the center of triangle $B_1B_2B_3$, where x axis is a unit vector between points P and B_1; moreover, y axis is normal to plane $B_1B_2B_3$ and z is parallel to the passive revolute joint of chain 1 (see Fig. 1).

2.1 Displacement Analysis

It is well known that the pose of any rigid body can be specified by knowing the coordinates of three points belonging to it, then the pose of the moving platform, with respect to the fixed reference frame O_{XYZ}, may be determined by computing the coordinates of points B_1, B_2 and B_3 (see Fig. 1). Then we write equations that include these variables using mechanical constraints expressions.

The length of the limbs of the manipulator (Fig. 1) are restricted to

$$\|\boldsymbol{B}_i - \boldsymbol{A}_i\| = d^2, \quad \forall i = 1, 2, 3 \tag{1}$$

where $A_{X_i} = q_i \cos\beta \cos\theta_i$, $A_{Y_i} = -q_i \sin\beta$, $A_{Z_i} = -q_i \cos\beta \sin\theta_i$, and θ_i and q_i represent the displacements of active revolute and prismatic joints of chain i.

Three compatibility closure equations for the distance e may be expressed as:

$$\|\boldsymbol{B}_i - \boldsymbol{B}_j\| = e^2, \quad \forall i, j = 1, 2, 3; \quad i \neq j. \tag{2}$$

As is shown in Fig. 1b, any revolution axis is parallel to its opposite side of the triangle $B_1B_2B_3$ and perpendicular to its corresponding limb. These constraints can be expressed as:

$$\left(\boldsymbol{B}_i - \boldsymbol{B}_j\right) \cdot (\boldsymbol{B}_k - \boldsymbol{A}_k) = 0 \quad \forall i, j, k = 1, 2, 3; \quad i \neq j \neq k. \tag{3}$$

Equations (1–3) form a 9×9 equation system on B_{X_i}, B_{Y_i} and B_{Z_i}. It can be solved by using homotopy continuation method or Sylvester elimination method [8].

2.2 Velocity Analysis

The analysis of velocity and acceleration of the manipulator are solved by the screw theory. The screws are modeled as shown in Fig. 2, and referred to the frame O_{XYZ}. It is understood that the screw ${}^{a}\$_i^b$ models the movement of the body b with respect to a, both belonging to the i-th chain. The spherical pair is decomposed into three revolute pairs, $\left\{{}^{2}\$_i^3, {}^{3}\$_i^4, {}^{4}\$_i^5\right\}$, whose axes are perpendicular to each other. The direction of ${}^{2}\$_i^3$ is parallel with $\hat{u}_i$; screw ${}^{3}\$_i^4$ is collinear with vector $(\boldsymbol{B}_i - \boldsymbol{A}_i)$.

The velocity state of the mobile platform with respect to the fixed base, $\boldsymbol{V}_O$, can be expressed as a screw [2],

$$\boldsymbol{V}_O = \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{v}_O \end{bmatrix} = {}_0\omega_1^i \, {}^{0}\$_i^1 + {}_1\omega_2^i \, {}^{1}\$_i^2 + \ldots + {}_5\omega_6^i \, {}^{5}\$_i^6 = \mathbf{J}_i \boldsymbol{\Omega}_i, \tag{4}$$

where $\boldsymbol{\omega}$ and $\boldsymbol{v}_O$ are angular and translational velocities, respectively, of the mobile platform respect to the reference pole O; $\mathbf{J}_i = \left[{}^{0}\$_i^1 \; {}^{1}\$_i^2 \; {}^{2}\$_i^3 \; {}^{3}\$_i^4 \; {}^{4}\$_i^5 \; {}^{5}\$_i^6\right]$ and $\boldsymbol{\Omega}_i$ is a

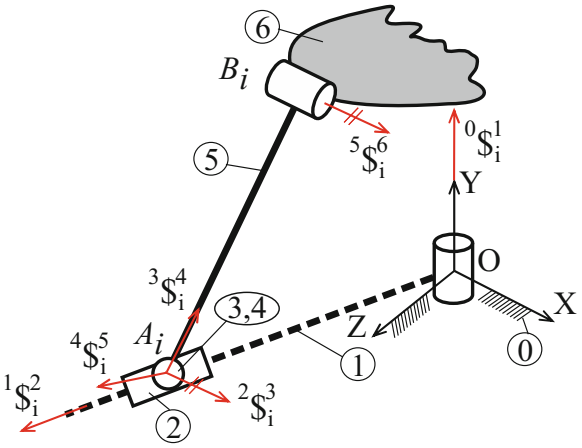


Fig. 2 Screws of the joints

vector that contains terms ${}_a\omega_b^i$, that represent the ratios of rotational or translational velocity of body b respect to a belonging to the i-th chain. Moreover, given the actuation scheme of the robot, ${}_0\omega_1^i = \dot{\theta}_i$ and ${}_1\omega_2^i = \dot{q}_i$.

With reference to Fig. 2, one can demonstrate that the screw ${}^2\$_i^3$ is reciprocal to all screws associated with revolute joints in the chain i, except to ${}^0\$_i^1$. Then, applying the Klein form, or reciprocal product, on both sides of Eq. (4), gives:

$$\left\{\boldsymbol{V}_O\,;\,{}^2\$_i^3\right\} = \dot{\theta}_i\left\{{}^0\$_i^1\,;\,{}^2\$_i^3\right\} + \dot{q}_i\left\{{}^1\$_i^2\,;\,{}^2\$_i^3\right\}. \tag{5}$$

In addition, applying the Klein form of screw ${}^3\$_i^4$ and Eq. (4):

$$\left\{\boldsymbol{V}_O\,;\,{}^3\$_i^4\right\} = \dot{\theta}_i\left\{{}^0\$_i^1\,;\,{}^3\$_i^4\right\} + \dot{q}_i\left\{{}^1\$_i^2\,;\,{}^3\$_i^4\right\}. \tag{6}$$

Grouping in matrix form Eqs. (5–6), the state of velocity can be calculated from the expression:

$$\mathbf{M}^{\mathrm{T}}\Delta\boldsymbol{V}_O = \dot{\mathbf{Q}}, \tag{7}$$

where $\mathbf{M} = \left[{}^2\$_1^3\ {}^2\$_2^3\ {}^2\$_3^3\ {}^3\$_1^4\ {}^3\$_2^4\ {}^3\$_3^4\right]$; $\Delta = \begin{bmatrix}\mathbf{0} & \mathbf{I}\\ \mathbf{I} & \mathbf{0}\end{bmatrix}$ is a polarity operator defined by 3×3 identity matrix, $\mathbf{I}$, and zero matrix $\mathbf{0}$, and finally,

$$\dot{\mathbf{Q}} = \begin{bmatrix}
\dot{\theta}_1\left\{{}^0\$_1^1\,;\,{}^2\$_1^3\right\} + \dot{q}_1\left\{{}^1\$_1^2\,;\,{}^2\$_1^3\right\}\\
\dot{\theta}_2\left\{{}^0\$_2^1\,;\,{}^2\$_2^3\right\} + \dot{q}_2\left\{{}^1\$_2^2\,;\,{}^2\$_2^3\right\}\\
\dot{\theta}_3\left\{{}^0\$_3^1\,;\,{}^2\$_3^3\right\} + \dot{q}_3\left\{{}^1\$_3^2\,;\,{}^2\$_3^3\right\}\\
\dot{\theta}_1\left\{{}^0\$_1^1\,;\,{}^3\$_1^4\right\} + \dot{q}_1\left\{{}^1\$_1^2\,;\,{}^3\$_1^4\right\}\\
\dot{\theta}_2\left\{{}^0\$_2^1\,;\,{}^3\$_2^4\right\} + \dot{q}_2\left\{{}^1\$_2^2\,;\,{}^3\$_2^4\right\}\\
\dot{\theta}_3\left\{{}^0\$_3^1\,;\,{}^3\$_3^4\right\} + \dot{q}_3\left\{{}^1\$_3^2\,;\,{}^3\$_3^4\right\}
\end{bmatrix}.$$

2.3 Acceleration Analysis

The reduced acceleration state, A_O, of a rigid body is defined as follows [2]

$$\boldsymbol{A}_O = \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{a}_O - \boldsymbol{\omega} \times \boldsymbol{v}_O \end{bmatrix} = \mathbf{J}_i \, \dot{\boldsymbol{\Omega}}_i + \mathscr{L}_i, \tag{8}$$

where $\boldsymbol{\alpha} = \dot{\boldsymbol{\omega}}$ and $\boldsymbol{a}_O$ is the angular and linear acceleration, respectily, of the mobile platform. $\dot{\boldsymbol{\Omega}}_i$ is the matrix whose elements are the joint acceleration ratios of the i-th chain, ${}_m\dot{\omega}^i_{m+1}$. Moreover $\mathscr{L}_i$ is given by the Lie products:

$$\mathscr{L}_i = \sum_{j=0}^{4} \begin{bmatrix} {}_j\omega^i_{j+1} \, {}^j\$_i^{j+1} & \sum_{k=j+1}^{5} {}_k\omega^i_{k+1} \, {}^k\$_i^{k+1} \end{bmatrix}.$$

Using a similar procedure shown in velocity analisys, and applying the Klein form between Eq. (8) and the reciprocal screws we have:

$$\mathbf{M}^{\mathrm{T}} \Delta \boldsymbol{A}_O = \ddot{\mathbf{Q}}, \tag{9}$$

where

$$\ddot{\mathbf{Q}} = \begin{bmatrix} \ddot{\theta}_1 \left\{ {}^0\$_1^1 \, ; \, {}^2\$_1^3 \right\} + \ddot{q}_1 \left\{ {}^1\$_1^2 \, ; \, {}^2\$_1^3 \right\} + \left\{ {}^2\$_1^3 \, ; \, \mathscr{L}_1 \right\} \\ \ddot{\theta}_2 \left\{ {}^0\$_2^1 \, ; \, {}^2\$_2^3 \right\} + \ddot{q}_2 \left\{ {}^1\$_2^2 \, ; \, {}^2\$_2^3 \right\} + \left\{ {}^2\$_2^3 \, ; \, \mathscr{L}_2 \right\} \\ \ddot{\theta}_3 \left\{ {}^0\$_3^1 \, ; \, {}^2\$_3^3 \right\} + \ddot{q}_3 \left\{ {}^1\$_3^2 \, ; \, {}^2\$_3^3 \right\} + \left\{ {}^2\$_3^3 \, ; \, \mathscr{L}_3 \right\} \\ \ddot{\theta}_1 \left\{ {}^0\$_1^1 \, ; \, {}^3\$_1^4 \right\} + \ddot{q}_1 \left\{ {}^1\$_1^2 \, ; \, {}^3\$_1^4 \right\} + \left\{ {}^3\$_1^4 \, ; \, \mathscr{L}_1 \right\} \\ \ddot{\theta}_2 \left\{ {}^0\$_2^1 \, ; \, {}^3\$_2^4 \right\} + \ddot{q}_2 \left\{ {}^1\$_2^2 \, ; \, {}^3\$_2^4 \right\} + \left\{ {}^3\$_2^4 \, ; \, \mathscr{L}_2 \right\} \\ \ddot{\theta}_3 \left\{ {}^0\$_3^1 \, ; \, {}^3\$_3^4 \right\} + \ddot{q}_3 \left\{ {}^1\$_3^2 \, ; \, {}^3\$_3^4 \right\} + \left\{ {}^3\$_3^4 \, ; \, \mathscr{L}_3 \right\} \end{bmatrix}.$$

3 Dynamic Analysis

In this section are calculated the driving forces of the pipe-bending machine, by means of a combination of the screw theory and the Principle of virtual work. This problem consist of: given the inertial, gravitational and external wrenches, to determine the driving forces/torques required to obtain the desired trajectory for the mobile platform.

According to the principle of D'Alembert [8], the inertial wrench acting on the j-th body of the i-th chain is given by:

$$\boldsymbol{F}^{j,i}_{I,*} = \begin{bmatrix} -m_{j,i} \, {}^0\boldsymbol{a}^j_{i,*} \\ -\mathbf{I}^0_{j,i} \, {}^0\boldsymbol{\alpha}^j_i - {}^0\boldsymbol{\omega}^j_i \times \mathbf{I}^0_{j,i} \, {}^0\boldsymbol{\omega}^j_i \end{bmatrix}; \tag{10}$$

where $m_{j,i}$ is the mass of the body, ${}^0\boldsymbol{a}^j_{i,*}$, is the translational acceleration of the mass center and $\mathbf{I}^0_{j,i}$ is the centroidal inertia tensor with respect to global reference system.

We consider the gravitational wrench $\boldsymbol{F}^{j,i}_{G,*} = \left[m_{j,i}\,\boldsymbol{g}\,;\,\boldsymbol{0}\right]^{\mathrm{T}}$, where $\boldsymbol{g}$ is the acceleration of gravity, and a external wrench $\boldsymbol{F}^{j,i}_{E,*}$ applied to the body j at its center of mass. Then, the resultant wrench acting on body j of the limb i is

$$\boldsymbol{F}^{j,i} = \boldsymbol{F}^{j,i}_{I,*} + \boldsymbol{F}^{j,i}_{G,*} + \boldsymbol{F}^{j,i}_{E,*}. \tag{11}$$

In [2] is shown that the power $w_{j,i}$ produced by $\boldsymbol{F}^{j,i}$ that acts on the j body of the i-th chain with velocity state ${}^0\boldsymbol{V}^{j,i}_*$, whose reference pole is its center of mass, can be determined by:

$$w_{j,i} = \left\{\boldsymbol{F}^{j,i}\,;\,{}^0\boldsymbol{V}^{j,i}_*\right\}. \tag{12}$$

In order to apply the principle of virtual work it is necessary to express ${}^0\boldsymbol{V}^{j,i}_*$ as a function of the generalized velocities, $\dot{q}_i$ and $\dot{\theta}_i$. From Eqs. (7) and (4):

$$\boldsymbol{\Omega}_i = (\mathbf{J}_i)^{-1}\left(\mathbf{M}^{\mathrm{T}}\,\Delta\right)^{-1}\dot{\mathbf{Q}} \tag{13}$$

In that way, the speed ratio of body j respect to body $j-1$, belonging to chain i, is expressed in terms of $\dot{q}_i$ and $\dot{\theta}_i$ as follows:

$${}_{j-1}\omega^i_j = \sum_{k=1}^{6}\left({}_{j-1}G^{k,i}_j\,\underline{\dot{q}}_k\right), \tag{14}$$

where $\underline{\dot{q}}_k = \dot{q}_k,\ \forall k = 1,2,3$ and $\underline{\dot{q}}_k = \dot{\theta}_k,\ \forall k = 4,5,6$. Also, the scalars ${}_{j-1}G^{m,i}_j$ are the first order kinematic influence coefficients. Then the velocity state of body j belonging to the i-th chain can be expressed as:

$${}^0\boldsymbol{V}^{j,i}_O = \sum_{k=0}^{j-1}\left[\sum_{m=1}^{6}\left({}_{k-1}G^{m,i}_k\,\underline{\dot{q}}_m\right)\right]{}^k\$^{k+1}_i. \tag{15}$$

Then, grouping the terms in $\underline{\dot{q}}_i$, and taking the mass center of the body as representation pole, leads to:

$${}^0\boldsymbol{V}^{j,i}_* = \sum_{k=1}^{6}\$^k_{j,i}\,\underline{\dot{q}}_k \tag{16}$$

where $\$^1_{j,i}$, $\$^2_{j,i}$ $\ldots$, $\$^6_{j,i}$ are called partial screws [2]. Thus, Eq. (12) can be rewritten as follows:

Table 1 Inertial properties

Body	**Mass** (kg)	I_x (kg· mm^2)	I_y	I_z	I_{xy}	I_{yz}	I_{xz}
Arm 1	13.072	53221.961	400773.773	428111.073	15739.096	4163.259	17844.419
Arm 2	12.210	47901.779	376056.888	401044	23041.162	−3383.654	−6010.009
Arm 3	13.240	53812.305	407679.281	432720.353	19736.403	−5021.887	7555.891
P. joint	1.529	2029.198	2132.738	2132.738	298.504	0	0
Leg	0.583	4316.549	94.821	4256.381	−1.294	0.027	0
M. Plat.	0.322	408.900	456.586	715.893	0.012	−0.001	23.185

$$w = \left\{ \boldsymbol{F}^6 \,;\, \boldsymbol{V}_*^6 \right\} + \sum_{i=1}^{6} \left(\sum_{j=1}^{5} \left\{ \boldsymbol{F}^{j,i} \,;\, {}^0\boldsymbol{V}_*^{j,i} \right\} + \tau_i \dot{\underline{q}}_i \right), \tag{17}$$

where τ_i is the driving force/torque associated to the generalized velocity $\dot{\underline{q}}_i$.

The principle of virtual work states that if a multi-body system is in equilibrium under the effect of external actions, then the global work produced by the external forces with any virtual velocity must be null [8]. Taking into a count the virtual velocities $\delta\dot{\underline{q}}_i$, substituting Eq. (16) in Eq. (17), and rearranging terms, leads to

$$\delta w = \sum_{i=1}^{6} \left[\left(\left\{ \boldsymbol{F}^6 \,;\, \$_6^i \right\} + \sum_{j=1}^{5} \left\{ \boldsymbol{F}^{j,i} \,;\, \$_{j,i}^i \right\} + \tau_i \right) \delta\dot{\underline{q}}_i \right] = 0. \tag{18}$$

Since the generalized virtual velocities $\delta\underline{q}_i$ are arbitrary, and $\delta w = 0$, it is necessary and sufficient that the coefficients of the virtual displacements $\delta\underline{q}_i$ are zero.

$$\sum_{i=1}^{6} \left(\left\{ \boldsymbol{F}^6 \,;\, \$_6^i \right\} + \sum_{j=1}^{5} \left\{ \boldsymbol{F}^{j,i} \,;\, \$_{j,i}^i \right\} + \tau_i \right) = 0. \tag{19}$$

Finally, from Eq. (19) the generalized forces τ_i can be computed directly.

4 Numerical Simulations

In order to show the potential of this method to calculate the driving forces/torques of the robot, a numerical example is presented. The geometrical parameters of the robot are $d = 250$ mm, $e = 55\sqrt{3}$ mm, $\beta = 20°$, and the inertial properties of the links are depicted in Table 1. Moreover, the initial position, in millimeters, of the robot is described in A_i and B_i points:

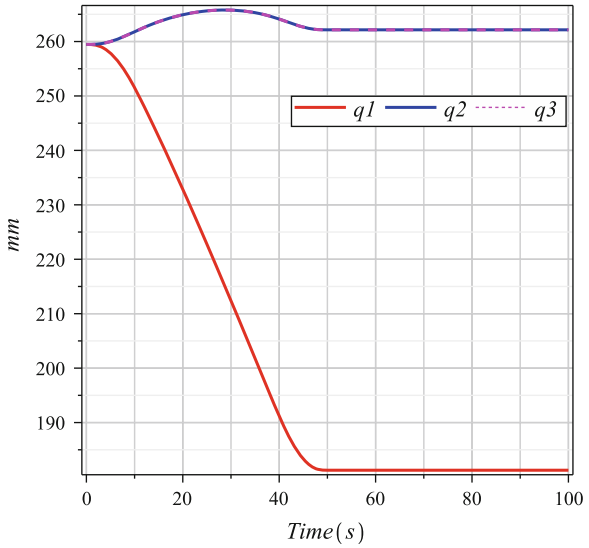


Fig. 3 Displacements of the P joints

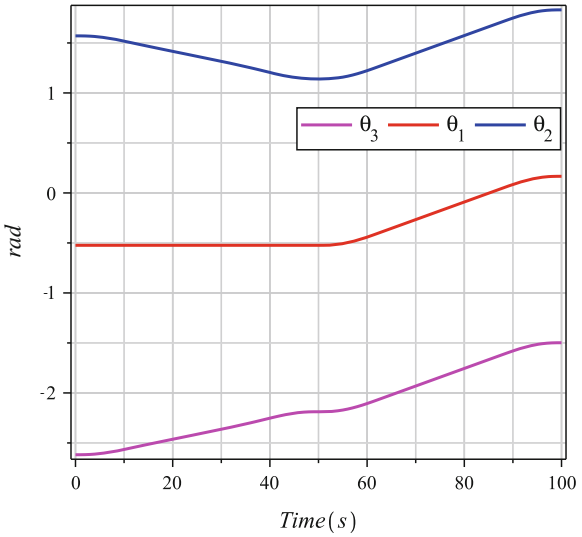


Fig. 4 Displacements of the R joints

$B_1 = (30.630, 76.384, 47.012)$; $A_1 = (211.150, -88.742, 121.908)$;
$B_2 = (28.866, 76.499, -48.234)$; $A_2 = (-0.001, -88.742, -243.816)$;
$B_3 = (-52.637, 72.359, 0.909)$; $A_3 = (-211.150, -88.742, 121.901)$;

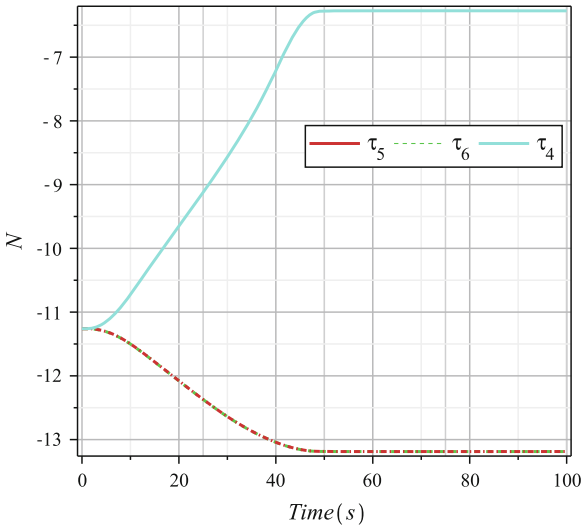


Fig. 5 Driving forces of P joints

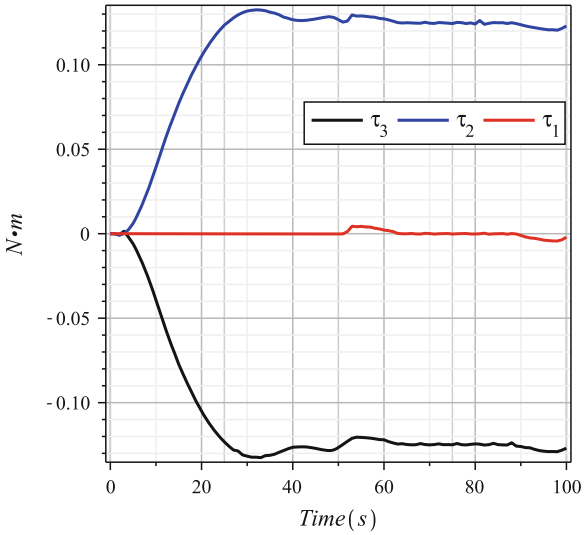


Fig. 6 Driving forces of R joints

On the other hand, the time history of the generalized coordinates are shown in Figs. 3 and 4. Please note that $q_2 = q_3$. The results are shown in Figs. 5 and 6. Note that $\tau_5 = \tau_6$.

5 Conclusions

In this chapter the dynamic analysis of a 3-RPSR parallel robot has been solved using a combination of screw theory and the principle of virtual work. Equation for displacement analysis has been derived based on mechanical constraint conditions of joints. The velocity and acceleration states of the moving platform are written in screw form through each one of the three chains of the robot. The expressions obtained are linear, simple and compact through the use of the Klein form. An advantage of this formulation is that it is not necessary to know the velocities and accelerations of passive joints to solve forward velocity and acceleration analysis, respectively.

Finally, the driving forces required to obtain the desired trajectory for the mobile platform are calculated considering both the screw theory and the principle of virtual work. Resulting equations are simple and linear, unlike those obtained by the Newton-Euler method, the Lagrangian and some others and it is not necessary to determine the instantaneous values of the internal reactions of the mechanism, or calculate the energy of the entire system.

References

1. Carbonari, L., Battistelli, M., Callegari, M., Palpacelli, M.C.: Dynamic modelling of a 3-CPU parallel robot via screw theory. Mech. Sci. **4**(1), 185–197 (2013)
2. Gallardo, J., Rico, J.M., Frisoli, A., Checcacci, D., Bergamasco, M.: Dynamics of parallel manipulators by means of screw theory. Mech. Mach. Theory **38**(11), 1113–1131 (2003)
3. Liu, W., Gong, Z., Wang, Q.: Investigation on Kane dynamic equations based on screw theory for open chain manipulator. Appl. Math. Mech. **26**(5), 627–635 (2005)
4. Lopes, A.M.: Dynamic modeling of a Stewart platform using the generalized momentum approach. Commun. Nonlinear Sci. Numer. Simul. **14**(8), 3389–3401 (2009)
5. Miller, K.: Optimal design and modeling of spatial parallel manipulators. Int. J. Robot. Res. **23**(2), 127–140 (2004)
6. Takeda, Y., Inada, S., Kawasumi, S., Matsuura, D., Hirose, K., Ichiryu, K.: Kinematic design of 3-RPSR parallel mechanism for movable-die drive mechanism of pipe bender. Rom. J. Tech. Sci. Appl. Mech. **58**, 47–72 (2013)
7. Takeda, Y., Xiao, X., Higuchi, M.: Development of a pipe bender using a parallel mechanism with 3-RPSR structure with six degrees of freedom. In: 13th World Congress in Mechanism and Machine Science, pp. 19–25. Guanajuato, México (2011)
8. Tsai, L.W.: Robot Analysis: The Mechanics of Serial and Parallel Manipulators, 3rd edn. John Wiley and Sons, New York (1999)

Kinematic Synthesis of a Watt I Six-Bar Linkage for Body Guidance

Mark Plecnik, J. Michael McCarthy and Charles W. Wampler

Abstract This chapter formulates the synthesis equations for a Watt I six-bar linkage that moves through N specified task positions. For the maximum number of positions, $N = 8$, the resulting polynomial system consists of 28 equations in 28 unknowns, which can be separated into a nine sets of variables yielding a nine-homogeneous Bezout degree of 3.43×10^{10}. We verify these synthesis equations by finding isolated solutions via Newton's method, but a complete solution for $N = 8$ seems beyond the capability of current homotopy solvers. We present a complete solution for $N = 6$ positions with both ground pivots specified.

Keywords Kinematic synthesis · Six-bar linkage

1 Introduction

This chapter presents the synthesis equations for a Watt I six-bar linkage, Fig. 1a, to guide a rigid body through N specified task positions. This is a generalization of the motion generation problem for four-bar linkages, see Hartenberg and Denavit [5], Erdman et al. [3], and McCarthy and Soh [6].

Our formulation yields 28 equations in 28 unknowns for the maximum number of task positions, $N = 8$, which has a nine-homogeneous Bezout degree of 3.43×10^{10}.

M. Plecnik (✉) · J. M. McCarthy
Robotics and Automation Laboratory, University of California, Irvine, CA 92697, USA
e-mail: mplecnik@uci.edu

J. M. McCarthy
e-mail: jmmccart@uci.edu

C. W. Wampler
General Motors R&D Center, Warren, MI 48090-9055, USA
e-mail: charles.w.wampler@gm.com

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_33,

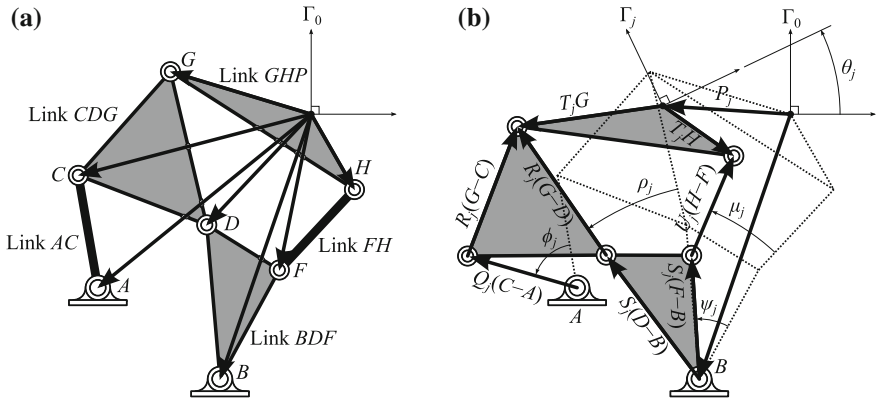


Fig. 1 **a** Vectors A, B, C, D, F, G, H locate the pivots in position $j = 0$ from the fixed frame coordinate system Γ_0. **b** The three vector loops in position j. Note that the vector A has not been drawn in order to make the figure more clear

The size of this problem seems to be beyond the ability of polynomial continuation software to formulate a start system. In what follows, we present the solution for six task positions with the locations of the two ground pivots specified.

2 Literature Review

The synthesis theory for six-bar linkages focuses on obtaining specified input-output angles for function generation. In 1944, Svoboda [12] used a nomograph formulation to design a six-bar linkage that generates a logarithmic function. Following the work by Freudenstein [4], McLarnan [7] formulated the loop equations for Watt II, Stephenson II and III six-bar linkages for use as function generators which he solved on an IBM 704 computer. Dhingra et al. [2] formulated synthesis equations for Watt II, Stephenson II and III function generators which they solved using numerical homotopy methods on a 486 PC.

The closest work to this chapter is Soh and McCarthy [10] who formulate the five-position body guidance problem for a Watt I linkage as a constrained 3R chain. Pennock and Israr [8] design an adjustable six-bar linkage for function generation. Other recent work is Shiakolas et al. [9], who used an optimization formulation to design six-bar linkages for the combination of crank angle and coupler point location.

This work follows Wampler et al. [14] and formulates the loop equations of the linkage using vectors in the complex plane and their conjugates together known as *isotropic coordinates*. We use the polynomial continuation software Bertini to solve the system of equations [1, 11].

3 Synthesis Equations

A Watt I six-bar linkage, Fig. 1a, can be viewed as a four-bar linkage sitting on top of another four-bar linkage. The base four-bar linkage consists of the ground link AB, the cranks AC and BDF, and the floating link CDG. The second four-bar linkage is attached to BDF, has cranks CDG and FH, and the floating link GHP. The floating link GHP is considered to be the end-effector of the system. The goal is to place a task frame attached to GHP in N positions Γ_j for $j = 0, \ldots, N-1$.

A task frame Γ_j is defined by the coordinates of its origin P_j and its orientation angle measured relative to a fixed frame. Complex coordinates are used to define these parameters, so

$$\Gamma_j = (x_j + iy_j, e^{i\theta_j}) = (P_j, T_j). \tag{1}$$

For convenience, choose the fixed frame to coincide with the first task frame, so $\Gamma_0 = (0, 1)$, Fig. 1.

The coordinates of the seven pivots are identified in position $j = 0$ by the complex vectors A, B, C, D, F, G, H as shown in Fig. 1a. The coordinates of these vectors are 14 unknowns of the linkage synthesis problem. Following Wampler [13], we consider these vectors and their complex conjugates as separate unknowns, which yields a total of 28 unknowns.

The synthesis equations are obtained from three vector loop equations for the Watt I six-bar linkage formulated in each of the specified task positions. The rotations of each joint relative to the initial configuration are defined by the pairs of complex vectors and their conjugates,

$$\begin{aligned} (Q_j, \bar{Q}_j) &= (e^{i\phi_j}, e^{-i\phi_j}), \quad (R_j, \bar{R}_j) = (e^{i\rho_j}, e^{-i\rho_j}), \\ (S_j, \bar{S}_j) &= (e^{i\psi_j}, e^{-i\psi_j}), \quad (U_j, \bar{U}_j) = (e^{i\mu_j}, e^{-i\mu_j}). \end{aligned} \tag{2}$$

The overbar denotes the complex conjugate. Notice that each pair satisfies the condition that their product equals one, that is,

$$Q_j\bar{Q}_j = 1, \quad R_j\bar{R}_j = 1, \quad S_j\bar{S}_j = 1, \quad U_j\bar{U}_j = 1, \qquad j = 1, \ldots, N-1. \tag{3}$$

This yields $4(N-1)$ equations in the unknown joint rotation angles. Using these rotation unit vectors, we obtain three sets of loop equations from Fig. 1b,

$$\mathscr{A}_j = \begin{cases} A + Q_j(C - A) + R_j(G - C) - T_jG &= P_j \\ \bar{A} + \bar{Q}_j(\bar{C} - \bar{A}) + \bar{R}_j(\bar{G} - \bar{C}) - \bar{T}_j\bar{G} &= \bar{P}_j \end{cases} \quad j = 1, \ldots, N-1, \tag{4}$$

$$\mathscr{B}_j = \begin{cases} B + S_j(D - B) + R_j(G - D) - T_jG &= P_j \\ \bar{B} + \bar{S}_j(\bar{D} - \bar{B}) + \bar{R}_j(\bar{G} - \bar{D}) - \bar{T}_j\bar{G} &= \bar{P}_j \end{cases} \quad j = 1, \ldots, N-1, \tag{5}$$

$$\mathscr{C}_j = \begin{cases} B + S_j(F - B) + U_j(H - F) - T_jH &= P_j \\ \bar{B} + \bar{S}_j(\bar{F} - \bar{B}) + \bar{U}_j(\bar{H} - \bar{F}) - \bar{T}_j\bar{H} &= \bar{P}_j \end{cases} \quad j = 1, \ldots, N-1. \tag{6}$$

This is $6(N-1)$ equations in the pivot location and joint angle unknowns.

The collection of the joint normalization conditions (3) and the three sets of loop equations $\mathscr{A}_j$, $\mathscr{B}_j$, and $\mathscr{C}_j$ yields $10(N-1)$ equations in the $8N+6$ unknowns,

$$\langle A, \bar{A}, B, \bar{B}, C, \bar{C}, D, \bar{D}, F, \bar{F}, G, \bar{G}, H, \bar{H}\rangle \langle Q_j, \bar{Q}_j, R_j, \bar{R}_j, S_j, \bar{S}_j, U_j, \bar{U}_j\rangle \quad j = 1, \ldots, N-1. \tag{7}$$

For $N = 8$, this system is square and has a total degree $2^{70} \approx 1.18 \times 10^{21}$.

The joint rotation angles $(Q_j, \bar{Q}_j)$ can be eliminated by solving the loop equations $\mathscr{A}_j$ and substituting the result into its normalization condition of (3). Similarly, $(U_j, \bar{U}_j)$ are eliminated using the loop equations $\mathscr{C}_j$ and the normalization condition of (3). The resulting equations are

$$(P_j - A + R_j(C-G) + T_j G)(\bar{P}_j - \bar{A} + \bar{R}_j(\bar{C}-\bar{G}) + \bar{T}_j\bar{G}) = (C-A)(\bar{C}-\bar{A}), \tag{8}$$

$$(P_j - B + S_j(B-F) + T_j H)(\bar{P}_j - \bar{B} + \bar{S}_j(\bar{B}-\bar{F}) + \bar{T}_j\bar{H}) = (H-F)(\bar{H}-\bar{F}), \quad j = 1, \ldots, N-1. \tag{9}$$

This yields a system of $6(N-1)$ equations in $4N+10$ unknowns. The degree of this system is $(2^4 \cdot 3^2)^{N-1}$, which for $N = 8$ is approximately 1.28×10^{15}.

Now, eliminate the variables $(R_j, \bar{R}_j)$ by solving the loop equations $\mathscr{B}_j$. Substitute the result into the normalization conditions of (3) to obtain,

$$(P_j - B + S_j(B-D) + T_j G)(\bar{P}_j - \bar{B} + \bar{S}_j(\bar{B}-\bar{D}) + \bar{T}_j\bar{G}) = (G-D)(\bar{G}-\bar{D}), \quad j = 1, \ldots, N-1. \tag{10}$$

Substitution of $(R_j, \bar{R}_j)$ into (8) gives,

$$\begin{aligned}((G-D)(P_j - A + T_j G) + w_j(C-G))((\bar{G}-\bar{D})(\bar{P}_j - \bar{A} + \bar{T}_j\bar{G}) + \bar{w}_j(\bar{C}-\bar{G})) \\ = (C-A)(\bar{C}-\bar{A})(G-D)(\bar{G}-\bar{D}), \quad j = 1, \ldots, N-1,\end{aligned} \tag{11}$$

where

$$w_j = P_j - B + S_j(B-D) + T_j G, \quad \bar{w}_j = \bar{P}_j - \bar{B} + \bar{S}_j(\bar{B}-\bar{D}) + \bar{T}_j\bar{G}. \tag{12}$$

After this elimination process, the system consists of the normalization conditions, $S_j\bar{S}_j = 1$ and (9), (10), and (11). which tallies to $4(N-1)$ equations in $2N+12$ unknowns. The total degree of these equations is $(2 \cdot 3^2 \cdot 5)^{N-1}$, which for $N = 8$ is approximately 4.78×10^{13}. To compute the multihomgeneous degree, we introduce the $N+1$ groups,

Table 1 Eight task positions

j	x_j	y_j	θ_j (°)
0	0	0	0
1	−0.08849958	0.63143282	5.95778240
2	−0.49237057	1.35531439	17.35017046
3	−1.14387347	1.93359099	31.25403761
4	−1.92567709	2.25400993	45.59910917
5	−2.68188520	2.29738239	58.34343392
6	−3.27440782	2.13240439	68.00251408
7	−3.61649767	1.86582938	73.93850323

$$\langle A, B, C, D, F, G, H\rangle, \langle \bar{A}, \bar{B}, \bar{C}, \bar{D}, \bar{F}, \bar{G}, \bar{H}\rangle,$$
$$\text{and } \langle S_j, \bar{S}_j\rangle, \quad j = 1, \ldots, N-1. \tag{13}$$

For $N = 8$ this grouping yields a nine-homogeneous Bezout degree of 3.43×10^{10}.

4 Numerical Solution for Eight Positions

The synthesis equations were validated by generating a few solutions using an implementation of Newton's method provided by the Mathematica computational software package called FindRoot. We solved for the task positions listed in Table 1. Example solutions are listed in Table 2. Solution 1 is shown in Fig. 2.

Newton's method was used to solve the synthesis equations for a randomized set of 100,000 of start points. The computation took 46 min. All computations of this chapter were done in parallel on a 64 core machine. The start point values of the isotropic coordinate pairs $(A, \bar{A}), \ldots, (H, \bar{H})$ were randomized within a 10×10 box centered on the origin of the complex plane. The start point values of $(S_j, \bar{S}_j)$, $j = 1, \ldots, 7$ were specified to be random complex numbers of unit magnitude. All isotropic coordinate start points maintained conjugate relationships throughout randomization.

This yielded 28,890 solutions, of which only 64 were suitable for linkage analysis. Solutions suitable for linkage analysis are those whose isotropic pairs are truly conjugates and that have link lengths AB, AC, BD, DF, CD, DG, FH, GH greater than 0.1. Only two of the suitable solutions corresponded to defect-free linkages, Solutions 1 and 4 in Table 2.

Table 2 A sample of solutions to the eight position synthesis equations

	Solution 1		Solution 2		Solution 3		Solution 4	
	Re	Im	Re	Im	Re	Im	Re	Im
A	−2.750557	−0.331168	−2.688674	−0.294929	−3.464722	−0.981841	−2.133969	−1.241728
B	−6.443286	−0.179046	−2.918281	−1.066570	−2.979378	−1.882345	−2.456924	−1.199599
C	−1.819766	−0.499668	−1.695131	−0.453426	−2.776606	−1.762104	−1.973392	−1.046694
D	−2.995680	−0.283615	−3.054520	−1.255878	−2.796190	−2.001717	−2.233036	−1.203616
F	−5.386180	−0.695399	−3.134324	−1.075385	−2.965783	−2.550331	−2.403520	−1.686455
G	−1.815717	−0.498056	−1.492112	−0.390568	−3.157675	−1.868198	−2.884410	−1.164443
H	−2.852650	−1.683542	−3.191124	−1.473735	−2.619687	−2.233975	−2.041865	−1.797497
S_1	0.999690	−0.024890	−0.384794	−0.923002	0.686117	0.727491	0.915234	0.402922
S_2	0.999171	−0.040701	−0.764101	0.645097	0.236524	0.971626	0.386722	0.922197
S_3	0.998217	−0.059687	−0.985651	−0.168793	−0.110517	0.993874	−0.931545	0.363626
S_4	0.838386	0.545077	−0.997586	−0.069446	−0.405476	0.914105	−0.664287	0.747478
S_5	0.831441	0.555613	−0.965251	0.261323	−0.617109	0.786878	−0.979973	0.199132
S_6	0.986016	−0.166650	−0.777383	0.629027	−0.824702	0.565568	−0.980540	−0.196322
S_7	0.915030	0.403386	0.789645	0.613565	−0.870772	0.491688	−0.847467	−0.530849

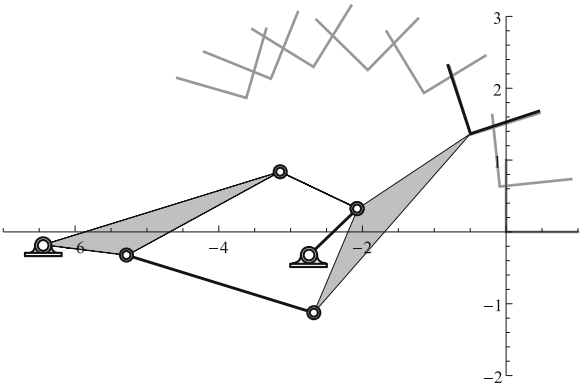


Fig. 2 Watt I six-bar linkage that passes through eight specified task positions

5 Solution for Six Positions

If only $N = 6$ positions are specified, then we may specify four conditions on the design parameters. One choice is to specify the ground pivot locations $(A, \bar{A})$ and $(B, \bar{B})$. This is natural for design purposes and also leads to considerable simplification in the numerical treatment of the problem. With $N = 6$ there are 20 synthesis equations in 20 unknowns with a total degree of $(2 \cdot 3^2 \cdot 5)^5 \approx 5.90 \times 10^9$. In this case, however, the unknowns separate into the ten groups

$$\langle C, \bar{C} \rangle, \langle D, \bar{D} \rangle, \langle F, \bar{F} \rangle, \langle G, \bar{G} \rangle, \langle H, \bar{H} \rangle, \text{and } \langle S_j, \bar{S}_j \rangle, \quad j = 1, \ldots, 5, \qquad (14)$$

Table 3 Paths tracked and nonsingular endpoints found at each level of regeneration

Level	Paths	Nonsingular endpoints
0	3	3
1	9	9
2	25	25
3	63	61
4	138	130
5	548	437
6	1421	788
7	2062	980
8	2022	868
9	1346	500
10	900	900
11	1600	1600
12	2400	2341
13	2941	2042
14	2042	808
15	1616	1548
16	3095	2880
17	5760	4867
18	9734	6638
19	13275	5735

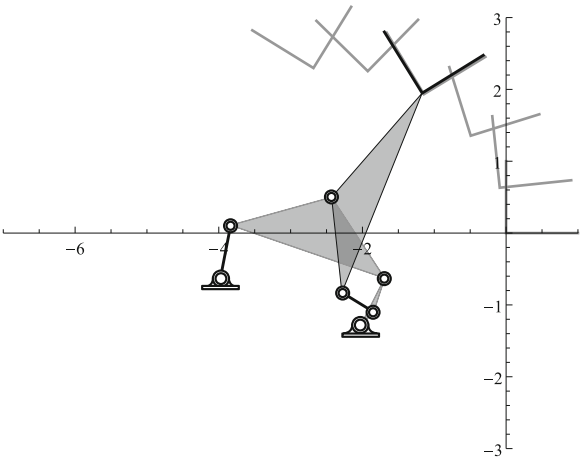


Fig. 3 A Watt I six-bar linkage that reaches six task positions

which yields the ten-homogeneous Bezout degree 1,998,720. The Bertini numerical continuation software was used to obtain a complete solution to this problem.

The solution of this problem takes advantage of regeneration. Regeneration tracks solutions for several levels of homotopies. In a square system of n equations and

Table 4 A sample of solutions to the six position synthesis equations

	Solution 1		Solution 2		Solution 3		Solution 4	
	Re	Im	Re	Im	Re	Im	Re	Im
C	−3.277663	−0.360479	−2.438554	−0.627923	−2.672919	−1.937923	−2.787652	−3.185862
D	−1.612335	−1.898356	−0.432594	−0.991854	−1.172132	−2.538575	−3.547796	−1.100739
F	−1.948622	−1.535115	−0.322911	−2.589864	−2.175566	−1.663265	−6.427970	−0.222094
G	−1.835116	−0.569740	−1.204924	−0.030339	−2.786399	−2.523568	−2.103420	−1.644206
H	−2.410335	−1.772735	0.116380	−2.016269	−4.636286	−1.395320	−2.387046	−2.239104
S_1	0.626133	0.779716	0.956723	0.290999	0.989602	0.143835	−0.978732	0.205142
S_2	0.040843	0.999166	0.783349	0.621582	−0.876928	−0.480621	0.999287	−0.037744
S_3	−0.473696	0.880688	0.493733	0.869614	−0.916843	−0.399247	−0.905977	−0.423328
S_4	−0.822092	0.569355	0.144621	0.989487	0.403962	0.914776	0.998733	0.050325
S_5	−0.978809	0.204776	−0.194393	0.980924	−0.977576	−0.210581	0.985576	0.169231

unknowns, the first level solves a system composed of 1 equation of the target system and $n-1$ arbitrarily specified linear equations in all unknowns. The finite isolated solutions of the first level are used to solve the second level system. At each succeeding level, one linear equation is replaced with one target system equation until the final level when the target system is solved. A detailed description of regeneration is found in Bates et al. [1].

Our level by level results of regeneration are shown in Table 3. Finally, 5,735 nonsingular solutions were found after tracking a total of 51,000 paths. The computation took about 7 h.

The solution set and parameters from the regeneration run provided a parameter homotopy for the synthesis equations. The base pivots were specified to be $A = -3.976225 - 0.623063i$ and $B = -2.024139 - 1.285906i$. This computation took about 5 min and yielded 5,556 nonsingular solutions of which 243 were suitable to analyze and 43 were defect-free. Example solutions are listed in Table 4. Solution 1 is shown in Fig. 3.

6 Conclusion

The chapter derives the synthesis equations for Watt I six-bar mechanisms that guides a body through N task positions. For $N = 8$, we were unable to solve this problem using polynomial continuation but did obtain solutions via repeated trials of Newton's method.

For the simpler problem of six positions with specified ground pivots, we computed a complete solution using polynomial regeneration. This gives a set of 5,735 solutions that serve as the start points for a parameter homotopy to solve any particular case in about 5 min on a 64 core parallel computer.

References

1. Bates, D.J., Hauenstein, J.D., Sommese, A.J., Wampler, C.W.: Numerically solving polynomial systems with Bertini. SIAM Books, Philadelphia, PA (2013)
2. Dhingra, A., Cheng, J., Kohli, D.: Synthesis of six-link, slider-crank and four-link mechanisms for function, path and motion generation using homotopy with m-homogenization. J. Mech. Des. **116**(4), 1122–1131 (1994)
3. Erdman, A.G., Sandor, G.N., Kota, S.: Mechanism Design: Analysis and Synthesis. Prentice Hall, Upper Saddle River (2001)
4. Freudenstein, F.: An analytical approach to the design of four-link mechanisms. Trans. ASME **76**, 483–492 (1954)
5. Hartenberg, R.S., Denavit, J.: Kinematic Synthesis of Linkages. McGraw-Hill, New York (1964)
6. McCarthy, J.M., Soh, G.S.: Geometric Design of Linkages, 2nd edn. Springer, New York (2010)
7. McLarnan, C.: Synthesis of six-link plane mechanisms by numerical analysis. J. Eng. Indus. **85**(1), 5–10 (1963)
8. Pennock, G.R., Israr, A.: Kinematic analysis and synthesis of an adjustable six-bar linkage. Mech. Mach. Theory **44**(2), 306–323 (2009)
9. Shiakolas, P., Koladiya, D., Kebrle, J.: On the optimum synthesis of six-bar linkages using differential evolution and the geometric centroid of precision positions technique. Mech. Mach. Theory **40**(3), 319–335 (2005)
10. Soh, G.S., McCarthy, J.M.: The synthesis of six-bar linkages as constrained planar 3r chains. Mech. Mach. Theory **43**(2), 160–170 (2008)
11. Sommese, A.J., Wampler, C.W.: The Numerical Solution of Systems of Polynomials Arising in Engineering and Science. World Scientific, Singapore (2005)
12. Svoboda, A.: Computing Mechanisms and Linkages. McGraw-Hill, New York (1948)
13. Wampler, C.W.: Isotropic coordinates, circularity and bezout numbers: planar kinematics from a new perspective. In: Proceedings of the 1996 ASME Design Engineering Technical Conference, Irvine, California August, pp. 18–22 (1996).
14. Wampler, C.W., Sommese, A., Morgan, A.: Complete solution of the nine-point path synthesis problem for four-bar linkages. J. Mech. Des. **114**(1), 153–159 (1992)

Collision-Free Workspace of 3-RPR Planar Parallel Mechanism via Interval Analysis

MohammadHadi FarzanehKaloorazi, Mehdi Tale Masouleh and Stéphane Caro

Abstract This chapter proposes an interval-based approach, to obtain the collision-free workspace of planar parallel mechanisms. This approach is represented through an example for a 3-RPR planar parallel mechanism, but it can be extended to investigate the obstacle avoidance of higher degree of freedom and spatial mechanisms. Three main feature of the collision-free workspace is taken into account: mechanical stroke of actuators, interference of limbs with the obstacle and interference of end-effector with the obstacle. In this chapter a circle shaped obstacle is considered and its mechanical interference with limbs and edges of the end-effector is taken into account. The results are represented and ongoing works are mentioned.

Keywords Collision-free · Workspace · Obstacle avoidance · Mechanical interference · Parallel mechanisms · Interval analysis

1 Introduction

Parallel mechanisms are known to be more precise and able to carry more loads with respect to serial manipulators [8]. Besides several advantages, they are restricted by their limited workspace. Therefore, in practice, the presence of an obstacle inside their workspace should be taken into consideration in order to alleviate this limitation. Obtaining the collision-free workspace of parallel mechanisms leads to have a

M. FarzanehKaloorazi (✉) · M. T. Masouleh
Faculty of New Sciences and Technologies, University of Tehran, Tehran, Iran
e-mail: hamidfarzane88@gmail.com

M. T. Masouleh
e-mail: m.t.masouleh@ut.ac.ir

S. Caro
IRCCyN/CNRS UMR 6597, 1 rue de la Noë, 44321 Nantes, France
e-mail: stephane.caro@irccyn.ec-nantes.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_34,

conservative workspace for which all actions inside the aforementioned workspace are free of collision. Furthermore, obtaining the collision-free workspace is a definite asset in path planning and obstacle avoidance while controlling parallel mechanisms.

The problems of path planning and obstacle avoidance have been frequently investigated in the literature [1, 2, 5, 11]. Brooks et al. in [2], propose an efficient algorithm that finds collision-free paths for a manipulator with five or six revolute joints. Yang et al. in [11], investigated dynamic collision-free trajectory generation in a non-stationary environment, using biologically inspired neural network approaches. Brocks and Khatib in [1], represented elastic strip framework that enables the execution of a previously planned motion in a dynamic environment for robots with many degrees of freedom. Khatib et al. in [5], presented developments of models, strategies, and algorithms dealing with a number of autonomous capabilities that are essential for robot operations in human environments. These capabilities include: integrated mobility and manipulation, cooperative skills between multiple robots, interaction ability with humans, and efficient techniques for real-time modification of collision-free paths. In [6], Komainda and Hiller represented a concept for motion control of redundant manipulators in a changing environment. In [4], Jiménez et al. represented a general approach to cover all distance computation algorithms, hierarchical object representations, orientation-based pruning criteria, and space partitioning schemes. In [10], Wenger and Chedmail represented the collision-free workspace of serial manipulators. Caro et al. in [3], introduced a new method, based on numerical constraint programming, to compute a certified enclosure of the generalized aspects.

The mathematical framework used in this chapter is based on interval analysis [9]. Interval analysis is a reliable method to evaluate functions and is used frequently in the field of robotics [7]. An interval variable of $[x] = [\underline{x}, \overline{x}]$ is a set of all real numbers from the lower bound, $\underline{x}$, to the upper bound, $\overline{x}$. In order to have more details, see interval analysis in [9].

The remainder of this chapter is organized as follows. First, the the concept of the algorithm to obtain the collision-free workspace is broadly reviewed. Then, the pseudo-code of the algorithm is explained. Finally, the results and the conclusion of the obtained collision-free workspace are presented.

2 Obstacle Avoidance Formulation via Geometrical Concept

In order to avoid an obstacle, in general, the links of limbs and the edges of the End-Effector (EE) are considered to be straight line segments. Through this chapter, the obstacle is a circle; P_x and P_y being the Cartesian coordinate of its center point and P_r its radius. The problem of obtaining the collision-free workspace via interval analysis can be divided into two cases: the collision between the obstacle and (a) a line passing through one point and one box (b) through two boxes. The first case is applicable for those limbs which are connected to the fixed frame via a revolute joint; for instance, A_1B_1, A_2B_2 and A_3B_3 in the 3-RPR Planar Parallel Mechanism (PPM)

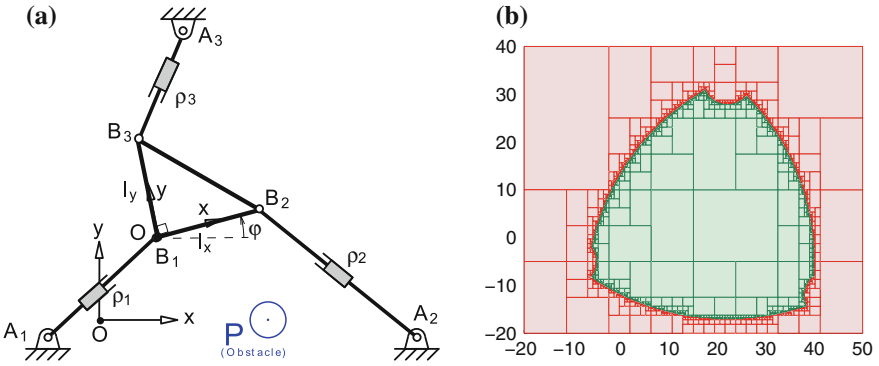


Fig. 1 **a** Schematic representation and **b** the constant-orientation workspace of a 3-RPR PPM. $A_1 = \{-10, -5\}$, $A_2 = \{50, -5\}$ and $A_3 = \{15, 40\}$; $P_x = 20$, $P_y = 0$ and $P_r = 3$; $l_x = l_y = 10$. **a** 3-RPR PPM and a circle shaped obstacle located at point P. $\rho_{\min} = 5$ and $\rho_{\max} = 50$, **b** The constant-orientation workspace of the 3-RPR PPM for $\phi = \pi/4$, via interval analysis, collisions are ignored. *Green boxes* are inside and *red boxes* are outside the workspace

shown in Fig. 1a. The second case is more general and is applicable for those links which are the medial or distal links of the limb; these links are not existed in the 3-RPR PPM. Moreover, the edges of the EE should be categorized in the second case; in the 3-RPR PPM, B_1B_2, B_2B_3 and B_3B_1.

A simple solution to obtain the collision-free workspace is to write the equation from the distance of a point to a line in the 2D space, which can be written as:

$$d = \sqrt{\left(\frac{P_x + mP_y - mc}{m^2+1} - P_x\right)^2 + \left(m\frac{P_x + mP_y - mc}{m} + c - P_y\right)^2} \qquad (1)$$
$$L : y = mx + c,$$

where d stands for the distance from point P to line L, P_x and P_y are the x and y coordinates of the obstacle P, respectively, and m stands for the slope of the line and c is a constant.

In the case of implementing an interval line using a point and a box, the first aforementioned case, one has;

$$m = \frac{A_y - [B_y]}{A_x - [B_x]}, \qquad c = A_y - mA_x, \qquad (2)$$

in which A_x and A_y stand for the coordinates of point A, $[B_x]$ and $[B_y]$ stand for the components of box $[B]$. These lines are referred to as the *collision lines*, which are assigned to the links of mechanisms and in some positions, may interfere with the obstacle.

Resorting to interval functions [8], one can apply interval variables to Eq. (1) and obtain an interval of distances, $[d]$. For example, in the case of the 3-RPR PM for the first limb, the collision line passes through A_1, which is a fixed point, and a box in

Algorithm 1 The pseudo-code of the algorithm to obtain the collision-free workspace of a 3-RPR PPM. Lines followed by % are comments. $\bigvee$ and $\bigwedge$ stand for logical OR and logical AND, respectively.

Input: Design parameters of the 3-RPR PPM; P_x, P_y and P_r; $[B]$ as 2D search space; $\rho_{\min}$ and $\rho_{\max}$ as mechanical strokes; ε as the desired accuracy
Output: $\mathscr{L}_{in}$ as the constant-orientation collision-free workspace of the 3-RPR PPM, $\mathscr{L}_{out}$ as boxes, which are outside the aforementioned workspace.
$\mathscr{L}(1) = [B]$ % Position of the EE;
while `IsEmpty`$(\mathscr{L}) \neq 1$ **do**
  $[B_1] = \mathscr{L}(1)$; % Position of box $[B_1]$ in the fixed frame O_{xy}
  $[B_2] = \mathscr{L}(1) + (\mathbf{b}_2 - \mathbf{b}_1)_{O_{xy}}$; % Position of box $[B_2]$ in the fixed frame O_{xy}
  $[B_3] = \mathscr{L}(1) + (\mathbf{b}_3 - \mathbf{b}_1)_{O_{xy}}$; % Position of box $[B_3]$ in the fixed frame O_{xy}
  for i from 1 to 3 **do**
    $[\rho_i] = \|A_i - [B_i]\|$; % Length of prismatic actuator ρ_i
    $[d_i] =$ `Distance(Line(`$A_i, [B_i]$`),` P`)`; % Distance from P to the line passing through A_i and $[B_i]$
  end for
  $[t_1] =$ `Distance(Line(`$[B_1], [B_2]$`),` P`)`;
  $[t_2] =$ `Distance(Line(`$[B_2], [B_3]$`),` P`)`;
  $[t_3] =$ `Distance(Line(`$[B_3], [B_1]$`),` P`)`;
  if $\rho_{\min} < [\rho_{1,2,3}] < \rho_{\max} \bigwedge (P_r < \overline{d}_{1,2,3} \bigvee \|A_{1,2,3} - [B_{1,2,3}]\|_2 < \|A_{1,2,3} - P\|_2) \bigwedge P_r < \overline{t}_{1,2,3}$ **then**
    $\mathscr{L}_{in} \longleftarrow [B_1]$
  else if $(\rho_{\max} < [\rho_1] \bigvee \rho_{\max} < [\rho_2] \bigvee \rho_{\max} < [\rho_3]) \bigvee \ldots$
  $([\rho_1] < \rho_{\min} \bigvee [\rho_2] < \rho_{\min} \bigvee [\rho_3] < \rho_{\min}) \bigvee \ldots$
  $(\overline{d}_1 < P_r \bigwedge \|A_1 - [B_1]\|_2 > \|A_1 - P\|_2) \bigvee \ldots$
  $(\overline{d}_2 < P_r \bigwedge \|A_2 - [B_2]\|_2 > \|A_2 - P\|_2) \bigvee \ldots$
  $(\overline{d}_3 < P_r \bigwedge \|A_3 - [B_2]\|_2 > \|A_3 - P\|_2) \bigvee \ldots$
  $\overline{t}_1 < P_r \bigvee \ldots$
  $\overline{t}_2 < P_r \bigvee \ldots$
  $\overline{t}_3 < P_r$ **then**
    $\mathscr{L}_{out} \longleftarrow [B_1]$
  else if `Size(`$[B_1]$`)` $> \epsilon$ **then**
    $\mathscr{L}(\text{end} - 1, \text{end}) \longleftarrow$ `Bisect(`$[B_1]$`)` % Bisect $[B_1]$ by the largest edge and add two new boxes at the end of $\mathscr{L}$
  end if
  `ShiftLeft(Empty(`$\mathscr{L}(1)$`))` % Erase data of $\mathscr{L}(1)$ and shift one cell to the left
end while

the search space of the interval algorithm. The task of the algorithm is to determine the distance from obstacle P to the collision line. If $[d] > P_r$, i.e., the distance of all possible lines passing through the fixed point and the box under investigation will be higher than the obstacle radius, then the corresponding box will be fully inside the collision-free workspace. On the other hand, if $[d] < P_r$, the box will be fully outside the aforementioned workspace. Eventually, if $0 \in [d] - P_r$ the box goes for further bisection. The result of the above procedure is illustrated in Fig. 2. Noteworthy, the green circles in figures correspond to the lower limitation of the actuators and even in the case of no obstacle, will appear in workspace.

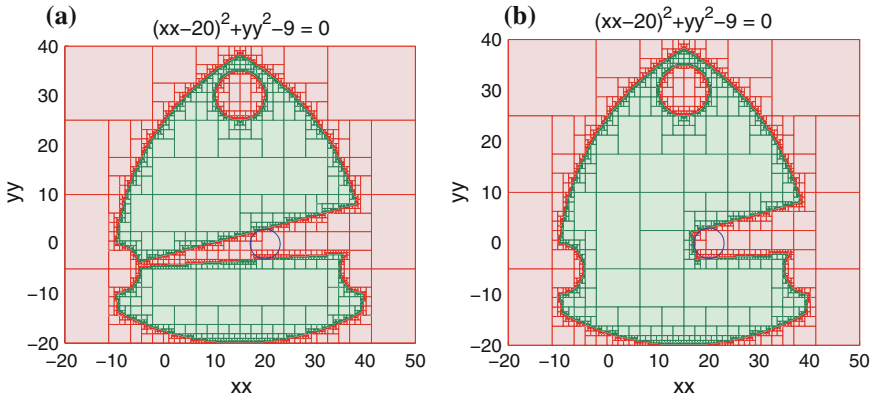


Fig. 2 The collision-free workspace of a 3-RPR PPM shown in Fig. 1a, for $\phi = 0$, only considering the first limb A_1B_1. *Green boxes* are inside and *red boxes* are outside the collision-free workspace. **a** A_1B_1 as a line satisfying $[d] > P_r$. **b** A_1B_1 as a segment line satisfying $[d] > P_r$ and $\|A - [B]\|_2 < \|A - P\|_2$

The procedure of obtaining the collision-free workspace is not complete yet. Indeed, the RPR limb must be regarded as a segment instead of a line. Therefore, if $\|A - [B]\|_2 < \|A - P\|_2$, then $[B]$ does not interfere the obstacle and should be a member of collision-free workspace, Fig. 2a.

In addition to the limbs, the interference between the edges of the EE and the obstacle should also be taken into account. In the case of 3-DOF planar mechanisms, a general EE can be regarded as a triangle. Therefore, its edges belong to those lines passing through two intervals, i.e., the second case. In this case, obtaining the distance from the obstacle to the line, using Eq. (1), leads to a very time consuming and inefficient process. Hence, a geometrical methodology is proposed in order to eliminate those parts of the workspace for which edges of the EE collide with the obstacle. Figure 3a illustrates the results of the foregoing methodology considering only the collision of edge B_1B_2 with the obstacle, for $\phi = 0$. In this figure, a rounded-end rectangle inside the workspace is eliminated. This shape is generated by shifting the obstacle circle through the segment line that connects B_1 to B_2. Practically, the center of the circle-shaped obstacle should be regarded as an interval. In the case of 3-RPR PPM shown in Fig. 1a, $[C_x] = P_x - [0, l_x \cos\phi]$ and $[C_y] = P_y - [0, l_y \cos\phi]$ are the x and y Cartesian coordinates of the center of the shifted obstacle, respectively. Upon considering all the three edges of the EE, Fig. 3b represents the collision-free workspace of the 3-RPR PPM for $\phi = \pi/4$.

Algorithm 1 represents the pseudo-code of the introduced method. It is based on a *branch and prune* algorithm [7]. In lines 5–7, the position of the three distal joints are determined as intervals in the fixed frame. In line 9, for all limbs, the distance from the distal to the proximal joints, i.e., the length of the actuator, is evaluated in order to be checked in line 15. In line 10, `Line`(*A, B*) is a function that creates a collision line that passes through points A and B. Since the second argument is an interval $[B_i]$, hence the collision line is an interval line. `Distance`(*L, P*) computes the

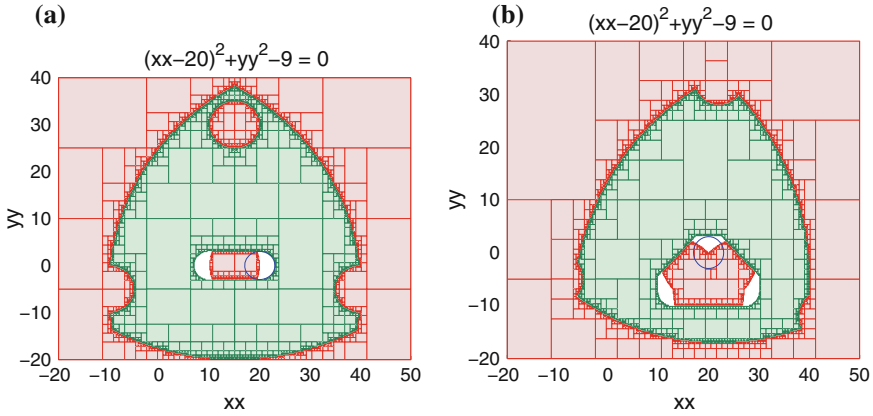


Fig. 3 The collision-free workspace of the mechanism sown in Fig. 1a by considering interval segment lines passing through two intervals. *Green boxes* are inside and *red boxes* are outside the collision-free workspace. **a** The case for which only B_1B_2 collides with the obstacle, $\phi = 0$. **b** Regarding all edges of the EE, i.e. B_1B_2, B_2B_3 and B_3B_1, $\phi = \pi/4$

Euclidean distance from point P to line L. Therefore, $[d_i]$, $i = 1, 2, 3$ are intervals of possible distances from P to the corresponding line of the ith actuator. Lines 12–14 are interval lines that passing through two intervals. Line 15 is the general if-clause for which if all $[\rho_i]$ are in the acceptable range, and if all $[d_i]$ distances from P to the collision lines are higher than P_r, then the box under study will be a member of the collision-free workspace, $\mathscr{L}_{in}$. There is an extra condition to ascertain that if the distance from the box $[b_i]$ to the fixed point A_i is lower than the distance from P to A_i, hence the box should be inside the collision-free workspace. On the other hand, if only one of the aforementioned criteria violates entirely, then the box will be moved to the outer boxes list, $\mathscr{L}_{out}$. In line 19, if the box under study is partially inside the collision-free workspace and at least one of its dimensions is still larger than the desired precision, then it will be bisected by the largest edge and two new boxes will be added at the end of list $\mathscr{L}$. The algorithm continues until the prescribed precision is reached.

3 Results

So far, an interval-based method to obtain the collision-free workspace of planar parallel mechanisms, for a 3-RPR PPM as a case study, has been introduced and the results of collision-free workspace, by considering only the collision of one limb, Fig. 2b, and only considering the collision of the edges of the EE, Fig. 3b, are depicted. The next step is to put together all limbs collision-free workspaces. Figure 4a represents the collision-free workspace of the mechanism for $\phi = \pi/4$, by considering the collisions all limbs with the obstacle. It should be noted that in Fig. 4a, the obstacle is only inside the collision space of the first limb and as it

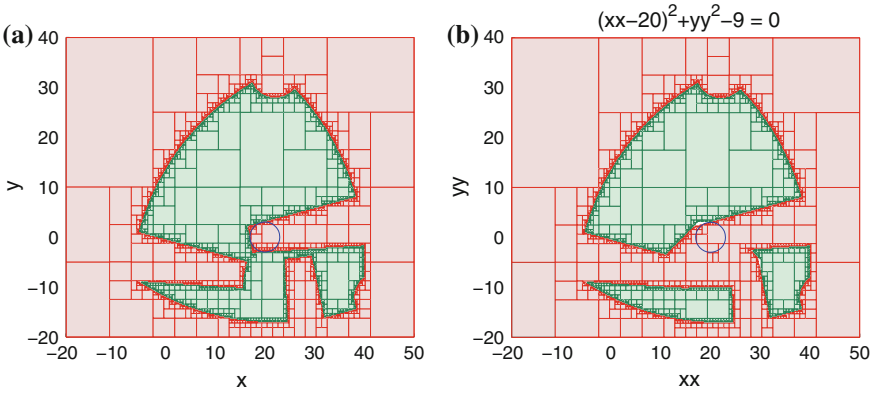


Fig. 4 The collision-free workspace of the 3-RPR PPM shown in Fig. 1a, for $\phi = \pi/4$, **a** only considering limbs **b** final result considering all collisions. *Green boxes* are inside and *red boxes* are outside the collision-free workspace. **a** Only limbs collisions are taken into account, $\phi = \pi/4$, **b** Intersection of all collision-free workspaces including all limbs and all edges of the EE

can be observed, for the second and third limbs the obstacle is located outside the corresponding red boxes. The latter is due to the fact that the represented workspace is depicted in the fixed coordinate frame and since a constant-orientation of the EE is considered, thus other limbs should be translated into the fixed coordinate frame via the transformation of the moving frame. For instance, in the case of the 3-RPR PPM for $\phi = \pi/4$, if we move the obstacle by the vector of $-\{l_x \cos(\pi/4), l_y \sin(\pi/4)\}^T$, which is the negative of position of joint B_2 represented in the moving frame, the obstacle will be located in the red boxes caused by considering the collision of the second limb and the obstacle.

The final step to obtain the collision-free workspace is to merge the results obtained and intersect them, i.e., considering the limb collisions, Fig. 4a, and all edges of the EE, Fig. 3b, with the obstacle. The final result is illustrated in Fig. 4 and as it could be expected, the obstacle divides the workspace into three separated parts, which are not connected to each others. The obtained collision-free workspace can be used in obstacle avoidance problems. As it is obtained via interval analysis, for all paths, which their points are located inside the collision-free workspace, it is guaranteed that the paths are free of obstacle collision.

The introduced method to obtain the collision-free workspace for planar parallel mechanisms can be readily extended to more complicated and spatial PMs. Indeed, one can solve the inverse kinematic problem and use the obtained equations to determine the interval position of all joints and define the aforementioned segment lines to obtain the distances to the obstacle. It takes approximately 3 min to compute the collision-free workspace of the 3-RPR PPM shown in Fig. 1a, precision of 10^{-6} percent of initial search space, with a 2 GHz processor, using INTLAB 6 toolbox. In the case of higher degree of freedom mechanisms having more links, collision computation of each limb will be added into the computational time.

4 Conclusion

In this chapter, an interval-based methodology was introduced in order to obtain the collision-free workspace of planar parallel mechanisms. The proposed approach for the collision-free workspace and the corresponding results were represented for different cases and orientations of the end-effector. First, the collisions of proximal segment lines, which pass through one point and one interval, with the obstacle were investigated. Then, a more general case in which the segment line passes through two intervals, was used to determine the collision of the obstacle with medial and distal limbs, and also edges of the EE. The proposed method is applicable for higher DOF mechanisms and spatial parallel robots. Ongoing work deals with the collision detection of limbs between themselves into the algorithm used for the computation of collision-free workspace of parallel mechanisms.

References

1. Brock, O., Khatib, O.: Elastic strips: a framework for motion generation in human environments. Int. J. Robot. Res. **21**(12), 1031–1052 (2002)
2. Brooks, R.A.: Planning collision-free motions for pick-and-place operations. Int. J. Robot. Res. **2**(4), 19–44 (1983)
3. Caro, S., Chablat, D., Goldsztejn, A., Ishii, D., Jermann, C.: A branch and prune algorithm for the computation of generalized aspects of parallel robots. In: Milano, M. (ed.) Principles and Practice of Constraint Programming, pp. 867–882. Springer, Berlin (2012)
4. Jiménez, P., Thomas, F., Torras, C.: 3d collision detection: a survey. Comput. Graphs. **25**(2), 269–285 (2001)
5. Khatib, O., Yokoi, K., Brock, O., Chang, K., Casal, A.: Robots in human environments: basic autonomous capabilities. Int. J. Robot. Res. **18**(7), 684–696 (1999)
6. Komainda, A., Hiller, M.: Control of heavy load manipulators in varying environments. In: Proceedings of IAARC/IFAC/IEEE International Symposium on Automation and Robotics in Construction, pp. 22–24. Madrid (1999)
7. Merlet, J.: Interval analysis and robotics. In: kaneko, M., Nakamura, Y. (eds.) Robotics Research, pp. 147–156. Springer, Berlin (2011)
8. Merlet, J.P.: Parallel Robots. Springer (2006)
9. Moore, R., Bierbaum, F.: Methods and Applications of Interval Analysis, vol. 2. Society for Industrial Mathematics (1979)
10. Wenger, P., Chedmail, P.: On the connectivity of manipulator free workspace. J. rob. syst. **8**(6), 767–799 (1991)
11. Yang, S.X., Meng, M.: Neural network approaches to dynamic collision-free trajectory generation. IEEE Trans. Syst. Man Cybern. B Cybern. **31**(3), 302–318 (2001)

Development of a One Degree of Freedom Mechanical Thumb Based on Anthropomorphic Tasks for Grasping Applications

Shramana Ghosh and Nina Patarinsky Robson

Abstract Our preliminary work on kinematic synthesis of serial chains shows that the curvature requirements of bodies in contact yield geometric task constraints on position, velocity and acceleration to yield design equations that can be solved to determine the dimensions of a mechanical chain. In this chapter, we show a systematic method for using these higher order motion constraints to formulate the synthesis equations for the design of a planar four bar linkage to be used as a thumb of an underactuated mechanical hand, based on an anthropomorphic task. The thumb will be combined with one degree of freedom multi-loop index and middle fingers to form an underactuated robotic hand for object grasping.

Keywords Kinematic synthesis · Planar linkages · Higher-order constraints · Robotic grasping

1 Introduction

Design of underactuated robotic hands has drawn a lot of interest in the recent years as they offer numerous advantages in terms of lower cost, size, weight and complexity over fully actuated robotic hands. Humanoid robotic hands that can perform challenging tasks like stable grasping of irregular objects, precision assembly etc., such

S. Ghosh (✉)
Mechanical and Aerospace Engineering, University of California, Irvine, CA, USA
e-mail: shramang@uci.edu

N. P. Robson
Mechanical Engineering, California State University, Fullerton, CA, USA
e-mail: nrobson@fullerton.edu; robson@entc.tamu.edu

N. P. Robson
Manufacturing and Mechanical Engineering Technology, Texas A&M University, College Station, TX, USA

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_35,

as the Stanford/JPL HND [1], Utah/MIT Hand [2], DLR-II Hand [3], Gifu II Hand [4], Shadow Dexterous Hand [5], require the use of a substantial number of actuators, resulting in complicated, expensive and bulky devices. Thus, there has been increasing interest in underactuated hand designs which are particularly well-suited for grasping applications, as under certain conditions the unconstrained freedoms result in passive adaptibilty of the fingers around the object to be grasped. Some prominent examples of underactuated hands are the SARAH Hand [6], Barett Hand [7], LARM Hand [8]. While underactuated hands that use tendon-based actuators are compact in size and dexterous in operation, they suffer from friction and elasticity related issues [9] and are generally limited to small grasping forces. For applications where high stability and large grasping forces are expected, linkage based mechanisms can be used, as demonstrated by the mechanical gripper designed by Gosselin and Laliberte [10] and the TUAT/Karlsruhe Hand [11]. However, there is little reported regarding the methods used to design these linkages.

The mobility of bodies in contact has been studied using first order theories that are based on notions of instantaneous force and velocities [12]. For example, Ohwovoriole and Roth describe the relative motion of contacting bodies in terms of Screw Theory. Using first order notions, Reuleaux [13], Markenskoff et al. [14] and Mishra et al. [15], derive bounds on the number of frictionless point contacts required for force closure, which is one means to immobilize an object. However, first order theories are inadequate in practice. The source of deficiency is that the relative mobility of an object in contact with finger bodies is not an infinitesimal notion but a local one. When developing a systematic methodology for the design of multi-fingered robotic hands, one must consider the local motions of the object and not the tangential aspects of the motions, as employed by the first-order theories. This calls for innovative ways to look at the design problems and formulate them in an effective manner such that the whole design space can be obtained.

The synthesis of planar linkages for velocity and acceleration constraints was formulated about 40 years ago by Tesar [16] and Dowler et al. [17]. The concept of using relative curvature of surfaces in contact to limit the movement of a workpiece was introduced by Rimon and Burdick [18, 19]. They generalized the study of the grasping constraint of a rigid body using the fingers of a mechanical hand by considering the configuration space of movement of the body relative to obstacles formed by the fingers, and introduced the idea of second order mobility of a constrained body. Currently [20–22], Robson and McCarthy, and later on Robson and Tolety [23] developed the theoretical framework for the geometric design of planar and spatial serial mechanical linkages to guide an end-effector that maintains contact with specified objects in the environment. They show that the contact geometry defines velocity and acceleration specifications for the movement of the end-effector that are used to formulate synthesis equations for the serial chain.

In this chapter we discuss how to use these higher order effects to formulate the synthesis equations of a planar one-degree-of-freedom four-bar linkage used for the design of a mechanical thumb. The addition of requirements for the accelerations of the fingertips allows for (1) consideration of the local motions of the tip of the

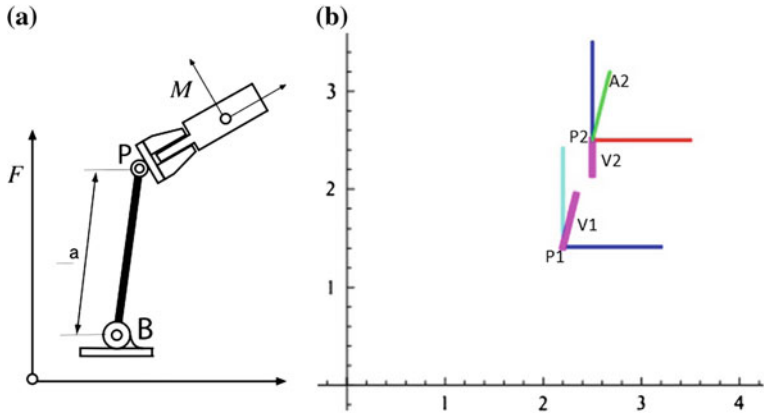


Fig. 1 **a** General structure of a planar RR chain and the five design parameters **B**, **P** and a. **b** Example of a typical task specification for the synthesis of an RR chain: P1 and P2 indicate position specifications, V1 and V2 velocity and A2 acceleration specification

thumb and a grasped object, (2) accurate definition of the anthropomorphic tasks in the vicinity of specified positions.

2 Kinematic Synthesis of a Planar Four Bar Linkage with Second Order Kinematic Constraints

A four-bar type of linkage is chosen to be used for the thumb design based on the kinematic structure of the human's thumb, as well as the advantages of low actuation, high grasping forces and stability that the parallel linkages offer. In order to synthesize the planar 4R chain, we formulate and solve the design equations for a planar RR serial chain. The planar RR chain, shown in Fig. 1a, plays a major role in the planar linkage synthesis, because it is the building block of many open and closed loop planar robot manipulators. The analytical solution of the RR likage design equations yield zero, two, or four sets of real values for the five design parameters for the chain that ensure that the floating link/finger tip moves through the task positions. When two of these chains are connected in parallel, the workspace of the system reduces from two to one dimension to form a single degree of freedom 4R chain.

In what follows, we briefly describe our recent results on kinematic synthesis of a planar RR chain with second order task specifications. As a next step, we present the geometric design of a one degree of freedom mechanical thumb for grasping applications, based on an anthropomorphic task obtained from a motion capture system and a sensor-based glove device.

2.1 Kinematics of a Planar Serial RR Chain

Assume that the planar task consists of positioning the tip of the thumb at a start and an end position which are given by P_1V_1 and $P_2V_2A_2$ (see Fig. 1b).

Let the the movement of the tip of the thumb be defined by the parameterized set of 3×3 homogeneous transforms $[T(t)] = [R(t), \mathbf{d}(t)]$ constructed from a rotation matrix, $R(t)$, and translation vector $\mathbf{d}(t)$. A point $\mathbf{p}$ fixed in the moving body traces a trajectory $\mathbf{P}(t)$ in a fixed coordinate frame F, given by

$$\begin{Bmatrix} P_x(t) \\ P_y(t) \\ 1 \end{Bmatrix} = \begin{bmatrix} \cos\phi(t) & -\sin\phi(t) & d_x(t) \\ \sin\phi(t) & \cos\phi(t) & d_y(t) \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} p_x \\ p_y \\ 1 \end{Bmatrix}, \tag{1}$$

or

$$\mathbf{P}(t) = [T(t)]\mathbf{p}. \tag{2}$$

Our goal is to determine the movement $[T(t)]$. The movement of M relative to a world frame F in the vicinity of a reference position, definoted by $t = 0$ can be expressed as the Taylor series expansion,

$$[T^j(t)] = [T_0^j] + [T_1^j]t + \frac{1}{2}[T_2^j]t^2 + \ldots, \quad j = 1, \ldots, n$$
$$\text{where} \quad [T_i^j] = \frac{d^i[T^j]}{dt^i}\Big|_{t=0}. \tag{3}$$

The matrices $[T_0^j]$, $[T_1^j]$ and $[T_2^j]$ are defined by the position, velocity and acceleration of the end-effector in the vicinity of the two task positions M^j. Therefore, a point $\mathbf{p^j}$ in M has the trajectory $\mathbf{P^j}(t)$ defined by the equation

$$\mathbf{P^j}(t) = [T^j(t)]\mathbf{p^j} = [T_0^j + T_1^j t + \frac{1}{2}T_2^j t^2 + \cdots]\mathbf{p^j}. \tag{4}$$

Let $\mathbf{p^j} = [T_0^j]^{-1}\mathbf{P^j}$, which yields

$$\begin{aligned} \mathbf{P^j(t)} &= [T^j{}_0 + T^j{}_1 t + \frac{1}{2}T^j{}_2 t^2 + \cdots][T^j{}_0]^{-1}\mathbf{P^j}, \\ &= [I + \Omega^j t + \frac{1}{2}\Lambda^j t^2 + \cdots]\mathbf{P^j}, \end{aligned} \tag{5}$$

where

$$[\Omega^j] = \begin{bmatrix} 0 & -\phi_1 & d_{x1} + d_{y0}\phi_1 \\ \phi_1 & 0 & d_{y1} - d_{x0}\phi_1 \\ 0 & 0 & 0 \end{bmatrix}, \quad [\Lambda^j] = \begin{bmatrix} -\phi_1^2 & -\phi_2 & d_{x2} + d_{x0}\phi_1^2 + d_{y0}\phi_2 \\ \phi_2 & -\phi_1^2 & d_{y2} + d_{y0}\phi_1^2 - d_{x0}\phi_2 \\ 0 & 0 & 0 \end{bmatrix} \tag{6}$$

are the planar velocity and planar acceleration matrices, which are defined by the velocity and acceleration specifications of the tip of the thumb in the vicinity of the task positions M_j, $j = 1, \ldots, n$.

2.2 The Constraint Equations and Solutions

The design parameters for the RR chain are the coordinates $\mathbf{B} = (B_x, B_y)$ of the fixed pivot, the coordinates $\mathbf{P^1} = (P_x, P_y)$ of the moving pivot when the floating link is in the first position, and the length a of the link (see Fig. 1). Notice that in each task position the moving pivot $\mathbf{P^j}$ is constrained to lie at the distance a from $\mathbf{B}$, so we have,

$$(\mathbf{P^j}(t) - \mathbf{B}) \cdot (\mathbf{P^j}(t) - \mathbf{B}) = a^2 \tag{7}$$

The derivatives of this equations provide the velocity constraint equation

$$\frac{d}{dt}\mathbf{P^j} \cdot (\mathbf{P^j} - \mathbf{B}) = 0 \tag{8}$$

and the acceleration constraint equation

$$\frac{d^2}{dt^2}\mathbf{P^j} \cdot \left(\mathbf{P^j} - \mathbf{B}\right) + \left(\frac{d}{dt}\mathbf{P^j}\right) \cdot \left(\frac{d}{dt}\mathbf{P^j}\right) = 0 \tag{9}$$

In order to determine the five design parameters, we require five design equations. Choosing one of the task positions to be the first and using the relative displacement matrices $[D_{1j}] = [T^j{}_0][T^1{}_0]^{-1}$ allow us to define coordinates $\mathbf{P^j}$ taken by the moving pivot as follows:

$$\mathbf{P^j} = [D_{1j}]\mathbf{P^1} \tag{10}$$

We now substitute $\mathbf{P^j}$ in Eq. (7) to obtain,

$$\left([D_{1j}]\mathbf{P^1} - \mathbf{B}\right) \cdot \left([D_{1j}]\mathbf{P^1} - \mathbf{B}\right) = a^2, \quad i = 1, \ldots, n \tag{11}$$

These are the position design equations. Note, that $[D_{11}]$ is the 3×3 identity matrix. From our definition of the 3×3 velocity matrix, we have $\frac{d}{dt}\mathbf{P^j} = [\Omega^j][D_{1j}]\mathbf{P^1}$ and substituting $\mathbf{P^j}$ into (8), we obtain the velocity design equations

$$\left([\Omega^j][D_{1j}]\mathbf{P^1}\right) \cdot \left([D_{1j}]\mathbf{P^1} - \mathbf{B}\right) = 0, \quad j = 1, \ldots, n \tag{12}$$

From our definition of the 3×3 acceleration matrix, we have $\frac{d^2}{dt^2}\mathbf{P^j} = [\Lambda_j][D_{1j}]\mathbf{P^1}$ and substituting $\mathbf{P^j}$ in Eq. (9) yields

$$\left([\Lambda_j][D_{1j}]\mathbf{P^1}\right)\cdot\left([D_{1j}]\mathbf{P^1}-\mathbf{B}\right)+\left([\Omega^j][D_{1j}]\mathbf{P^1}\right)\cdot\left([\Omega^j[D_{1j}]\mathbf{P^1}\right)=0,\quad j=1,\ldots,n \tag{13}$$

These are the acceleration design equations. Thus, for each of the n task positions, the position, velocity and acceleration design equations have the following form

$$\begin{aligned}
\mathscr{P}_j &: \left([D_{1j}]\mathbf{P^1}-\mathbf{B}\right)\cdot\left([D_{1j}]\mathbf{P^1}-\mathbf{B}\right)=a^2,\\
\mathscr{V}_j &: \left([\Omega^j][D_{1j}]\mathbf{P^1}\right)\cdot\left([D_{1j}]\mathbf{P^1}-\mathbf{B}\right)=0,\\
\mathscr{A}_j &: \left([\Lambda^j][D_{1j}]\mathbf{P^1}\right)\cdot\left([D_{1j}]\mathbf{P^1}-\mathbf{B}\right)\\
&\quad+\left([\Omega_j][D_{1j}]\mathbf{P^1}\right)\cdot\left([\Omega_j][D_{1j}]\mathbf{P^1}\right)=0,\quad j=1,\ldots,n.
\end{aligned} \tag{14}$$

The constraint equations (14) can be solved algebraically, or numerically using *Mathematica's* `Nsolve` function (see [22] for more details). The algebraic solution to the set of four bilinear equations for an RR chain is presented in McCarthy [24] for the case of five position synthesis and applies without any changes to the design equations (14).

3 Geometric Design of One Degree-of-Freedom Thumb Based on Anthropomorphic Task

To specify the task, we obtain experimental kinematic motion task data (position, velocity, acceleration specifications) from a subject performing a grasping task using a 3D motion capture system (Vicon, OMG PLC., U.K.), shown in Fig. 2a. Recently, the authors developed a cost-effective sensor based glove, which is used in conjunction with the motion capture system to define directly the accelerations, measured at the tip of the fingers and the thumb.

The glove is shown in Fig. 2b, c and is equipped with tri-axis accelerometers (ADXL 335) at the fingertips in addition to the infra-red (IR) markers neccessary for the motion capture system data. Details on the operation of the device can be found in [25].

During each capture session, a left-handed subject sits at a table such that their arm rests on the table in a relaxed position with a pen placed within their reach, and when prompted grasps it. A smaller object with higher curvature is specifically chosen for the task with the idea of testing later the extent of generalization/applicability of the synthesis method to grasping of bigger objects with different geometries. The obtained thumb trajectory can be seen in Fig. 2d.

For defining the synthesis task, we chose two critical positions on the trajectory (the start and end), in which the local motions aspect is important for the thumb performance. At the start position we define a velocity, in order to keep the finger moving as closely along a tangent as possible to the original trajectory in the vicinity

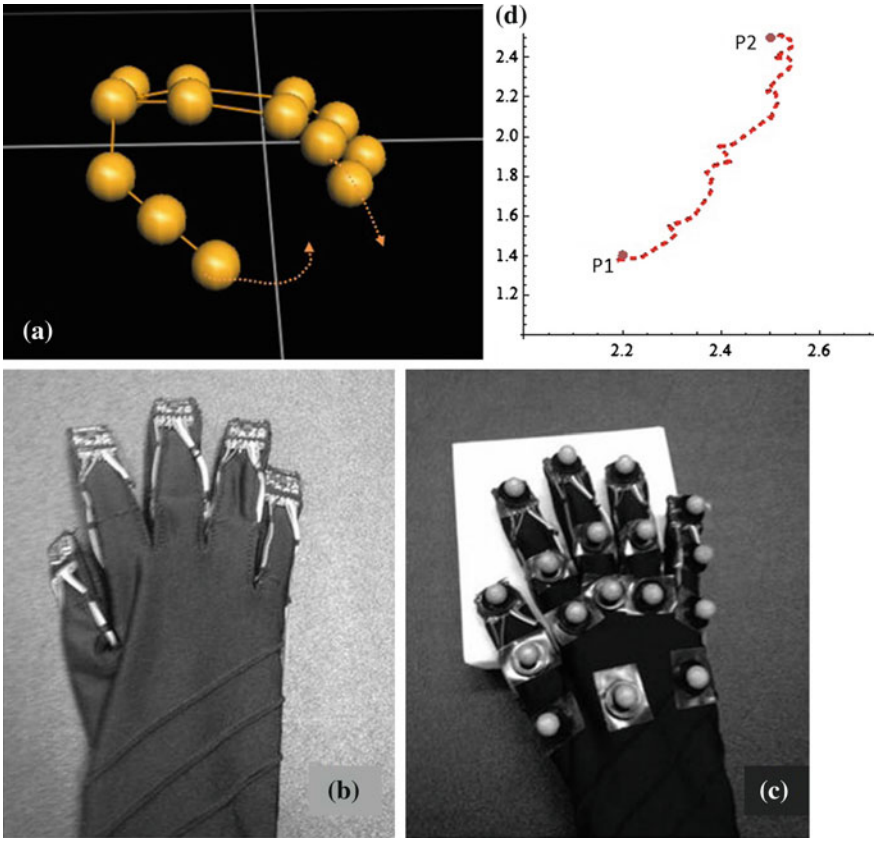


Fig. 2 **a** View of the human hand performing object grasping task. Sensor based glove prototype **b** without and **c** with IR markers. **d** Task trajectory obtained from motion capture system

Table 1 The two positions, velocities and one acceleration, obtained from the anthropomorphic task data from the thumb motion of a left-handed subject performing a pen grasping task

Task	1	2
Position (θ, d_x, d_y)	(0, 2.2, 1.4)	(0, 2.5, 2.5)
Velocity $(\dot{\theta}, \dot{d}_x, \dot{d}_y)$	(1, 0.53, 2.09)	(1, 0.33, −1)
Acceleration $(\ddot{\theta}, \ddot{d}_x, \ddot{d}_y)$	–	(0, 1.18, 4.63)

The positions, velocities and accelerations are expressed in a coordinate frame centered at the MCP joint of the index finger. Lengths are measured in inches, angles in radians and time in seconds

of that position. The second position was defined at the point of contact between the tip of the thumb and the pen. Velocity and acceleration constraints, obtained from the sensor based glove were defined in this position, since the local motions within the vicinity of the contact are critical to the overall performance of the thumb. The task at each of the two selected positions is shown in Table 1.

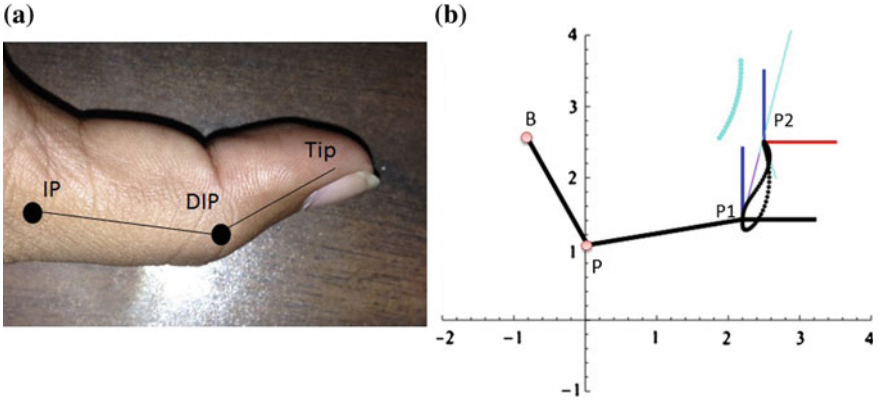


Fig. 3 **a** Kinematic model of thumb as a 2R chain. The • represents a revolute joint. **b** The trajectory of the RR chain, moving from the first to the second position and back

Table 2 The two real solutions for the fixed **B** and the moving pivots **P** (in inches)

Solution	**B**	**P**
1	(−0.82, 2.58)	(0.62, 1.59)
2	(0.56, 1.14)	(0.94, 0.45)

Next, Eq. (14) are applied to synthesize an RR chain denoted by $\mathbf{B_1P_1}$, with a length a within the range of the average distance between the interphalangeal (IP) and distal interphalangeal (DIP) joints for human thumb (see Fig. 3a). The obtained two real solutions from the synthesis of the RR chain are listed in Table 2 and are connected in parallel to form a 4R chain, shown in Fig. 4. The $\mathbf{B_1P_1}$ is obtained to be equal to 1.75 inches and its trajectory is shown in Fig. 3b. Note, that the pick of the trajectory in the vicinity of the second position is where the contact between the thumb and the object occurs and is due to the specified acceleration.

The link length of the second RR chain $\mathbf{B_2P_2}$ is equal to 0.79 inches, which is less than half of the size of $\mathbf{B_1P_1}$ (see Table 2). Last, we animate the linkage to ensure that the tip of the thumb moves smoothly through the given task.

The synthesis results for the planar four-bar mechanical finger moving through the anthropomorphic trajectory is shown in Fig. 4a. The CAD model of the synthesized linkage is shown in Fig. 4b, and a physical model of the thumb linkage is shown in Fig. 4c.

4 Conclusions

The chapter describes a systematic method for the kinematic synthesis of a one degree-of-freedom mechanical thumb to ensure that it does not violate normal direction and curvature constraints, related to velocity and acceleration specifications,

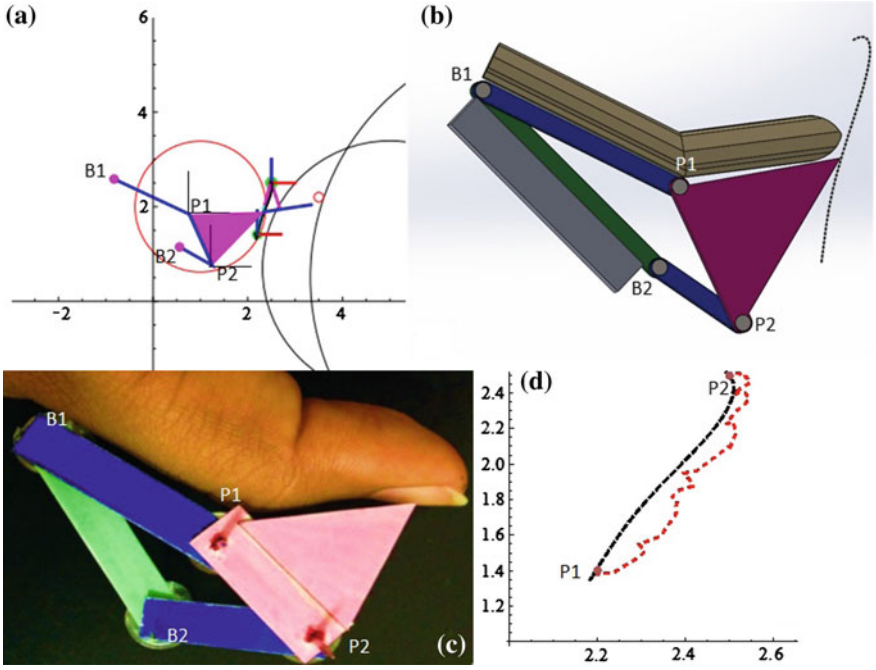


Fig. 4 **a** Result from the synthesis of a one degree of freedom mechanical thumb at its initial position in Mathematica, **b** A CAD model of the synthesized thumb **c** A physical model of the thumb, **d** The - - - represents the experimentally obtained trajectory and ⋯ represents the path trajectory of the synthesized thumb linkage

imposed by contact with objects. The theoretical foundation presented in this chapter serves as a basis for the further design of middle and index fingers for an underactuated mechanical hand and assists in solving some of the open problems in the field, providing results on the synthesis of parallel mechanical chains, based on human's dimensions and anthropomorphic tasks.

Acknowledgments The authors gratefully acknowledge the support of NSF Grant, Award Id: IIS-1208412, sub-award Id: 2013-2908.

References

1. Loucks, C., Johnson, V., Boissiere, P.: Modeling and control of the stanford/JPL hand. IEEE Int. Conf. Robot. Autom. **3**, 573–578 (1987)
2. Jacobsen, S.C., Wood, J.E., Knutti, D.F., et al.: The UTAH/MIT dexterous hand: work in progress. Int. J. Robot. Res. **3**(4), 21–50 (1984)
3. Butterfass, J., Grebenstein, M., Liu, H., Hirzinger G.: DLR-hand II: next generation of a dexterous robot hand. In: IEEE International Conference on Robotics and Automation, pp. 109–114. Seoul (2001)

4. Kawasaki, H., Komatsu, T., Uchiyama, K.: Dexterous anthropomorphic robot hand with distributed tactile sensor: gifu hand II. IEEE/ASME Trans. Mechatron **7**(3), 296–303 (2002)
5. Shadow Robot Company. Shadow Deterous Hand E1 Series—Technical Specification. http://www.shadowrobot.com/wp-content/uploads/shadow_dexterous_hand_technical_specification_E1_20180101.pdf
6. Martin, E., Desbiens, A.L., Laliberte, T., Gosselin C.: SARAH hand used for space operation on STVF robot. In: Intelligent Manipulation and Grasping, pp. 279–284. Genova (2004)
7. William, T.T.: The Barrett Hand grasper—programmably flexible part handling and assembly. Ind. Robot. Int. J. **27**(3), 181–188 (2000)
8. Ceccarelli, M., Jauregui, J.M., Parada, J.E., Nava, N.E., Lanni, C., Carbone, G.: Experimental activity for designing a hand with 1Dof anthropomorphic fingers of human size. In: 12th International Workshop on Robotics in Alpe-Andria-Danube Region RAAD. Cassino (2003)
9. Ceccarelli, M.: Fundamentals of Mechanics of Robotic Manipulation. Springer, Dordrecht (2004)
10. Gosselin, C.M., Laliberte, T.: Underactuated mechanical finger with return actuation. US Patent 5762390 (1998)
11. Fukaya, N., Toyama, S., Asfour, T., Dillmann, R.: Design of the TUAT/Karlsruhe Humanoid Hand. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1754–1759. Takamatsu (2000)
12. Ohwovoriole, M.S., Roth, B.: An extension of screw theory. J. Mech. Des. **103**, 725–735 (1981)
13. Reuleaux, F.: The Kinematics of Machinery. Dover, New York (1963)
14. Markenskoff, X., Ni, L., Papadimitriou, C.H.: The geometry of grasping. Int. J. Robot. Res. **9**, 61–74 (1990)
15. Mishra, B., Schwarz, J.T., Sharir, M.: On the existence and synthesis of multi-finger positive grips. Algorithmica **2**, 541–558 (1987)
16. Tesar, D., Sparks, J.W.: The generalized concept of five multiply separated positions in coplanar motion. J. Mech. **3**(1), 25–33 (1968)
17. Dowler, H.J., Duffy, J., Tesar, D.: A generalized study of four and five multiply separated positions in spherical kinematics-II. Mech. Mach. Theory **13**, 409–435 (1978)
18. Rimon, E., Burdick, J.W.: A configuration space analysis of bodies in contact - I: !-st order mobility. Mech. Mach. Theory **30**(6), 897–912 (1995)
19. Rimon, E., Burdick, J.W.: A configuration space analysis of bodies in contact - II: 2-nd order mobility. Mech. Mach. Theory **30**(6), 913–928 (1995)
20. Robson, N., McCarthy, J.M.: Second order task specifications used in the geometric design of spatial mechanical linkages. Int. J. Mod. Eng. **11**(1), 5–11 (2010)
21. Robson, N., McCarthy J.M., Tumer, I.: Applications of the geometric design of mechanical linkages with task acceleration specifications. In: ASME International Design Engineering Technical Conferences. San Diego, CA (2009)
22. Robson, N., McCarthy, J.M.: Kinematic synthesis with contact direction and curvature constraints on the workpiece. In: Proceedings of ASME International Design Engineering Technical Conferences, vol. 8, pp. 581–588. Las Vegas, NV (2007)
23. Robson, N., Tolety, A.: Geometric design of spherical serial chains with curvature constraints in the environment. In: ASME International Design Engineering Technical Conference. Washington DC (2011)
24. McCarthy, J.M.: Geometric Design of Linkages. Springer, New York (2000)
25. Robson, N., Ghosh, S., Soh, G.S.: Development of a sensor-based glove device for extracting human finger motion data used in the design of minimally-actuated mechanical fingers. In: 3rd IFToMM International Symposium on Robotics and Mechatronics. Singapore (2013)

Trifurcation of the Evolved Sarrus-Motion Linkage Based on Parametric Constraints

Ketao Zhang and Jian S. Dai

Abstract This chapter investigates a single-loop overconstrained mechanism which is an evolved Sarrus-motion linkage extracted from origami-type cardboard model. The mechanism is characterized by bilateral symmetry in the three-dimensional Euclidean space, revealing the mechanism is an interesting case of plane-symmetric Bricard 6R loops. The parametric constraints of this overconstrained 6R mechanism are derived according to its inherent geometry. This mechanism is capable of transforming rotary motion into spherical motion. Constraints analysis in terms of reciprocal screws reveals a unique set of design parameters resulting in constraints degeneration and singularity. This leads to further identification of a very interesting behaviour, trifurcation, of the Sarrus-motion linkage specified by the unique set of design parameters.

Keywords Overconstrained mechanism · Sarrus-motion linkage · Parametric constraints · Trifurcation

1 Introduction

Link-motion, which came to attract a great deal of attention, initially arose in practical problems in various fields such as generation of rectilinear motion in the design of steam engines [1]. It was recognized as a beautiful discovery in link-motion when Peaucellier [2] first reported his solution for generating a straight line and also a conic before G. T. Bennett pointed out that P. F. Sarrus apparatus invented in 1853 [3] is

K. Zhang (✉) and J. S. Dai
Centre for Robotics Research, Kings College London, Strand, London WC2R 2LS, UK
e-mail: ketao.zhang@kcl.ac.uk

J. S. Dai
e-mail: jian.dai@kcl.ac.uk

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics,*
DOI: 10.1007/978-3-319-06698-1_36,

the first mechanism for generating straight-line motion. Starting from the attention called to the Sarrus linkage, a number of overconstrained 6R linkages have been presented. Bricard discovered three different types of mobile 6R-loops during an earlier period beginning from1897 and the other three in 1927 [4, 5]. Schatz [6] discovered the invertible cube due to his art studies, and his discovery of the inversion movement led to the first practical application, Turbula, which was patented in 1971 as one typical overconstrained linkage used in industry [7]. Further study reveals that Schatz linkage can be derived from a special plane-symmetric Bricard loop. Besides the aforementioned inventions of overconstrained 6R mechanisms by mathematicians and engineers, systematic approaches for searching possible overconstrained 6R mechanisms were explored subsequently. Goldberg [8] presented asymmetrical overconstrained mobile 6R linkages by firstly attaching three selected Bennett loops in series. Altmann presented a workable linkage [9] which was revealed as a special case of the Bricard line-symmetric 6R linkage. Waldron obtained a class of overconstrained mechanisms and the six-bar linkage by combining two Bennett linkages [10]. Baker [11] went trough a thorough study of the Bricard 6R-loops and classified these linkages into five distinct cases according to the parametric constraints and closure equations.

In the last two decades, the overconstrained 6R linkage has been extensively studied [12] and their capabilities of being used in deployable structures and foldable devices [13] were explored. A number of works have been reported on the kinematics of overconstrained 6R mechanisms. However, there are very few of them focusing on reconfiguration and motion branches variation of these mechanisms with one degree of freedom (DOF). A very closing work was presented by Chen and Chai on the bifurcated motion of a special line- and plane-symmetric Bricard linkage [14].

When referring to reconfiguration and motion branches variation, there are mainly two kinds of such mechanisms, including kinematotropic mechanisms and metamorphic mechanisms, both of which emerged in the mid-1990s. These two kinds of mechanisms are commonly characterized by their variable mobility in different motion branches and mobility configurations. The kinematotropic linkages [15, 16] were coined as one kind of linkages which are able to change their full-cycle mobility by passing transitory positions [17, 18] where the mechanism experiences singularity. While the metamorphic mechanisms [19] are capable of changing their structure, topological configuration and subsequently mobility resorting to link annex or joint property changes.

The preventative phenomenon of the above two kinds of reconfigurable mechanisms is the mobility change [20]. The subjects of present chapter capture partial features of both kinematotropic mechanisms and metamorphic mechanisms, but not exactly belong to any one of those two kinds as the mobility of the presented mechanism keeps unchanged in different motion branches. Further, the structure of the presented mechanisms is changed in different motion branches and hence the mechanisms are distinguished with the planar four-bar linkage which does not show kinematotropy [15] as well. Here we term the mechanisms with behaviour of reconfiguring their structure but having invariant mobility as RSIM, reconfigurable structure with invariant mobility.

This chapter presents an evolved plane-symmetric Sarrus-motion linkage, especially its special case which can reconfigure the structure by passing transitory position. The parametric constraints and geometric constraints induced trifurcation of the special case is investigated in terms of reciprocal screws. The layout of the chapter is as follows. Section 2 presents the design of an evolved Sarrus-motion linkage and its geometric properties. In the Sect. 3, the parametric constraints and motion characteristics of the linkage are investigated. The geometric constraints induced structure reconfiguration is analysed in details in Sect. 4. The chapter is then concluded in Sect. 5.

2 The Evolved Plane-Symmetric Sarrus-Motion Linkage

Since the concept of mechanisms is often embedded in artistic origami and pop-ups, the origami-inspired mechanical systems and pop-up paper mechanisms [21] attracted substantial interests in recent years. Kinematic chains made from cardboard can be traced back to construction of simple models in three dimensional Euclidean space arisen from Mathematics, the quintessence of truth, by Cundy and Rollett [22]. The typical flat card diagram and corresponding kinematic chain for generating Sarrus motion were presented. Besides the flat card diagram for Sarrus linkage, two more Sarrus-motion models were briefly described. Since the focus of present chapter is originated from and inspired by the models generating Sarrus-motion, the following is going to recall a C-shaped flat card models and present the corresponding closed-loop mechanism.

The C-shaped diagram with creases pattern drawn on a flat card is shown in Fig. 1a. The creases R_1 and R_2 have common point A. The creases R_3 and R_4 are parallel and both of them are perpendicular to R_1. The other creases R_5, R_6 and R_7 are the reflection of R_3, R_2 and R_1 with respect to the axis aligned with crease R_4, respectively. The common point of R_6 and R_7 is denoted as A'. The angles γ_i ($i = 1$ and 2) denote the interior angle of each pair of intersecting creases. The distance between point A and the crease R_3 is denoted by h. The C-shaped cardboard is divided into six parts as panels by these creases. The panels p_1, p_2, p_5 and p_6 are in right trapezoid shape while panels p_3 and p_4 in rectangular shape. According to the bilateral symmetry, panels p_1, p_2 and p_3 are congruent with p_6, p_5 and p_4, respectively, and $\gamma_1 = \gamma_2$. A closed-loop card model is then formed in Fig. 1b by jointing the panels p_1 and p_6 and aligning creases R_1 and R_7 of the C-shaped cardboard in Fig. 1a.

Taking creases as revolute joints and panels as links, an evolved Sarrus-motion linkage is derived in Fig. 2. This 6R linkage is a plane-symmetric mechanism with respect to the plane Π which is formed by the axes of joint R_4 and the common point A. The axes of joints R_1, R_2 and R_6 have the common point A which is the mapping of aligned points A and A' in Fig. 1. The axes of the other three joints, R_2, R_4 and

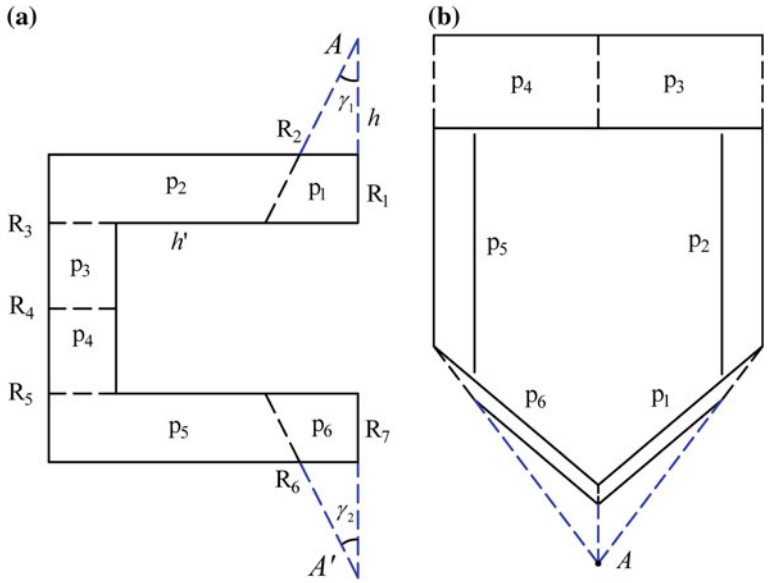


Fig. 1 A pre-grooved C-shaped cardboard and its evolved closed loop [22]. **a** Flat card diagram, **b** Closed-loop model

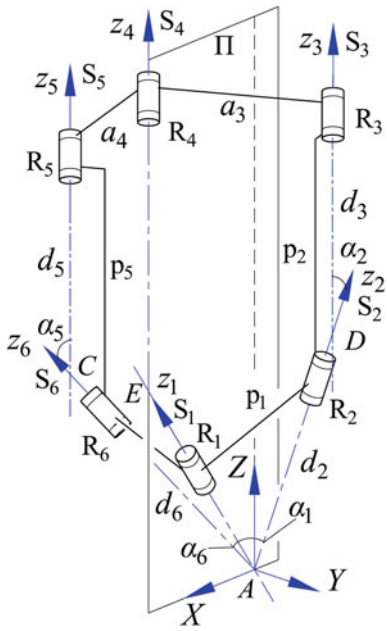


Fig. 2 A Sarrus-motion linkage extracted from the closed-loop card model

R_5 in serials are in parallel. The axes of joints R_2, R_1 and R_6 have common points D, E and C with axes of joints R_3, R_4 and R_5, respectively.

3 Parametric Constraints and Reciprocal Screws

Following the Denavit-Hartenberg convention, a Cartesian coordinate frame is attached to each link with consideration of the geometry of the linkage in Fig. 2. The z_i-axis is aligned with the $(i+1)$th revolute joint axis, and the x_i-axis is defined along the common normal between the ith and $(i+1)$th joint axes, pointing from the ith to the $(i+1)$th joint axis. Here each pair of x_i-axis having origins in symmetric position are either pointing towards or against the plane of symmetry simultaneously.

The parameters of the closed-loop linkage including the offset distance between two adjacent joint axes a_i, the translational distance between two incident normals of a joint axis d_i and the twist angle between two adjacent joint axes α_i are uniquely determined by the geometry of each mechanism. The joint variables of the revolute joints are θ_i.

As illustrated in Fig. 2, the coordinate frame O-XYZ attached at point A is set as the global frame of the 6R mechanism. The Z-axis is parallel to the axes of R_4 and pointing upward, X-axis is located in the plane of symmetry and Y-axis is set following the right-handed rule.

The parametric constraints of the plane-symmetric overconstrained 6R mechanisms is then derived as

$$a_1 = a_3 = a_5 = a_6 = 0, a_3 = a_4 = r \tag{1}$$

$$\alpha_1 = \alpha_6 = \gamma_1, \alpha_2 = \alpha_5 = \gamma - 1 + \pi/2, \alpha_3 = \alpha_4 = 0 \tag{2}$$

$$d_1 = d_4 = 0, d_2 = -d_6 = h/\cos\gamma_1, d_3 = -d_5 = h' \tag{3}$$

in which r, γ_1, h and h' are the design parameters of the mechanism. Further, the value of h' depends on the selection of origin of coordinate frame and can be set as 0. These parametric constraints reveal that the 6R linkage is a case of general plane-symmetric Bricard 6R loop.

The motion screws of the 6R mechanism in Fig. 2 expressed in the global frame O-XYZ are

$$\mathbb{S}_{\mathrm{m1}} = \begin{cases} \mathbf{S}_{11} = [l_1 & 0 & n_1 & 0 & 0 & 0]^{\mathrm{T}} \\ \mathbf{S}_{12} = [l_2 & m_2 & n_2 & 0 & 0 & 0]^{\mathrm{T}} \\ \mathbf{S}_{13} = [0 & 0 & 1 & hm_2 & -hl_2 & 0]^{\mathrm{T}} \\ \mathbf{S}_{14} = [0 & 0 & 1 & 0 & -hl_1 & 0]^{\mathrm{T}} \\ \mathbf{S}_{15} = [0 & 0 & 1 & -hm_2 & -hl_2 & 0]^{\mathrm{T}} \\ \mathbf{S}_{16} = [l_2 & -m_2 & n_2 & 0 & 0 & 0]^{\mathrm{T}} \end{cases} \tag{4}$$

where

$$
\begin{aligned}
l_1 &= \sqrt{1-(s\theta_{10}t\alpha_1)^2}+\sqrt{(r/h)^2-(s\theta_{10}t\alpha_1)^2}, \quad n_1 = t\gamma_1,\\
l_2 &= \sqrt{1-(s\theta_{10}t\alpha_1)^2}, \quad m_2 = s\theta_{10}t\alpha_1 \quad n_2 = t\gamma_1,\\
\theta_{10} &= \theta_1/2.
\end{aligned}
$$

The constraint-screw system is reciprocal to the motion-screw system and can be derived as

$$
\mathbb{S}_1^{\mathrm{c}} = \mathbf{S}_1^{\mathrm{r}} = [0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0]^{\mathrm{T}} \tag{5}
$$

The above constraint screw is a pure force along Z-axis and the dimension of the constraint-screw system, $\dim(\mathbb{S}^{\mathrm{c}})$, equals to one. With this constraint force, the links p_2 and p_5 are rotating around the Z-axis while p_1 and p_6 are implementing spherical motion. This implies the mechanism can transform rotary motion into spherical motion.

4 Constraints Induced Transitory Position and Trifurcation

It is evident that the 6R linkage is in boundary singularity when the axes of each pair of joints located in the symmetric position become collinear. In other words, the boundary singularity occurs when the common points C and D are concurrent. Besides the boundary singularity, the other singular position can be identified by checking the dependency of motion screws.

According to the geometry of the overconstrained mechanism in Fig. 2, the corresponding motion screws in Eq. (4) form a five-system in a general configuration and the screw system only degenerates when the design parameters h and r and the motion parameter θ_1 satisfy the following condition,

$$
\begin{cases}
\gamma_1 = \pi/4,\\
h \;\; = r,\\
\theta_1 = \pi.
\end{cases} \tag{6}
$$

Under such a condition, the motion screws of the linkage become

$$
\mathbb{S}_{\mathrm{m2}} = \begin{cases}
\mathbf{S}_{21} = [0 \quad 0 \quad 1 \quad 0 \quad 0 \; 0]^{\mathrm{T}}\\
\mathbf{S}_{22} = [0 \quad 1 \quad 1 \quad 0 \quad 0 \; 0]^{\mathrm{T}}\\
\mathbf{S}_{23} = [0 \quad 0 \quad 1 \quad r \quad 0 \; 0]^{\mathrm{T}}\\
\mathbf{S}_{24} = [0 \quad 0 \quad 1 \quad 0 \quad 0 \; 0]^{\mathrm{T}}\\
\mathbf{S}_{25} = [0 \quad 0 \quad 1 \quad -r \quad 0 \; 0]^{\mathrm{T}}\\
\mathbf{S}_{26} = [0 \quad -1 \quad 1 \quad 0 \quad 0 \; 0]^{\mathrm{T}}
\end{cases} \tag{7}
$$

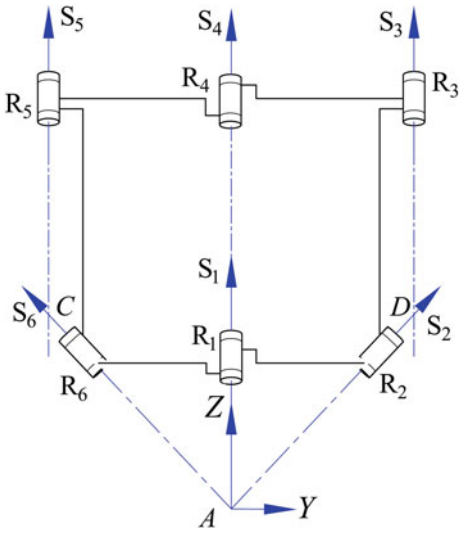


Fig. 3 Transitory position of the Sarrus-motion linkage with special parameters

The constraint screw system, which is reciprocal to the motion screw system, is subsequently derived as

$$\mathbb{S}_2^{\mathrm{c}} = \begin{cases} \mathbf{S}_{21}^{\mathrm{r}} = [1 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0]^{\mathrm{T}} \\ \mathbf{S}_{22}^{\mathrm{r}} = [0 \;\; 1 \;\; 0 \;\; 0 \;\; 0 \;\; 0]^{\mathrm{T}} \\ \mathbf{S}_{23}^{\mathrm{r}} = [0 \;\; 0 \;\; 1 \;\; 0 \;\; 0 \;\; 0]^{\mathrm{T}} \end{cases} \tag{8}$$

The above constraint screws are the bases of the degenerated constraint-screw system of order 3. It implies that the mechanisms moves to a singular configuration in which the axes of all revolute joints are in one plane and the axes of joints R_1 and R_4 are collinear as illustrated in Fig. 3.

According to the geometry of the 6R mechanism and the constraint-screw system at the singular position, the mechanism can move to other two motion branches by passing this singular position as a transitory position.

The first motion branch in Fig. 4a is a spherical 4R linkage. In this motion branch, the axis of joint R_4 passes the common point A and the two joints R_3 and R_5 are geometrically locked in the full-cycle motion. The constraint-screw system for this spherical 4R linkage is a three-system.

The second motion branch in Fig. 4b is a planer 4R linkage with equal link length. In this motion branch, the axes of joints R_1 and R_4 keep parallel and the two joints R_2 and R_6 are geometrically locked in the full cycle motion. The constraint-screw system for this planer 4R linkage is a three-system.

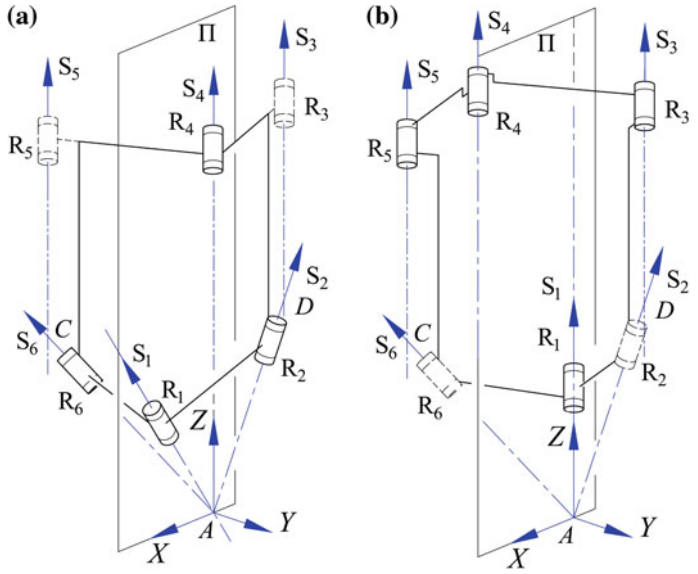


Fig. 4 Two motion branches evolved from the overconstrained 6R linkage **a** *Branch 1* spherical 4R linkage, **b** *Branch 2* planer 4R linkage

5 Conclusions

This chapter presented a Sarrus-motion linkage extracted from a closed-loop cardboard model. The parametric constraints of the overconstrained 6R linkage are derived. It revealed the 6R linkage is capable of transforming rotary motion into spherical motion in a general configuration. The screw system based constraints analysis further revealed the specific parametric constraints leading to inner singularity of the mechanism. The 6R linkage with specific parameters has ability to reconfigure its motion branches and change its configuration to a spherical 4R linkage and a planner 4R linkage with invariant mobility one by passing the transitory position where singularity occurs. This overconstrained 6R linkage can be employed for development of remote centre of motion (RCM) mechanism.

Acknowledgments The authors thank the support of European Commission—FP7 under Grant No. 270436.

References

1. Cajori, F.: A History of Mathematics. American Mathematical Society, Providence (1991)
2. Gerono, C., Peaucellier, M.: Capitaine du genie (nice). Nouvelles annales de mathmatiques 2e srie, **1864**, 414–416 (1864)

3. Sarrus, P.T.: Note sur la transformation des mouvements rectilignes alternatifs, en mouvements circulaires; et reciproquement. Acad. Sci. **36**, 1036–1038 (1853)
4. Bricard, R.: Mmoire sur la thorie de loctaedre articul. J. Mathmatiques Pures Appl. Liouville **3**, 113–148 (1897)
5. Bricard, R.: Leçons de cinématique. Tome II cinématique appliquée, Gauthier-Villars, Paris, 7–12 (1927)
6. Schatz, P.: Rhythmusforschung und Technik. Verlag Freies Geistesleben, Stuttgart (1975)
7. Paul Schatz Stiftung Foundation. http://www.paul-schatz.ch/en/seine-entdeckungen/dieinversina/
8. Goldberg, M.: New five-bar and six-bar linkages in three dimensions. Trans. ASME **65**, 649–663 (1943)
9. Altmann, P.G.: (Communications to Grodzinski, P. and Mewen, E.) Link mechanisms in modern kinematics. Proc. Inst. Mech. Eng. **168**, 889–896 (1954)
10. Waldron, K.J.: Hybrid overconstrained linkages. J. Mech. **3**, 73–78 (1968)
11. Baker, J.E.: An analysis of Bricard linkages. Mech. Mach. Theory **15**, 267–286 (1980)
12. Cui, L., Dai, J.S.: Axis constraint analysis and its resultant 6R double-centered overconstrained mechanisms. J. Mech. Rob. **3**, 031004 (2011)
13. Chen, Y., You, Z.: Square deployable frames for space applications. Part 1: theory. Proc. Inst. Mech. Eng. G: J. Aerosp. Eng. **220**, 347–354 (2006)
14. Chen, Y., Chai, W.: Bifurcation of a special line and plane symmetric Bricard linkage. Mech. Mach. Theory **46**, 515–533 (2011)
15. Wohlhart, K.: Kinematotropic Linkages. Recent Advances in Robot Kinematics. Kluwer, Dordrecht (1996)
16. Calletti, C., Fanghella, P.: Single-loop kinematotropic mechanism. Mech. Mach. Theory **36**, 743–761 (2001)
17. Gogu, G.: Branching singularities in kinematotropic parallel mechanisms. In: Proceedings of the 5th International Workshop on Computational Kinematics, Duisburg, Germany (2009)
18. Zhang, K.T., Dai, J.S., Fang, Y.F.: Constraint analysis and bifurcated motion of the 3PUP parallel mechanism. Mech. Mach. Theory **49**(3), 256–269 (2012)
19. Dai, J.S., Jones, J.R.: Mobility in metamorphic mechanisms of foldable/erectable kinds. 25th ASME Biennial Mechanisms and Robotics Conference, Atlanta, USA (1998)
20. Zhang, K.T., Dai, J.S., Fang, Y.F.: Topology and constraint analysis of phase change in the metamorphic chain and its evolved mechanism. ASME J. Mech. Des. **132**, 121001 (2010)
21. Winder, B.G., Magleby, S.P., Howell, L.L.: Kinematic representations of pop-up paper mechanisms. ASME J. Mech. Rob. **1**, 021009 (2009)
22. Cundy, H.M., Rollett, A.P.: Mathematical Models. Tarquin, Suffolk (1981)

The Kinematics of Containment

Gregory S. Chirikjian and Yan Yan

Abstract This chapter is concerned with two problems related to what we call the kinematics of containment: (1) Given a small convex body in n-dimensional Euclidean space, such as an ellipsoid, that is contained inside of a large convex body, characterize the range of allowable motions for which the boundaries of the bodies do not collide, and calculate the corresponding volume of this motion in the group of rigid-body motions, $SE(n)$; (2) If the smaller body is almost the size of the larger one, and can only execute small collision-free motions, analyze the range of these motions. Both of these problems are addressed fully here. The first uses methods of the fields of Integral Geometry and Geometric Probability and derives an expression similar to the so-called Principal Kinematic Formula. The second uses the kinematics of infinitesimal motions and properties of ellipsoids.

Keywords Rigid-body motion · Minkowski sum · Principal kinematic formula · Ellipsoids · Collision · Containment

1 Introduction

Knowing how much free room to move that a small body has inside a large cavity has applications in the design of mechanical parts-handling machines [3, 11], characterizing the entropy of parts in manufacturing settings [5, 16], and in assessing the entropy of biomolecules inside of cellular environments [6]. This chapter first develops new formulas that parameterize the region of the group of rigid-body motions for the case when one small body is capable of free rotation inside of the larger body. Then, the other extreme case in which small motions of an ellipsoidal body

G. S. Chirikjian (✉) · Y. Yan
Department of Mechanical Engineering, Johns Hopkins University, Baltimore, USA
e-mail: gchirik1@jhu.edu

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_37,

contained in a slightly larger body is analyzed. In both cases, the volume of free motion is computed in closed form.

Throughout this chapter, as a matter of notation $\mathbb{I}_n$ is the $n \times n$ identity matrix consisting of entries of 1 on the diagonal and 0 otherwise, A^T denotes the transpose of a matrix, and A^{-1} denotes the inverse of a square matrix with nonzero determinant, $\det A \neq 0$. The set of $n \times n$ special orthogonal (or proper rotation) matrices is denoted as

$$SO(n) \doteq \{R \in \mathbb{R}^{n \times n} \mid RR^T = \mathbb{I}_n, \det R = +1\}.$$

This set forms a group under the operation of matrix multiplication. In fact, it is an $n(n-1)/2$-dimensional matrix Lie group. The set of all special Euclidean motions (or proper rigid-body motions) in n-dimensional Euclidean space is [1, 4, 14]

$$SE(n) \doteq \{(R, \mathbf{t}) \in SO(n) \times \mathbb{R}^n \mid (R_1, \mathbf{t}_1) \circ (R_2, \mathbf{t}_2) = (R_1 R_2, R_1 \mathbf{t}_2 + \mathbf{t}_1)\}.$$

It is well known that an arbitrary motion can be decomposed into a product of translations and rotations as

$$(R, \mathbf{t}) = (\mathbb{I}_n, \mathbf{t}) \circ (R, \mathbf{0}). \tag{1}$$

The group operation $\circ$ can be captured by representing motions as $(n+1) \times (n+1)$ homogeneous transformation matrices. If $g = (R, \mathbf{t}) \in SE(n)$, the notation $g \cdot \mathbf{x} = R\mathbf{x} + \mathbf{t}$ for any $\mathbf{x} \in \mathbb{R}^n$ defines an action of $SE(3)$ on $\mathbb{R}^n$. And similarly, if $B \subset \mathbb{R}^n$ is a finite rigid body, then $g \cdot B$ is the moved version of B defined as

$$g \cdot B = \{g \cdot \mathbf{x} \,|\, \mathbf{x} \in B\}.$$

Let $\chi_B : \mathbb{R}^n \to \mathbb{R}$ be the characteristic function of B which takes a value of 1 when $\mathbf{x} \in B$ and a value of 0 otherwise. It is not difficult to show that

$$\chi_{g \cdot B}(\mathbf{x}) = \chi_B(g^{-1} \cdot \mathbf{x}).$$

If $f \in (L^1 \cap L^2 \cap C^1)(\mathbb{R}^n)$, i.e., $f : \mathbb{R}^n \to \mathbb{R}$ is absolutely and square integrable with respect to the Lesbegue measure $d\mathbf{x} = dx_1 dx_2 \cdots dx_n$, and it is also differentiable with continuous derivative, then $f : \mathbb{R}^n \to \mathbb{R}$ will be called a "nice" or "well-behaved function". For functions $\varphi : SE(n) \to \mathbb{R}$ similar concepts exist as defined in [6] where instead of the Lesbegue measure, the Haar measure $dg = dR d\mathbf{t}$ is used. Here dR is the Haar measure for $SO(n)$ [6, 7] and $d\mathbf{t}$ is the Lesbegue measure for the translational part of the motion. Even though $SE(n)$ is not compact and does not possess a bi-invariant metric, this integration measure is nevertheless bi-invariant [7], and as a result

$$\int_{SE(n)} \varphi(g)\, dg = \int_{SE(n)} \varphi(g^{-1})\, dg = \int_{SE(n)} \varphi(g_0 \circ g)\, dg = \int_{SE(n)} \varphi(g \circ g_0)\, dg$$

where

$$g^{-1} = (R^T, -R^T\mathbf{t}) \quad \text{and} \quad \int_{SE(n)} \varphi(g)\, dg \doteq \int_{\mathbb{R}^n} \int_{SO(n)} \varphi(R, \mathbf{t}) dR d\mathbf{t}$$

and g_0 is an arbitrary constant element of $SE(n)$.

This background material will be used in the following sections. Section 2 reviews the classical Principal Kinematic Formula, giving a new derivation that leads to a new containment formula for convex bodies. Section 3 derives closed-form expressions for the Minkowski sum of ellipsoids. Section 4 relates the formula for the Minkowski sum of convex bodies to the Principal Kinematic Formula. These are used to derive expressions for the allowable motion of small bodies inside of larger ones. Section 5 examines the opposite extreme in which the contained body is almost the same size as the larger one, and computes nontrivial lower bounds on the free room to move in this case.

2 The Principal Kinematic Formula

In classical integral geometry, the Principal Kinematic Formula plays a central role. The formula expresses the average Euler characteristic of the intersections of rigid bodies moving uniformly at random in terms of fundamental quantities of these bodies (volume, area, etc.). When the bodies are convex, the intersections are convex, and the Euler characteristic can be replaced by the set indicator function, $\iota(\cdot)$ which takes a value of 1 when the argument is nonempty, and 0 otherwise. The resulting formula works in $\mathbb{R}^n$ and has been extended to general spaces of constant curvature. But we are concerned only with two- and three-dimensional Euclidean space, in which case the result is as follows.

Theorem 1 (See [2, 17] for proof): *Given convex bodies C_0 and C_1 in $\mathbb{R}^2$, then*

$$\int_{SE(2)} \iota(C_0 \cap g \cdot C_1)\, dg = 2\pi[A(C_0) + A(C_1)] + P(\partial C_0) \cdot P(\partial C_1). \qquad (2)$$

where $A(C_i)$ is the area of planar body C_i and $P(\partial C_i)$ is its perimeter (i.e., the length of the closed curve that bounds it). Here $dg = dt_1 dt_2 d\theta$. For example, if the bodies are circular disks of radii r_0 and r_1, then if the center of C_1 translates anywhere within a circle of radius $r_0 + r_1$ at any orientation, it will be in collision with body C_0 with center pinned at the origin. Therefore, in this case the collision space, as a subset of $SE(2)$, is the Cartesian product of a disk of radius $r_0 + r_1$ and $SO(2)$, i.e., $D_{r_0+r_1} \times SO(2)$. The volume of this domain is easily computed as $2\pi \times \pi(r_0 + r_1)^2$. The formula in (2) gives $2\pi[\pi r_0^2 + \pi r_1^2] + (2\pi r_0)(2\pi r_0)$, which is the same thing. In $\mathbb{R}^3$ the analogous formula is [2, 17]

$$\int_{SE(3)} \iota(C_0 \cap g \cdot C_1)\, dg$$
$$= 8\pi^2[V(C_0) + V(C_1)] + 2\pi[F(\partial C_0)M(\partial C_1) + F(\partial C_1)M(\partial C_0)] \tag{3}$$

where $F(\partial C_i)$ and $M(\partial C_i)$ are respectively the surface area and integral of mean curvature of the bounding surface enclosing body i, and $V(C_i)$ is the volume of body i. To be concrete, if rotations are expressed in terms of ZXZ Euler angles, then the volume element is $dg = \sin\beta d\alpha d\beta d\gamma dt_1 dt_2 dt_3$. These equations generalize to n-dimensional Euclidean spaces and even to other spaces of constant sectional curvature. For the proof and pointers to the literature, see [2, 12, 17, 19].

An alternative proof specifically for convex bodies was given in [6]. In that proof, the center of the moving body, C_1, visits every point in the fixed body, C_0, and rotates freely, each time contributing to the integral, and resulting in the $2\pi V(C_0)$ and $8\pi^2 V(C_0)$ terms in the 2D and 3D cases, respectively. Then, the moving body is decomposed into concentric shells, and as each shell makes every possible point contact with the boundary ∂C_0, intersections of the original bodies is also guaranteed. Adding up these contributions results in the above formulas.

This alternative proof is mentioned, because a new kind of kinematic formula can be derived in essentially the same way. In this new formula, we are concerned not with measuring the volume in $SE(n)$ corresponding to all possible intersections of bodies, but rather the integral of the volume in $SE(n)$ corresponding to all possible ways that C_1 can move while being contained in C_0. To this end, let $b(g \cdot C_1 \subset C_0)$ be defined to take a value of 1 when $g \cdot C_1 \subset C_0$ and a value of zero otherwise, corresponding to the binary truth of the statement that the moving body is contained in the stationary one.

Theorem 2 *Given convex bodies C_0 and C_1 in $\mathbb{R}^n$ such that it is possible to slide ∂C_1 through all orientations while remaining completely inside of C_0 with the the boundaries $g \cdot \partial C_1$ and ∂C_0 always in contact at a single point, then for for $n = 2$*

$$\int_{SE(2)} b(g \cdot C_1 \subset C_0)\, dg = 2\pi[A(C_0) + A(C_1)] - P(\partial C_0) \cdot P(\partial C_1) \tag{4}$$

and for for $n = 3$

$$\int_{SE(3)} b(g \cdot C_1 \subset C_0)\, dg$$
$$= 8\pi^2[V(C_0) - V(C_1)] - 2\pi F(\partial C_0)M(\partial C_1) + 2\pi F(\partial C_1)M(\partial C_0). \tag{5}$$

For example, if C_0 and C_1 are spheres in 3D with radii $r_0 > r_1$, then $F(\partial C_i) = 4\pi r_i^2$, $M(\partial C_i) = 4\pi r_i$, $V(C_i) = \frac{4}{3}\pi r_i^3$, the above 3D formula gives $8\pi^2$ times the volume of a sphere of radius $r_0 - r_1$ (where $8\pi^2$ is the volume of $SO(3)$).

We know of no other work that addresses this problem. The closest works are those of Zhang [21] and Zhou [22] that address when one body can be contained within another (but not the amount of motion allowed for a contained body). In some practical engineering contexts, this can be quite important [3, 5, 11].

3 The Minkowski Sum of Ellipsoids

The results of the previous section can be expressed in terms of the Minkowski sum of two bodies. Recall that the Minkowski sum and difference of two convex point sets (or bodies) each centered at the origin, P_1 and P_2 in $\mathbb{R}^n$, are respectively defined as

$$P_1 \oplus P_2 \doteq \{p_1 + p_2 \mid p_1 \in P_1, p_2 \in P_2\} \text{ and } P_1 \ominus P_2 \doteq \bigcap_{p_2 \in P_2} (P_1 + p_2). \quad (6)$$

Alternatively, the Minkowski difference of two convex bodies can be defined relative to the Minkowski sum as the body $P_1' = P_1 \ominus P_2$ for which $P_1 = P_1' \oplus P_2$.

Interestingly, the boundary of the Minkowski sum of two ellipsoids can be computed in closed form, but appears not to have been reported previously in the literature, despite a vast literature on related work concerned with checking collisions between ellipsoids (see e.g., [13, 15]).

We now present how to compute this Minkowski sum of two arbitrary ellipsoids. To begin, let $\mathbf{u}(\phi)$ be the standard parameterization of the hyper-sphere S^{n-1} with $n-1$ angles $\phi = [\phi_1, \ldots, \phi_{n-1}]$. Note that in addition to the defining fact that $\mathbf{u}^T\mathbf{u} = 1$, the entries of $\mathbf{u}$ must also satisfy the conditions

$$-1 \le u_i \le 1 \;\; ; \;\; 0 \le u_i^2 \le 1 \;\; ; \;\; -1/2 \le u_i u_j \le 1/2 \text{ for } i \ne j. \quad (7)$$

Whereas the first two sets of inequalities are obvious, the third is perhaps less so, and can be proven using induction.

Let E_1 and E_2 be two arbitrary ellipsoids in n-dimensional Euclidean space with semi-axis lengths given by the row vectors $\mathbf{a} = [a_1, \ldots, a_n]$ and $\mathbf{b} = [b_1, \ldots, b_n]$, respectively. In the frame of reference with origin at the center of mass and aligned with the principal axes of E_1, it has implicit and parametric equations of the form

$$\mathbf{x}^T \, \Lambda^{-2}(\mathbf{a}) \, \mathbf{x} \; = \; 1 \quad \text{and} \quad \mathbf{x} = \Lambda(\mathbf{a})\mathbf{u}(\phi) \quad (8)$$

where $\Lambda(\mathbf{a})$ is the $n \times n$ diagonal matrix with entries $\lambda_{ij} = a_i \delta_{ij}$ and $\Lambda^m(\mathbf{a})$ is shorthand for the mth power $[\Lambda(\mathbf{a})]^m$. An ellipsoid of the same shape and same center

of mass but at an arbitrary orientation will have parametric and implicit descriptions of the form $\hat{\mathbf{x}} = R\Lambda(\mathbf{a})\mathbf{u}(\phi)$ and $\hat{\mathbf{x}}^T R\Lambda^{-2}(\mathbf{a})R^T\hat{\mathbf{x}} = 1$ where R is an $n \times n$ rotation matrix.

As with any convex set, the Minkowski sum of E_1 and E_2 is $E_1 \oplus E_2$ as defined in (6). Let E_2 translate around E_1 and attach a reference point in the center of E_2. Then $E_1 \oplus E_2$ can be interpreted as the locus of positions, t, of the reference point of E_2 for which $E_1 \cap (t \cdot E_2) \neq \emptyset$. This can be evaluated over all translations $t \in \mathbb{R}^n$ where $t \cdot E_2$ denotes a translated version of E_2 defined by $t \cdot E_2 = \{x + t | x \in E_2\}$.

The Minkowski sum of E_1 and $R_2 \cdot E_2$ (a rotated version of E_2) can be computed as follows: (1) Apply a linear transformation to the whole scene that preserves the origin of E_1 yet shrinks $R_2 \cdot E_2$ to a sphere; (2) Compute the offset surface of the transformed version of E_1; (3) Apply the inverse of the linear transformation to the whole scene.

Let $r = \min\{b_1, b_2, ..., b_n\}$. The boundary of $E_1 \oplus E_2$ in this case becomes an offset curve/surface. After this affine operation E_1 still remains an ellipsoid but with changed semi-axis lengths $\tilde{\mathbf{a}} = [\tilde{a}_1, \tilde{a}_2, ..., \tilde{a}_n]$ and changed orientation. This new ellipsoid is called $\tilde{E}_1$.

The first linear operation on E_1 transforms it to

$$\tilde{\mathbf{x}} = R_2\Delta^{-1}(\mathbf{b}/r)R_2^T\mathbf{x} \iff \mathbf{x} = R_2\Delta(\mathbf{b}/r)R_2^T\tilde{\mathbf{x}} \tag{9}$$

where $\mathbf{x}$ and $\tilde{\mathbf{x}}$ specify the coordinates of the original E_1 and transformed version, $\tilde{E}_1$, respectively, and R_2 is the rotation matrix describing the orientation of E_2.

The implicit expression for the original version of E_1 is given in (8). By substituting (9) into (8), we can get the implicit expression for $\tilde{E}_1$ of the form $\tilde{\mathbf{x}}^T A^{-2}\tilde{\mathbf{x}} = 1$, where $A = Q\Lambda(\tilde{\mathbf{a}})Q^T$ depends on the rotation matrix R_2, and $\mathbf{a}$ and $\mathbf{b}$. The eigenvalues A_1, which are the entries of $\tilde{\mathbf{a}}$, are the changed semi-axis lengths of E_1, and the rotation matrix Q gives the orientation of its principal axes. In this principal axes frame the parametric and implicit equations of the shrunk motionless ellipsoid are the same as in (8) but with $\mathbf{a}$ replaced by $\tilde{\mathbf{a}}$.

An offset hyper-surface $\tilde{\mathbf{x}}_{ofs}(\phi)$ of an orientable, closed, and differentiable hyper-surface $\tilde{\mathbf{x}}(\phi) \in \mathbb{R}^n$ with the offset radius r is defined as

$$\tilde{\mathbf{x}}_{ofs}(\phi) = \tilde{\mathbf{x}}(\phi) + r\,\tilde{\mathbf{n}}(\phi), \tag{10}$$

where $\tilde{\mathbf{n}}$ is the outward-pointing unit surface normal, and these surfaces are parameterized by $\phi \in \mathbb{R}^{n-1}$. In the case of an ellipsoidal surface defined by the level set $\Phi(\tilde{\mathbf{x}}) \doteq \tilde{\mathbf{x}}^T A^{-2}\tilde{\mathbf{x}} - 1 = 0$, the outward pointing normal can first be computed as $\nabla\Phi(\tilde{\mathbf{x}}) = 2A^{-2}\tilde{\mathbf{x}}$, and then evaluated with the parametric equation and normalized:

$$\tilde{\mathbf{n}}(\phi) = \left.\frac{\nabla\Phi(\tilde{\mathbf{x}}(\phi))}{\|\nabla\Phi(\tilde{\mathbf{x}}(\phi))\|}\right|_{\tilde{\mathbf{x}}=A\mathbf{u}} = \frac{A^{-1}\mathbf{u}}{\|A^{-1}\mathbf{u}\|}. \tag{11}$$

This and (10) gives a closed-form expression for the offset of $\tilde{E}_1$ in the linearly transformed coordinates. To convert back to the original coordinates, we must apply the inverse of the applied linear transformation to get

$$\mathbf{x}_{eb}(\phi) = R_2 \Delta(\mathbf{b}/r) R_2^T \tilde{\mathbf{x}}_{ofs}(\phi), \tag{12}$$

which is a closed-form parametric expression for the boundary of the Minkowski sum $E_1 \oplus (R_2 \cdot E_2)$. The Minkowski difference is computed in exactly the same way, with r replaced by $-r$ in (10), and is valid as long as the resulting sphere can be fully contained in $\tilde{E}_1$ when the boundaries meet. This is essentially the condition in the statement of Theorem 2.

4 Relationship Between Minkowski Sums and Kinematic Formulae

The integral in the Principal Kinematic formula can be rewritten for convex bodies as

$$\begin{aligned} \int_{SE(n)} \iota(C_0 \cap g \cdot C_1)\, dg &= \int_{SO(n)} \int_{\mathbb{R}^n} \iota(C_0 \cap \mathbf{t} \cdot (R \cdot C_1))\, d\mathbf{t}\, dR \\ &= \int_{SO(n)} V(C_0 \oplus (R \cdot \bar{C}_1)) dR \end{aligned} \tag{13}$$

where (1) has been used and the translation integral has been replaced with the volume of the Minkowski sum of ellipsoids, with $\bar{C}_1 \doteq \{-\mathbf{x} \,|\, \mathbf{x} \in C_1\}$. For centrosymmetric bodies like ellipsoids, $\bar{E}_i = E_i$.

The subject of translative kinematic formulas for general bodies that compute integrals of the form $\int_{\mathbb{R}^n} \iota(C_0 \cap \mathbf{t} \cdot C_1)\, d\mathbf{t}$ for convex bodies has been addressed extensively in [8–10, 18, 20]. The same can be written for containment, with $\oplus$ replaced by $\ominus$. Specifically, given ellipsoidal bodies E_0 and E_1 with the conditions of Theorem 2 holding, we can compute a translative integral geometric formula for containment of the form

$$\int_{\mathbb{R}^n} b(\mathbf{t} \cdot E_1 \subset E_0) d\mathbf{t} = V(E_0 \ominus E_1).$$

Having a parametric expression for the boundary $\partial(E_0 \ominus E_1)$ such as that in (12) with $r \to -r$, gives a way to compute the volume efficiently (using the divergence theorem).

The result of this formula is related to the formulas given in Theorem 2 because

$$\int_{SE(n)} b(g \cdot C_1 \subset C_0)\, dg = \int_{SO(n)} V(C_0 \ominus (R \cdot \bar{C}_1))dR. \tag{14}$$

Moreover, if the range of allowable rotations does not meet the conditions of Theorem 2, it is possible to compute the allowable room to move with a restricted rotation range. This involves computing the integral over a region of $SO(n)$ rather than all of it, which can be computationally intensive.

5 Motion of Frustrated Ellipsoids

Suppose that E_1 and E_0 are both axis-aligned ellipsoids with their centers of mass at the origin. Then both are described by (8), with $\mathbf{a}$ describing the semi-axis lengths for the smaller (moving) body E_1 and $\mathbf{b}$ the semi-axis lengths for the fixed body, E_0. Suppose that $b_i/a_i = 1 + \varepsilon_i$ where $0 < \varepsilon_i << 1$. Then there is very little room for E_1 to "rattle around" inside of E_0, and the formulations of the previous sections will fail. This scenario represents an opposite extreme where different methods are applicable. Here it is safe to assume that the allowable motion is small, and so

$$R \approx \mathbb{I} + S(\mathbf{r}) \text{ where } S(\mathbf{r})^T = -S(\mathbf{r}) = S(-\mathbf{r}) \in so(n) \tag{15}$$

is a skew-symmetric matrix function linear in $\mathbf{r}$. In the n-dimensional case $\mathbf{r} \in \mathbb{R}^{n(n-1)/2}$. In 3D, this is the 3D dual vector such that $S(\mathbf{r})\mathbf{v} = \mathbf{r} \times \mathbf{v}$. For such small motions, elements of $SE(3)$ can be approximated well with those in $se(3)$, the corresponding Lie algebra. And if we want to compute volumes of motion, $dg \approx d\mathbf{r}d\mathbf{t}$.

If E_1 can move inside of E_0 without their boundaries intersecting at more than one point, then substituting the parametric equation for E_1 into the implicit equation for E_0 must satisfy

$$(R\Lambda(\mathbf{a})\mathbf{u} + \mathbf{t})^T \Lambda^{-2}(\mathbf{b})(R\Lambda(\mathbf{a})\mathbf{u} + \mathbf{t}) \leq 1 \tag{16}$$

for all values of $\mathbf{u} \in S^{n-1}$. Moreover, substituting (15) into (16) gives an equation that can be written in the form

$$C(\mathbf{z}) = \mathbf{z}^T H\mathbf{z} + \mathbf{h}^T\mathbf{z} + c \leq 1 \tag{17}$$

where $\mathbf{z} = [\mathbf{r}^T, \mathbf{t}^T]^T$ and $0 < c < 1$. Note that H, $\mathbf{h}$, and c all depend on $\mathbf{u}$, and for each $\mathbf{u}$ the inequality (17) defines a solid hyper-ellipsoid in $se(n) \cong \mathbb{R}^{n(n+1)/2}$. The intersection of all of these hyper-ellipsoids taken over all $\mathbf{u} \in S^{n-1}$ will give the allowable range of motion.

A much easier quantity to compute is the subset of this that is of the form of a centered hyper-ellipsoid in $se(n)$ parameterized as $\mathbf{z} = B\mathbf{w}$ where $\mathbf{w}$ is a unit vector in $\mathbb{R}^{n(n+1)/2}$ (or equivalently, a point in the hyper-sphere $S^{n(n+1)/2-1}$). Substituting this in (17), and using the properties of consistent matrix norms (in this case using the property that $\|H\mathbf{v}\| \leq \|H\| \cdot \|\mathbf{v}\|$ where $\|H\|$ can be either the induced 2-norm or the Frobenius norm), then

$$\begin{aligned} C(B\mathbf{w}) &= \mathbf{w}^T B^T H(\mathbf{u}) B\mathbf{w} + [\mathbf{h}(\mathbf{u})]^T B\mathbf{w} + c(\mathbf{u}) \leq |\mathbf{w}^T B^T H(\mathbf{u}) B\mathbf{w}| \\ &\quad + |[\mathbf{h}(\mathbf{u})]^T B\mathbf{w}| + c(\mathbf{u}) \\ &= \|\mathbf{w}^T B^T H(\mathbf{u}) B\mathbf{w}\| + \|[\mathbf{h}(\mathbf{u})]^T B\mathbf{w}\| + c(\mathbf{u}) \leq \|B^T H(\mathbf{u}) B\| \\ &\quad + \|[\mathbf{h}(\mathbf{u})]^T B\| + c(\mathbf{u}) \\ &\leq \|H(\mathbf{u})\| \cdot \|B\|^2 + \|\mathbf{h}(\mathbf{u})\| \cdot \|B\| + c(\mathbf{u}) \\ &\leq \widehat{\|H\|} \cdot \|B\|^2 + \widehat{\|\mathbf{h}\|} \cdot \|B\| + \hat{c} \end{aligned} \tag{18}$$

where $\widehat{\|H\|}, \widehat{\|\mathbf{h}\|}$, and $\hat{c}$ are upper bounds on $\|H(\mathbf{u})\|$, $\|\mathbf{h}(\mathbf{u})\|$, and $c(\mathbf{u})$ that do not depend on $\mathbf{u}$. For example, $c(\mathbf{u}) = \mathbf{u}^T \Lambda(\mathbf{a}) \Lambda^{-2}(\mathbf{b}) \Lambda(\mathbf{a}) \mathbf{u} \leq \|\Lambda(\mathbf{a}) \Lambda^{-2}(\mathbf{b}) \Lambda(\mathbf{a})\|_2 = \max_i a_i^2 / b_i^2$. Space limits do not permit explicit formulas for $\widehat{\|H\|}$ and $\widehat{\|\mathbf{h}\|}$ in this venue, but such bounds can be found using (7). Then, from the above equation, $\|B\|$ must satisfy the quadratic formula

$$\|B\| \leq \frac{-\widehat{\|\mathbf{h}\|} + \sqrt{\widehat{\|\mathbf{h}\|}^2 + 4(1-\hat{c})\widehat{\|H\|}}}{2\widehat{\|H\|}}.$$

This imposes conditions on the maximal motion ellipsoid, and the volume of the resulting motion (which is a lower bound on the total allowable motion) of the form

$$V(D^{n(n+1)/2}) \cdot |\det B| \leq \int_{SE(n)} b(g \cdot E_1 \subset E_0)\, dg.$$

where $V(D^{n(n+1)/2})$ is the volume of the unit hyper-disk (or ball) in $\mathbb{R}^{n(n+1)/2} \cong se(n)$.

6 Conclusions

New expressions that describe the available room to move for one convex body contained inside of another are given. In the extreme case when one body is very small relative to the other, and it can freely rotate and translate such that all contacts of the surfaces are single-point contacts, closed form formulas for containment based on the Principal Kinematic formula are possible. Likewise, in the other extreme case in

which one body fits inside the other with only a little bit of wiggle room, we provide closed-form lower bounds on the allowable motion. For the intermediate case of possibly hindered rotation, we show how Minkowski sum can be used to compute allowable translational motion for each fixed orientation of the inner moving body, and these can be added up by integrating over all allowable rotations.

References

1. Angeles, J.: Rational Kinematics. Springer, New York (1988/2011)
2. Blaschke, W.: Vorlesungen Über Integralgeometrie. VEB Dt. Verlag d Wiss., Berlin (1955)
3. Boothroyd, G., Redford, A.H.: Mechanized Assembly: Fundamentals of Parts Feeding, Orientation, and Mechanized Assembly. McGraw-Hill, New York (1968)
4. Bottema, O., Roth, B.: Theoretical Kinematics. Dover Publications, New York (2011)
5. Chirikjian, G.S.: Parts entropy and the principal kinematic formula. In: IEEE International Conference on Automation Science and Engineering, CASE 2008, pp. 864–869. IEEE (2008)
6. Chirikjian, G.S.: Stochastic Models, Information Theory, and Lie Groups, Volume 2: Analytic Methods and Modern Applications. Springer, Birkhäuser (2012)
7. Chirikjian, G.S., Kyatkin, A.B.: Engineering Applications of Noncommutative Harmonic Analysis. CRC Press, Boca Raton (2001)
8. Glasauer, S.: Translative and kinematic integral formulae concerning the convex hull operation. Mathematische Zeitschrift **229**(3), 493–518 (1998)
9. Goodey, P., Weil, W.: Translative integral formulae for convex bodies. aequationes mathematicae **34**(1), 64–77 (1987)
10. Groemer, H.: On translative integral geometry. Archiv der Mathematik **29**(1), 324–330 (1977)
11. Karnik, M., Gupta, S.K., Magrab, E.B.: Geometric algorithms for containment analysis of rotational parts. Comput. Aided Des. **37**(2), 213–230 (2005)
12. Klain, D.A., Rota, G.C.: Introduction to Geometric Probability. Cambridge University Press, Cambridge (1997)
13. Kurzhanskiy, A.A., Varaiya, P.: Ellipsoidal toolbox (et). In: Forty-fifth IEEE Conference on Decision and Control, pp. 1498–1503. IEEE (2006)
14. McCarthy, J.M.: Introduction to Theoretical Kinematics. MIT Press/iTunes, Boston (1990/2013)
15. Ros, L., Sabater, A., Thomas, F.: An ellipsoidal calculus based on propagation and fusion. IEEE Trans. Syst. Man Cybern. Part B Cybern **32**(4), 430–442 (2002)
16. Sanderson, A.: Parts entropy methods for robotic assembly system design. In: Proceedings of the 1984 IEEE International Conference on Robotics and Automation (ICRA '84), pp. 600–608 (1984)
17. Santaló, L.A.: Integral Geometry and Geometric Probability. Cambridge University Press, Cambridge (2004)
18. Schneider, R., Weil, W.: Translative and kinematic integral formulae for curvature measures. Mathematische Nachrichten **129**(1), 67–80 (1986)
19. Schneider, R., Weil, W.: Stochastic and Integral Geometry. Springer, New York (2008)
20. Weil, W.: Translative integral geometry. In: Hübler, A.,et al. (eds.) Geobild **89**, pp. 75–86. Akademie-Verlag, Berlin (1989)
21. Zhang, G.: A sufficient condition for one convex body containing another. Chin. Ann. Math. Ser. B **9**(4), 447–451 (1988)
22. Zhou, J.: When can one domain enclose another in $\mathbb{R}^3$? J Aust Math Soc A **59**(2), 266–272 (1995)

The Dimensional Synthesis of 3-RPR Parallel Mechanisms for a Prescribed Singularity-Free Constant-Orientation Workspace

Amirhossein Karimi, Mehdi Tale Masouleh and Philippe Cardou

Abstract This chapter addresses the dimensional synthesis of a planar 3-DOF parallel mechanism, namely 3-RPR platform, in order to obtain a geometry for this mechanism whose constant orientation workspace contains a prescribed workspace while avoids the singularity configurations within the whole prescribed workspace. To this end, first the constraint expressions describing mathematically the desired geometry to contain the prescribed workspace and eliminate the singularities are introduced. Moreover, a multi-objective programming problem is presented in which a scaling factor is maximized to enlarge the workspace region in addition to a parameter to maximize the singularity-free workspace area. Then, the problem is relaxed to a convex form. Finally, to prevent the relaxation from becoming too loose, a branch-and-prune algorithm is applied.

Keywords Parallel mechanism · Dimensional synthesis · Convex programming · 3-RPR platform · Singularity-free workspace

1 Introduction

The synthesis of parallel mechanisms is much more complex than that of serial robots, due to many reasons such as the increased number of parameters required to determine the geometry of the mechanism and complicated analysis of workspace

A. Karimi (✉) · M. T. Masouleh
Faculty of New Sciences and Technologies, University of Tehran, Tehran, Iran
e-mail: ah.karimi@ut.ac.ir

M. T. Masouleh
e-mail: m.t.masouleh@ut.ac.ir

P. Cardou
Department of Mechanical Engineering, Laval University, Quebec City,
QC G1V 0A6, Canada
e-mail: pcardou@gmc.ulaval.ca

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_38,

and singularity configurations [6]. The studies conducted in the synthesis of parallel mechanisms are mostly based on heuristic methods such as genetic algorithm [3, 8], and interval analysis [5] or using a sequential procedure in which two arms are designed first to satisfy the workspace requirements, then the third arm is designed to provide a singularity-free workspace [10]. It should be noted that in the latter, a feasible subset for the problem of the dimensional synthesis of a 3-R$\underline{\text{P}}$R parallel mechanism is obtained in order to contain a prescribed workspace while it is singularity-free.

There are various indices according to which a parallel mechanism can be designed. It is indispensable for a mechanism which performs any special task, to have a prescribed workspace. Therefore, the workspace area and shape are of the foremost indices which can be taken into account for the purpose of dimensional synthesis. Another important criterion for the aim of synthesis is the presence of singularities. This chapter deals with these two indices for the purpose of synthesizing of 3-R$\underline{\text{P}}$R parallel mechanisms.

The remainder of the chapter is organized as follows. First, the optimization problem for the synthesis of 3-R$\underline{\text{P}}$R parallel mechanism is expressed by introducing the constraints required for the optimal geometry to contain the prescribed workspace while avoiding singularities. Then, a relaxation method is presented to convexify the non-convex constraints. A branch-and-prune algorithm is then proposed, which converges to the global optimal solution. Finally, the chapter concludes by providing some remarks and describing related ongoing work.

2 Formulation of the Dimensional Synthesis Problem of 3-R$\underline{\text{P}}$R Parallel Mechanisms

Figure 1 depicts schematically a 3-R$\underline{\text{P}}$R Parallel Mechanism (PM). The position vector of the operation point of the mobile platform with respect to the fixed frame is represented by $\mathbf{p} = [x, y]^T$. This operation point is chosen to be point B_1. The position vectors of point B_i in the fixed and mobile frames are denoted by $\mathbf{b}_i$ and $\mathbf{b'}_i$, respectively. The design parameters of this platform are listed as (x_1, y_1), (x_2, y_2), (x_3, y_3) which stand for the coordinates of the points attached to the base, l_1, l_2 which are the lengths of two sides of the Mobile Platform (MP) triangle, $\rho_{\max_i} (i = 1, 2, 3)$ as the maximum lengths of each leg, and θ_1, which is the angle between $\overline{\mathbf{B}_1\mathbf{B}_2}$ and $\overline{\mathbf{B}_1\mathbf{B}_3}$. In this chapter it is assumed that $x_1 = y_1 = 0$. For the sake of simplicity, it is considered that θ_1 is fixed. Thus, the number of main design parameters for 3-R$\underline{\text{P}}$R PM reduces to nine, i.e., $x_2, y_2, x_3, y_3, l_1, l_2, \rho_{\max_i} (i = 1, 2, 3)$. It should be noted that $\rho_{\min}$ is assumed to be zero in this approach. One can define the dimensional synthesis problem as that of finding the geometry (design parameters) of a 3-R$\underline{\text{P}}$R PM such that the workspace $\mathcal{W}$ includes the desired workspace $\mathcal{W}_d$ and $g(\mathbf{x}) \neq 0$, for $\forall \mathbf{x} \in \mathcal{W}_d$; where $g(\mathbf{x})$ is the singularity loci expression. The desired workspace, $\mathcal{W}_d$, is considered to be a prescribed circle with the center point (x_d, y_d) and radius R_d.

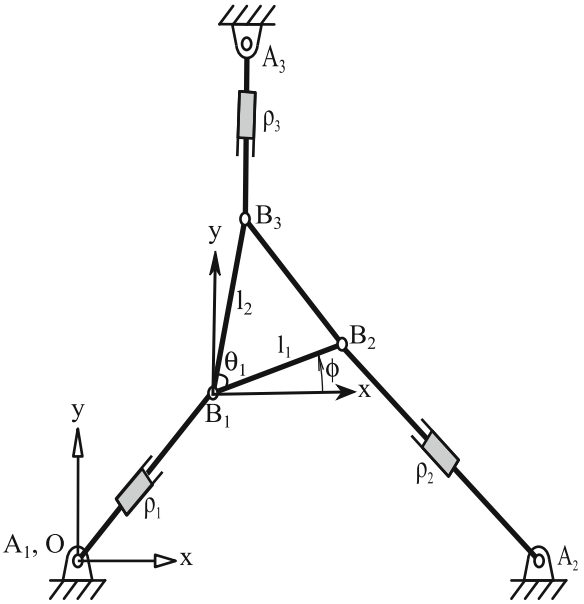


Fig. 1 A 3-RPR PM

Therefore, a mathematical description of the dimensional synthesis problem under study in this chapter can be written as:

$$\begin{aligned} \text{max.} \quad & \mathscr{W} \\ \text{s.t.} \quad & \mathscr{W}_d \subseteq \mathscr{W}, \\ & g(\mathbf{x}) \neq 0, \ \text{for } \forall \mathbf{x} \in \mathscr{W}_d, \\ & h_i(\mathbf{t}) \leq 0, \ \mathbf{t} = [x_2, x_3, y_2, y_3, l_1, l_2, \rho_{\max_1}, \rho_{\max_2}, \rho_{\max_3}], \quad i = 1, \ldots, m. \end{aligned} \tag{1}$$

where $h_i(\mathbf{t}),\ i = 1, \ldots, m$, are the constraint inequalities on the design parameters of the mechanism. In the following, the aim is to obtain the objective function and the constraints mentioned in Eq. (1).

2.1 Workspace Constraints

According to Fig. 2, in which v_2 is the constraint imposed by the second limb of the mechanism on the workspace, one can obtain the constraints required for the workspace to include the internal region of the prescribed circle as:

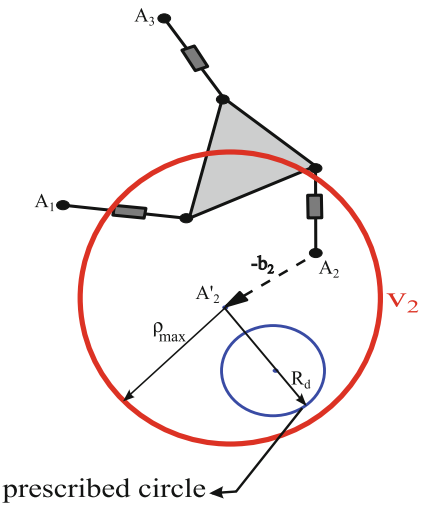


Fig. 2 Schematic illustration of the workspace constraints

$$
\begin{aligned}
&||(x_d, y_d)||_2 + R_d \leq \rho_{\max_1},\\
&||(x_2 - l_1\cos\phi - x_d, y_2 - l_1\sin\phi - y_d)||_2 + R_d \leq \rho_{\max_2},\\
&||(x_3 - l_2\cos(\phi+\theta_1) - x_d, y_3 - l_2\sin(\phi+\theta_1) - y_d)||_2 + R_d \leq \rho_{\max_3}.
\end{aligned}
\tag{2}
$$

where $||\cdot||_2$ stands for the Euclidean norm of its vector argument. It should be noted that above constraints are Second-Order Cone (SOC) constraints and are thus convex. Moreover, this approach can readily be generalized to any polyhedron, convex or not, since, in order to include the polyhedron, it is sufficient to contain all its vertices.

2.2 Geometric Constraints

In order to avoid impractical solutions, the following linear geometric constraints are considered to restrict the geometry of the platform:

$$
\begin{aligned}
&\underline{x_2} \leq x_2 \leq \overline{x_2}, \quad \underline{y_2} \leq y_2 \leq \overline{y_2},\\
&\underline{x_3} \leq x_3 \leq \overline{x_3}, \quad \underline{y_4} \leq y_4 \leq \overline{y_4},\\
&\underline{l_1} \leq l_1 \leq \overline{l_1}, \quad \underline{l_2} \leq l_2 \leq \overline{l_2},\\
&\underline{\rho_{\max_i}} \leq \rho_{\max_i} \leq \overline{\rho_{\max_i}}, \quad i = 1, 2, 3.
\end{aligned}
\tag{3}
$$

where the underline and overline indicate the lower and upper bound for each variable, respectively. In order to prevent the mobile platform from becoming larger than the base, one can add the following constraints:

$$\begin{aligned} &||(x_2, y_2)||_2 - l_1 \geq 0, \\ &||(x_3, y_3)||_2 - l_2 \geq 0, \\ &||((x_2, y_2) - (x_3, y_3))||_2 - l_3 \geq 0. \end{aligned} \tag{4}$$

where $l_3 = \sqrt{l_1^2 + l_2^2 - 2l_1 l_2 \cos\theta_1}$. The above are not convex constraints and thus one can exert the McCormick or Al-Khayyal relaxation [1, 7, 9]. This relaxation is based on the convex (concave) envelope of a function $f(\mathbf{x})$ over $\boldsymbol{\Lambda}$, i.e., the pointwise supremum (infimum) of all convex (concave) functions which underestimate (overestimate) $f(\mathbf{x})$ over $\boldsymbol{\Lambda}$, where:

$$\boldsymbol{\Lambda} = \{\mathbf{x} \in \mathbb{R}^n | \underline{\mathbf{x}} \leq \mathbf{x} \leq \overline{\mathbf{x}}\}. \tag{5}$$

From [1], one can prove for $\mathbf{x} = [x, y]^T$, $\underline{\mathbf{x}} = [\underline{x}, \underline{y}]^T$ and $\overline{\mathbf{x}} = [\overline{x}, \overline{y}]^T$ that:

$$\begin{aligned} \mathrm{CVX}_{\boldsymbol{\Lambda}}(xy) &= \max(\underline{x}y + \underline{y}x - \underline{x}\,\underline{y}, \overline{x}y + \overline{y}x - \overline{x}\,\overline{y}), \\ \mathrm{CCV}_{\boldsymbol{\Lambda}}(xy) &= \min(\overline{x}y + \underline{y}x - \overline{x}\,\underline{y}, \underline{x}y + \overline{y}x - \underline{x}\,\overline{y}). \end{aligned} \tag{6}$$

where $\mathrm{CVX}_{\boldsymbol{\Lambda}}$ and $\mathrm{CCV}_{\boldsymbol{\Lambda}}$ stand for convex and concave envelopes over $\boldsymbol{\Lambda}$, respectively. Therefore, to convexify the constraints in Eq. (4), the following substitution of variables is used:

$$X_2 = -x_2^2, \quad X_3 = -x_3^2, \quad Y_2 = -y_2^2, \quad Y_3 = -y_3^2, \quad Z_1 = x_2 x_3, \quad Z_2 = y_2 y_3. \tag{7}$$

As a result, the relaxation of the constraints in Eq. (4) is:

$$\begin{aligned} &l_1^2 + X_2 + X_3 + Y_2 + Y_3 + 2Z_1 + 2Z_2 \leq 0, \\ &l_2^2 + X_2 + Y_2 \leq 0, \\ &l_1^2 + l_2^2 - 2l_1 l_2 \cos\theta_1 + X_3 + Y_3 \leq 0, \\ &\max(\underline{x_2}x_3 + \underline{x_3}x_2 - \underline{x_2}\,\underline{x_3}, \overline{x_2}x_3 + \overline{x_3}x_2 - \overline{x_2}\,\overline{x_3}) \leq Z_1 \\ &Z_1 \leq \min(\overline{x_2}x_3 + \underline{x_3}x_2 - \overline{x_2}\,\underline{x_3}, \underline{x_2}x_3 + \overline{x_3}x_2 - \underline{x_2}\,\overline{x_3}), \\ &\max(\underline{y_2}y_3 + \underline{y_3}y_2 - \underline{y_2}\,\underline{y_3}, \overline{y_2}y_3 + \overline{y_3}y_2 - \overline{y_2}\,\overline{y_3}) \leq Z_2 \\ &Z_2 \leq \min(\overline{y_2}y_3 + \underline{y_3}y_2 - \overline{y_2}\,\underline{y_3}, \underline{y_2}y_3 + \overline{y_3}y_2 - \underline{y_2}\,\overline{y_3}), \\ &\underline{x_2}\,\overline{x_2} - (\overline{x_2} + \underline{x_2})x_2 \leq X_2 \leq \min(-2\overline{x_2}x_2 + \overline{x_2}^2, -2\underline{x_2}x_2 + \underline{x_2}^2), \\ &\underline{x_3}\,\overline{x_3} - (\overline{x_3} + \underline{x_3})x_3 \leq X_3 \leq \min(-2\overline{x_3}x_3 + \overline{x_3}^2, -2\underline{x_3}x_3 + \underline{x_3}^2), \\ &\underline{y_2}\,\overline{y_2} - (\overline{y_2} + \underline{y_2})y_2 \leq Y_2 \leq \min(-2\overline{y_2}x_2 + \overline{y_2}^2, -2\underline{y_2}y_2 + \underline{y_2}^2), \\ &\underline{y_3}\,\overline{y_3} - (\overline{y_3} + \underline{y_3})y_3 \leq Y_3 \leq \min(-2\overline{y_3}y_3 + \overline{y_3}^2, -2\underline{y_3}y_3 + \underline{y_3}^2). \end{aligned} \tag{8}$$

2.3 Singularity Constraint

The singularity loci expression, $g(\mathbf{x}, \mathbf{t})$, is a quadratic polynomial with respect to $\mathbf{x}$ and a fourth-order polynomial with respect to $(\mathbf{x}, \mathbf{t})$. To avoid the singularities within the prescribed circle, one of the following implications must be satisfied:

$$\begin{aligned}(x - x_d)^2 + (y - y_d)^2 - R_d^2 \leq 0 &\Longrightarrow g(\mathbf{x}, \mathbf{t}) < 0,\\ (x - x_d)^2 + (y - y_d)^2 - R_d^2 \leq 0 &\Longrightarrow g(\mathbf{x}, \mathbf{t}) > 0.\end{aligned} \tag{9}$$

To simplify the procedure of the dimensional synthesis, it is advantageous to divide the optimization problem into two steps: first, obtain l_1, l_2 in order to maximize the workspace area as discussed in the next subsection, and then, optimize the other decision variables to make the prescribed workspace singularity-free while maximizing the workspace area.

Once l_1, l_2 are obtained, all the coefficients of $g(\mathbf{x}, \mathbf{t})$ with respect to $\mathbf{x}$ will be linear functions of the design variables, except for four terms, i.e., x_2x_3, x_2y_3, x_3y_2 and y_2y_3. Thus, one can use the following substitutions in order to linearize all the coefficients:

$$Z_1 = x_2x_3,\ Z_2 = y_2y_3,\ Z_3 = x_2y_3,\ Z_4 = x_3y_2. \tag{10}$$

Therefore, similar inequalities to Eq. (8) can be obtained for $Z_3 = x_2y_3$, $Z_4 = x_3y_2$ in order to convexify them. By using the *S-procedure* [2], one can convert the constraint in Eq. (9) to a LMI (Linear Matrix Inequality). If one reformulates the singularity loci and the prescribed circle equations as follows:

$$\begin{aligned}&g(\mathbf{x}, \mathbf{t}) = \mathbf{x}^T\mathbf{E}\mathbf{x} + 2\mathbf{f}^T\mathbf{x},\\ &(x - x_d)^2 + (y - y_d)^2 - R_d^2 = \mathbf{x}^T\mathbf{P}\mathbf{x} + 2\mathbf{q}^T\mathbf{x} + r.\end{aligned} \tag{11}$$

then the singularity constraint can be reformulated as:

$$\begin{bmatrix}\lambda\mathbf{P} + \mathbf{E} & \lambda\mathbf{q} + \mathbf{f}\\ (\lambda\mathbf{q} + \mathbf{f})^T & \lambda r - \eta\end{bmatrix} \succeq 0,\ \lambda \geq 0. \tag{12}$$

where η is an optimizatiom variable which should be positive in order to $g(\mathbf{x}) > 0$. It should be noted that this approach can readily be modified for $g(\mathbf{x}) < 0$. Therefore, by solving two optimization problems, one for $g(\mathbf{x}) < 0$, the other for $g(\mathbf{x}) > 0$, the design parameters with the maximum objective value will be obtained.

2.4 Objective Function

The objective function should be selected in order to maximize the workspace area. The most reasonable way to maximize the workspace area is to use a scaling factor,

where by maximizing it a larger workspace can be obtained. To do so, we define s as the scaling factor and one can modify the workspace constraints in Eq. (2) for this purpose, i.e.,

$$\begin{aligned} \max. \quad & s + \eta \\ \text{s.t.} \quad & ||(x_d, y_d)||_2 + sR_d \leq \rho_{\max_1}, \\ & ||(x_2 - l\cos\phi - x_d, y_2 - l\cos\phi - y_d)||_2 + sR_d \leq \rho_{\max_2}, \\ & ||(x_3 - l\cos(\phi + \theta_1) - x_d, y_3 - l\cos(\phi + \theta_1) - y_d)||_2 + sR_d \leq \rho_{\max_3}. \end{aligned} \tag{13}$$

If $s \geq 1$, it is guaranteed that the obtained geomtry of the platform includes the prescribed circle within its workspace. Furthermore, by maximizing η in the objective function as a second parameter, one can make a safety margin for the singularity-free prescribed workspace. Also, if the objective in the problem of synthesis is to have smaller values for $\rho_{\max_i}$, $i = 1, 2, 3$, one can modify the objective function to minimize $\rho_{\max_1} + \rho_{\max_2} + \rho_{\max_3}$ while $s \geq 1$ is one of the constraints of the problem.

3 Branch and Prune Algorithm

The final challenge is proposing a branch-and-prune algorithm in order to avoid the relaxations exerted in Eqs. (8) and (10) to become too loose, which due to the space limitation is only described briefly.

The main idea behind this algorithm is to bisect the intervals bounding the design parameters. If the solution obtained from the relaxed optimization problem is feasible for the original optimization problem, then the algorithm is finished. Otherwise, in the next step, the search space is divided into 16 equal sections and the relaxed problem is solved for each section. One of the three following items can be true for the solution obtained in each section:

1. The original and relaxed problems are both infeasible: the dividing procedure does not continue for this section.
2. The original problem is infeasible and the relaxed problem is feasible: the dividing procedure continues for this section.
3. The original and relaxed problems are both feasible: the dividing procedure does not continue for this section.

The procedure of bisecting the intervals stops when there is no section left to be divided into subsections. It should be noted that since the relaxed problem is convex, the branch and prune algorithm will converge to the global optimal solution in a low computational time.

By resorting to CVX [4], a package for solving convex programming problems, the optimal design parameters are obtained for the 3-RPR parallel mechanism. The lower bound and upper bound for all the design parameters are selected to be -2

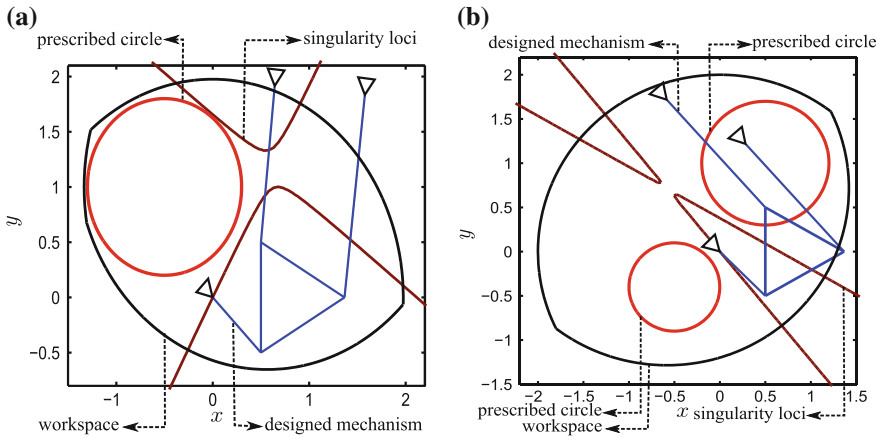


Fig. 3 The results of synthesis problem for 3-RPR PM ($\phi = \frac{\pi}{6}$). **a** One circle is prescribed, **b** two circles are prescribed

and 2, respectively. The problem is solved for $\phi = \frac{\pi}{6}$ when the objective function is s and the prescribed workspace is a circle represented with the following set:

$$\{(x, y) \in \mathbb{R}^2 | (x + 0.5)^2 + (y - 1)^2 \leq 0.8^2\}. \tag{14}$$

As a second example, two circles are prescribed and the objective function is $s + \eta$. The problem is solved in order to find the optimal geometry of the mechanism. The results of the designed mechanisms are depicted in Fig. 3a, b. The computational time obtained by a PC equipped with an Intel(R) Core(TM) i5-2430M CPU @ 2.40 GHz, and 4 GB RAM, for the first and second results illustrated in Fig. 3 are 9.2 and 4.3 s, respectively.

4 Conclusion

In this chapter, an optimal dimensional synthesis approach was presented for 3-RPR parallel mechanisms. The required constraints for containing the prescribed workspace and avoiding the singularities within the workspace were proposed. A multi-objective programming problem was considered in order to maximize the workspace area and maximize a parameter for the aim of singularity avoidance. Furthermore, a relaxation approach was introduced to convexify the problem and also a branch-and-prune based algorithm was applied in order to converge to the global optimal solution for the geometry of the mechanism. Ongoing work includes solving the problem of dimensional synthesis for other planar and spatial parallel

mechanisms such as the 3-PRR and 6-UPS parallel mechanisms. Moreover, another aim is to find an approach which guarantees the existence of a singularity-free path between two or more separated parts of a prescribed workspace.

References

1. Al-Khayyal, F.A., Falk, J.E.: Jointly constrained biconvex programming. Math. Oper. Res. **8**(2), 273–286 (1983)
2. Boyd, S.P., Vandenberghe, L.: Convex Optimization. Cambridge university press, Cambridge (2004)
3. Caro, S., Chablat, D., Ur-Rehman, R., Wenger, P.: Multiobjective design optimization of 3-PRR planar parallel manipulators. In: Global Product, Development, pp. 373–383 (2011)
4. Grant, M., Boyd, S.: CVX: Matlab software for disciplined convex programming, version 2.0 beta (2013)
5. Hao, F., Merlet, J.P.: Multi-criteria optimal design of parallel manipulators based on interval analysis. Mech. Mach. Theory **40**(2), 157–171 (2005)
6. Husty, M.L., Pfurner, M., Schrocker, H.P., Brunnthaler, K.: Algebraic methods in mechanism analysis and synthesis. Robotica **25**(6), 661–675 (2007)
7. McCormick, G.P.: Computability of global solutions to factorable nonconvex programs: Part I-convex underestimating problems. Math. program. **10**(1), 147–175 (1976)
8. Saadatzi, M., Tale Masouleh, M., Taghirad, H., Gosselin, C., Teshnehlab, M.: Multi-objective scale independent optimization of 3-RPR parallel mechanisms. In: Proceedings of the IFToMM (2011)
9. Scott, J.K., Stuber, M.D., Barton, P.I.: Generalized McCormick relaxations. J. Global. Optim. **51**(4), 569–606 (2011)
10. Yang, Y., O'Brien, J.F.: A sequential method for the singularity-free workspace design of a three legged parallel robot. Mech. Mach. Theory **45**(11), 1694–1706 (2010)

Approximating Constrained Hand Paths Via Kinematic Synthesis with Contact Specifications

Hyosang Moon, Nina P. Robson and Reza Langari

Abstract This chapter presents part of our recent efforts in developing computational models of human motor coordination with reduced mobility. A recently developed kinematic synthesis is implemented to approximate natural human hand profiles with an elbow joint constraint. The constraint condition holds the hand on a spherical workspace centered at the shoulder. The assumption that the hand is in contact with the surface during the entire movement, allows us to describe the contact condition by using higher order kinematic constraints such as velocities and accelerations. By adopting contact specifications at an initial and a final task positions, kinematic synthesis and path planning techniques enable us to generate an entire hand path connecting the two positions. It was found that the proposed method closely approximates an actual human hand path, obtained experimentally.

Keywords Kinematic synthesis · Higher order kinematic specifications · Human motor coordination

1 Introduction

Human arm can perform versatile reaching actions in activities of daily living (ADL) mostly for achieving a desired hand manipulation such as pointing or grasping. Coordinating such movements is not trivial since the central nervous system (CNS)

H. Moon (✉) · N. P. Robson · R. Langari
Texas A&M University, College Station, TX 77843, USA
e-mail: hsmoon@tamu.edu

R. Langari
e-mail: rlangari@tamu.edu

N. P. Robson
California State University, Fullerton, CA 92843, USA
e-mail: nrobson@fullerton.edu

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_39,

needs to resolve a highly redundant mapping problem to realize required hand kinematics and dynamics with a set of control input for the complex musculoskeletal system. However, every time we encounter such a problem in ADL, the CNS almost automatically finds and executes an optimal solution without significant conscious efforts. It can be explained that there are some governing rules within the CNS that impose additional constraints on the assigned problem and induce a finite set of patterned solutions. In his experimental study on point-to-point reaching actions on a horizontal plane, Morasso [7] concluded that the hand kinematics tends to follow a straight path connecting two task points with a smooth bell-shaped speed profile. Flash and Hogan [3] formulated a mathematical expression called the minimum jerk model to approximate such features of natural hand profiles in reaching movements. Uno et al. [12] considered that the CNS takes a motion dynamics (i.e., variations in joint torque values) into account for its motor coordination to explain slightly curved experimental hand paths.

In this chapter, we hypothesize that hand contact conditions play an important role in the governing rules to coordinate a point-to-point reaching in a stereotyped manner. Contact conditions of a hand with one or more objects can define velocity and acceleration specifications in the vicinity of those contact points based on theoretical backgrounds by Rimon and Burdic [8, 9]. Recently, Robson and McCarthy [11] introduced a systematic method for the kinematic synthesis of planar mechanical linkages such that they do not violate normal direction and curvature constraints imposed by contacts with objects. Using the geometry of the task, they showed how to transform these constraints into velocity and acceleration specifications of the moving body/end-effector. Their work was further continued by Robson and Tolety [10], who extended the contact geometry problem to the three dimensional case.

We believe that the CNS senses directional constraints on hand velocity, acceleration and higher derivative vectors due to relative curvatures of contact geometries and utilizes them while generating a hand profile. Consider that a targeting hand position is assigned as an only input for planning and executing a point-to-point reaching. Then, the CNS needs to generate a hand path to fill out a gap to the targeting hand position from an initial state. When there is a contact condition imposed at each task point, directions of hand velocity and acceleration vectors are constrained by contact geometries in the vicinity of each task point. As a result, the specified contact conditions can assist the CNS to reduce a range of possible hand profile solutions: i.e., the solution set can be filtered to meet those kinematic specifications simultaneously. In this manner, we believe that the entire hand profile can be approximated/reproduced/predicted by using the linkage kinematic synthesis techniques with the given contact conditions. In this study, elbow constrained reaching movements on a spherical workspace are approximated via the previously developed spatial SS linkage synthesis for contact specifications shown in [10]. The spherical contact condition was realized by an elbow joint constraint with a medical brace in the experiment.

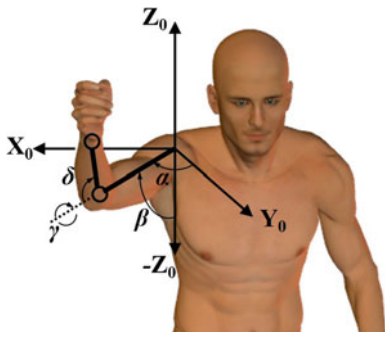


Fig. 1 The defined joint coordinates to represent an elbow constrained human arm motion kinematics [6]

2 Human Arm Kinematic Model with a Constrained Elbow Joint

Human arm kinematics can be simplified as a seven DOF (degrees of freedom) SRS chain (i.e., one DOF revolute joint at the elbow and three DOF ball and socket joints at the wrist and the shoulder). When the elbow joint is fixed, the arm kinematics changes to a serial SS chain, characterized by a spherical workspace centered at the shoulder. In order to represent motion kinematics of the elbow constrained arm, joint coordinates shown in Fig. 1 is defined [6]. The coordinate system consists of the shoulder azimuth α, the humeral elevation β, the humeral rotation γ, and the elbow flexion δ. Note that δ is fixed as a constant in this study due to the elbow joint constraint condition. For the sake of simplicity, the three wrist DOF are neglected and considered as fixed due to their minor roles in pointing motions.

The hand location $(\mathbf{X_h}, \Theta_h)^{\mathrm{T}} = (x_h, y_h, z_h, \theta_h, \phi_h, \psi_h)^{\mathrm{T}}$ in Cartesian space can be obtained by the forward kinematics:

$$\begin{bmatrix} x_h \\ y_h \\ z_h \\ \theta_h \\ \phi_h \\ \psi_h \end{bmatrix} = \begin{bmatrix} l_u s_\alpha s_\beta - l_f(s_\delta(c_\alpha s_\gamma - s_\alpha c_\beta c_\gamma) - s_\alpha s_\beta c_\gamma) \\ l_u c_\alpha s_\beta + l_f(s_\delta(s_\alpha s_\gamma + c_\alpha c_\beta c_\gamma) + c_\alpha s_\beta c_\delta) \\ -l_u c_\beta - l_f(c_\beta c_\delta - s_\beta c_\gamma s_\delta) \\ \arcsin(c_\delta(s_\alpha s_\gamma + c_\alpha c_\beta c_\gamma) - c_\alpha s_\beta s_\delta) \\ \arctan 2(s_\alpha c_\gamma - c_\alpha c_\beta s_\gamma, s_\delta(s_\alpha s_\gamma + c_\alpha c_\beta c_\gamma) + c_\alpha s_\beta c_\delta) \\ \arctan 2(c_\delta(c_\alpha s_\gamma - s_\alpha c_\beta c_\gamma) + s_\alpha s_\beta s_\delta, -c_\beta s_\delta - s_\beta c_\gamma c_\delta) \end{bmatrix}, \quad (1)$$

where l_u and l_f indicate the upper arm and the forearm link lengths, and cos(.) and sin(.) are noted as $c_{(.)}$ and $s_{(.)}$, respectively. Here, θ_h, ϕ_h and ψ_h indicate pitch, yaw and roll orientation angles of the hand that are corresponding to directions of anatomical joint articulations: wrist flexion, radial deviation and forearm pronation, respectively. From the inverse kinematics, each joint angle can be derived as [1]:

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{bmatrix} = \begin{bmatrix} \arctan 2(x_e, y_e) \\ \arccos(-z_e/l_u) \\ \arctan 2\Big(l_u(x_e y_h - x_h y_e),\, y_e(y_e z_h - y_h z_e) - x_e(z_e x_h - z_h x_e)\Big) \\ \arccos\Big((x_h^2 + y_h^2 + z_h^2 - l_u^2 - l_f^2)/(2 l_u l_f)\Big) \end{bmatrix}, \quad (2)$$

where $\mathbf{X_e} = (x_e, y_e, z_e)^{\mathrm{T}}$ depicts the elbow position in Cartesian space. In what follows we briefly present the background, described in details in [10], needed for the development of our model for approximating the elbow joint constrained hand path.

3 Higher Order Motion Specifications Defined from Relative Curvatures of Contact Geometries: Background

Let the the movement of the moving frame M, located at the wrist joint, be defined by the parameterized set of 4×4 homogeneous transforms $[T(t)] = [R(t), \mathbf{d}(t)]$ constructed from a rotation matrix $R(t)$, composed of roll $\psi(t)$, pitch $\theta(t)$ and yaw $\phi(t)$ angles, and translation vector $\mathbf{d}(t) = (d_x(t), d_y(t), d_z(t))^{\mathrm{T}}$:

$$[T(t)] = \begin{bmatrix} c_{\phi(t)}c_{\psi(t)} + s_{\theta(t)}s_{\phi(t)}s_{\psi(t)} & s_{\theta(t)}c_{\phi(t)}s_{\psi(t)} - s_{\phi(t)}c_{\psi(t)} & c_{\theta(t)}s_{\psi(t)} & d_x(t) \\ c_{\theta(t)}s_{\phi(t)} & c_{\theta(t)}c_{\phi(t)} & -s_{\theta(t)} & d_y(t) \\ s_{\theta(t)}s_{\phi(t)}c_{\psi(t)} & s_{\phi(t)}s_{\psi(t)} + s_{\theta(t)}c_{\phi(t)}c_{\psi(t)} & c_{\theta(t)}c_{\psi(t)} & d_z(t) \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3)$$

A point $\mathbf{p}$ fixed in the moving frame M traces a trajectory $\mathbf{P}(t)$ in a fixed global frame F by the $[T(t)]$ and can be approximated by the Taylor series expansion,

$$\mathbf{P}(t) = [T(t)]\mathbf{p} = \left[T_0 + T_1 t + \frac{1}{2}T_2 t^2 + \cdots\right]\mathbf{p} \quad \text{where} \quad [T_i] = \left.\frac{d^i[T(t)]}{dt^i}\right|_{t=0}. \quad (4)$$

The matrices $[T_0^j]$, $[T_1^j]$ and $[T_2^j]$ are defined by the position, velocity and acceleration of the end-effector in the vicinity of the two task positions.

Figure 2 represents a schematic plot of an elbow constrained arm, as well as the geometry of the spatial contact problem in the vicinity of a particular position. It can be assumed that the hand is in contact at three points with three spherical objects, with radii of curvature R_A, R_B and R_C, defined from object geometries at three points. The orientation angles $\theta(t)$, $\phi(t)$ and $\psi(t)$ of the moving frame M are directly derived from hand contact positions, which are obtained from the motion capture system in this study, as presented in [10]. Note that the forward kinematics (1) requires an additional step of deriving joint angles from the captured motion data through the inverse kinematics (2) to obtain orientation angles. The position coordinate transformation $[T_0]$ in (3) can be specified by the geometrically derived orientation angles:

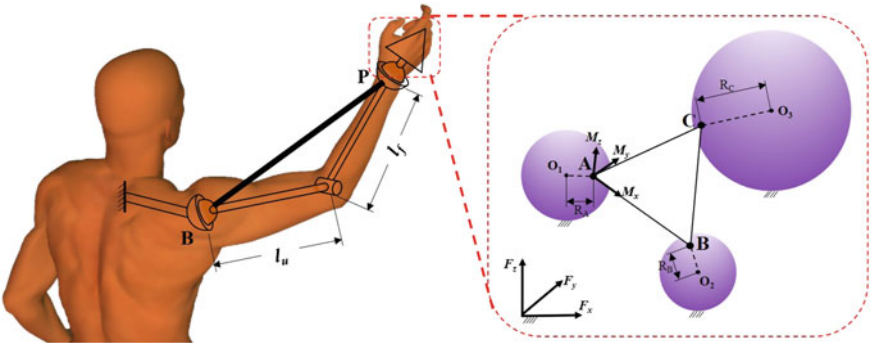


Fig. 2 Schematic plot of an elbow constrained arm with contact specifications. **B** and **P** refer positions of the base and the moving pivot, respectively

$$
\begin{aligned}
\theta(t) &= \arctan 2\left(\frac{\hat{j}\cdot(\mathbf{B}-\mathbf{A})}{|\mathbf{B}-\mathbf{A}|}, \frac{\hat{i}\cdot(\mathbf{B}-\mathbf{A})}{|\mathbf{B}-\mathbf{A}|}\right),\\
\phi(t) &= \arctan 2\left(\frac{\hat{k}\cdot((\mathbf{B}-\mathbf{A})\times(\mathbf{C}-\mathbf{A}))\times(\mathbf{B}-\mathbf{A})}{|((\mathbf{B}-\mathbf{A})\times(\mathbf{C}-\mathbf{A}))\times(\mathbf{B}-\mathbf{A})|}, \frac{\hat{k}\cdot((\mathbf{B}-\mathbf{A})\times(\mathbf{C}-\mathbf{A}))}{|((\mathbf{B}-\mathbf{A})\times(\mathbf{C}-\mathbf{A}))|}\right),\\
\psi(t) &= -\arcsin\left(\frac{\hat{k}\cdot(\mathbf{B}-\mathbf{A})}{|\mathbf{B}-\mathbf{A}|}\right),
\end{aligned}
\tag{5}
$$

where $\hat{i}$, $\hat{j}$ and $\hat{k}$ are unit vectors along each axis of the fixed frame F.

From the geometry of contact conditions shown in Fig. 2, the velocity specifications of contact points can be derived using:

$$
\begin{aligned}
\dot{\mathbf{A}} &= \mathbf{w}\times(\mathbf{A}-\mathbf{d})+\dot{\mathbf{d}} = \mathbf{w}_{\mathbf{O_1A}}\times(\mathbf{A}-\mathbf{O_1}),\\
\dot{\mathbf{B}} &= \mathbf{w}\times(\mathbf{B}-\mathbf{d})+\dot{\mathbf{d}} = \mathbf{w}_{\mathbf{O_2B}}\times(\mathbf{B}-\mathbf{O_2}),\\
\dot{\mathbf{C}} &= \mathbf{w}\times(\mathbf{C}-\mathbf{d})+\dot{\mathbf{d}} = \mathbf{w}_{\mathbf{O_3C}}\times(\mathbf{C}-\mathbf{O_3}),
\end{aligned}
\tag{6}
$$

where the moving frame's angular velocity vector $\mathbf{w}$ is a function of $\dot{\theta}$, $\dot{\phi}$ and $\dot{\psi}$. By solving (6) for $\mathbf{w}$, the velocity coordinate transformation $[T_1]$ in (4) can be specified. In the same manner, the acceleration specifications at the contact points are:

$$
\begin{aligned}
\ddot{\mathbf{A}} &= \mathbf{a}_{\mathbf{O}_1A}\times(\mathbf{A}-\mathbf{O}_1)+\mathbf{w}_{O_1A}\times(\mathbf{w}_{\mathbf{O}_1A}\times(\mathbf{A}-\mathbf{O}_1)) = \mathbf{a}\times(\mathbf{A}-\mathbf{d})+\mathbf{w}\times(\mathbf{w}\times(\mathbf{A}-\mathbf{d}))+\ddot{\mathbf{d}},\\
\ddot{\mathbf{B}} &= \mathbf{a}_{\mathbf{O}_2B}\times(\mathbf{B}-\mathbf{O}_2)+\mathbf{w}_{O_2B}\times(\mathbf{w}_{\mathbf{O}_2B}\times(\mathbf{B}-\mathbf{O}_2)) = \mathbf{a}\times(\mathbf{B}-\mathbf{d})+\mathbf{w}\times(\mathbf{w}\times(\mathbf{B}-\mathbf{d}))+\ddot{\mathbf{d}},\\
\ddot{\mathbf{C}} &= \mathbf{a}_{\mathbf{O}_3C}\times(\mathbf{C}-\mathbf{O}_3)+\mathbf{w}_{O_3C}\times(\mathbf{w}_{\mathbf{O}_3C}\times(\mathbf{C}-\mathbf{O}_3)) = \mathbf{a}\times(\mathbf{C}-\mathbf{d})+\mathbf{w}\times(\mathbf{w}\times(\mathbf{C}-\mathbf{d}))+\ddot{\mathbf{d}},
\end{aligned}
\tag{7}
$$

where $\mathbf{a}$ is the time derivative of $\mathbf{w}$. In order to calculate the acceleration coordinate transformation $[T_2]$ in (4), (7) is solved for $\mathbf{a}$ which is a function of $\ddot{\theta}$, $\ddot{\phi}$ and $\ddot{\psi}$. In this study, higher order motion specifications (i.e., linear and angular velocities and accelerations in Cartesian space) can be numerically computed from a motion capture data.

4 Elbow Constrained Trajectory Generation in Joint Space

The hand trajectory of the elbow constrained arm in the vicinity of the specified task positions can be generated by using (4). In order to produce an entire hand trajectory with a smooth speed profile, a standard robotic trajectory planning technique introduced in [2] is adopted to approximate the elbow constrained hand path, which can be kinematically modeled as a spatial SS linkage. At each task point, the inverse kinematics of the elbow constrained arm shown in (2) enables the conversion of a specified hand position into joint angles. The joint angular velocity vector $\dot{\mathbf{q}}_i = (\dot{\alpha}_i, \dot{\beta}_i, \dot{\gamma}_i, \dot{\delta}_i)^{\mathrm{T}}$ at the task point i can be solved by

$$V_i = J_i \dot{\mathbf{q}}_i, \tag{8}$$

where $V_i = (\mathbf{v}_\mathbf{i}^{\mathrm{T}}, \boldsymbol{\omega}_\mathbf{i}^{\mathrm{T}})$ is linear and angular velocity specifications of the moving frame M in Cartesian coordinates and J_i refer the Jacobian of the forward kinematics [see (1)] at the task point i. Since the Jacobian J_i is not a square matrix, a pseudo-inverse is utilized to solve (8).

The prescribed linear and angular accelerations of the moving frame M in Cartesian coordinates, $A_i = (\mathbf{a}_\mathbf{i}^{\mathrm{T}}, \boldsymbol{\alpha}_\mathbf{i}^{\mathrm{T}})$, can be mapped to a corresponding joint angular acceleration vector $\ddot{\mathbf{q}}_i = (\ddot{\alpha}_i, \ddot{\beta}_i, \ddot{\gamma}_i, \ddot{\delta}_i)^{\mathrm{T}}$ by the time derivative of (8),

$$A_i = \dot{J}_i \dot{\mathbf{q}}_i + J_i \ddot{\mathbf{q}}_i. \tag{9}$$

Since $\dot{J}_i \dot{\mathbf{q}}_i$ is known from (8), the acceleration conversion (9) can be solved with a Jacobian pseudo-inverse,

$$\ddot{\mathbf{q}}_i = [J_i^T J_i]^{-1} [J_i^T](A_i - \dot{J}_i \dot{\mathbf{q}}_i). \tag{10}$$

Following [2], a set of fifth order polynomials is defined as

$$\mathbf{q}(t) = D[1 \quad t \quad t^2 \quad t^3 \quad t^4 \quad t^5]^{\mathrm{T}}, \tag{11}$$

where the coefficient matrix D can be solved to generate a smooth joint trajectory between $(\mathbf{q}_1, \dot{\mathbf{q}}_1, \ddot{\mathbf{q}}_1)$ and $(\mathbf{q}_2, \dot{\mathbf{q}}_2, \ddot{\mathbf{q}}_2)$ over the time range $t_1 \leq t \leq t_2$.

5 Experimental Setup for Obtaining Elbow Constrained Hand Paths

In order to acquire actual human hand profiles with an elbow joint constraint, a point-to-point reaching experiment is designed. The elbow joint was immobilized at 60° of flexion by a light weight brace (Aircast Mayo Clinic Elbow Brace, DJO Global Inc., USA). The wrist joint was also fixed with a brace (AirCast A2 Wrist Stabilizing

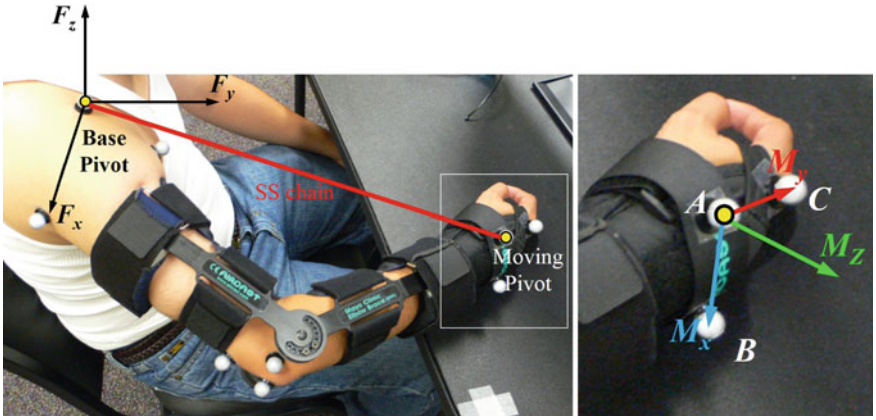


Fig. 3 Attached markers on the elbow constrained arm. Three contact points of the hand are indicated as **A**, **B** and **C**

Table 1 Task specifications captured and computed at two task points

Position spec. (mm; rad)	$(d_x, d_y, d_z; \theta, \phi, \psi)$
Initial location	(−47.10, 543.7, 51.89; −0.897, −0.853, 0.869)
Final location	(210.0, 493.5, −113.5; −0.561, −0.612, 1.260)
Velocity spec. (mm/s; rad/s)	$(\dot{d}_x, \dot{d}_y, \dot{d}_z; \dot{\theta}, \dot{\phi}, \dot{\psi})$
Initial location	(11.88, 0.8770, 1.589; 0.0314, −0.0326, 0.0696)
Final location	(−5.084, 5.984, 16.60; −0.0297, −0.0280, 0.0257)
Acceleration spec. (mm/s^2; rad/s^2)	$(\ddot{d}_x, \ddot{d}_y, \ddot{d}_z; \ddot{\theta}, \ddot{\phi}, \ddot{\psi})$
Initial location	(222.4, 24.25, −55.30; −0.562, −2.91, 3.63)
Final location	(−93.72, 41.28, 8.978; −1.35, 0.380, 1.27)

Brace, DJO Global Inc., USA). Four target points were displayed on a computer monitor and six reaching directions were defined among those targets. See [4, 5] for details. During the experiment, the motion kinematics was recorded by a 3D motion capture system (Vicon, OMG Plc., UK) with 100 Hz sampling rate. Three reflective markers were attached to each shoulder, elbow and wrist joint regions as shown in Fig. 3. The origin of the moving frame M is defined on the marker **A**. Its x axis M_x was defined as $\mathbf{B} - \mathbf{A}$, the z axis M_z is computed by $(\mathbf{B} - \mathbf{A}) \times (\mathbf{C} - \mathbf{A})$ and the y is determined by $M_z \times M_x$.

6 Comparison Between the Approximated and the Experimentally Obtained Elbow Constrained Hand Paths: A Preliminary Result

A diagonal point-to-point reaching task was selected. From the captured motion data, contact specifications of the hand in the vicinity of each task point were com-

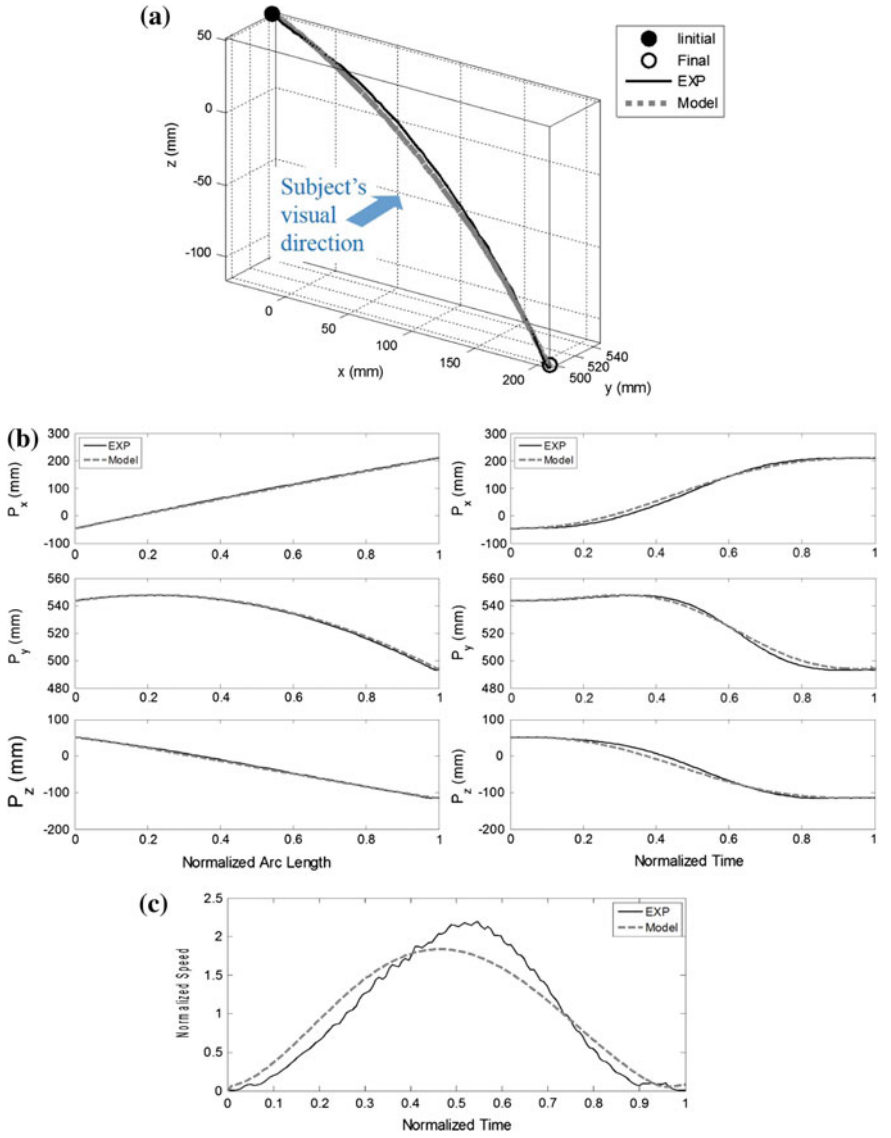


Fig. 4 Comparison of the approximated hand path with the experimental data. **a** 3D spatial path comparison **b** Coordinate comparisons with respect to the normalized arc length and the normalized time, respectively **c** Hand speed profile comparison

puted as shown in Table 1 by numerically differentiating linear positions and solving Eqs. (5–7). The contact specifications at the two task positions in Cartesian space are then converted into the joint space via Eqs. (2) and (8–10). Finally, the joint trajectory is formulated by (11), and the hand path in Cartesian space is recovered by the forward kinematics (1).

The approximated hand path profile with higher order motion constraints for the selected trial is compared with the experimental data and shown in Fig. 4. In Fig. 4a, two spatial curve geometries are compared in Cartesian space. The approximation result (see grey dotted line) closely follows the experimental hand path (see black continuous line) with no significant deviations. In order to take a closer look, two curves are compared in the geometry and the spatio-temporal perspectives. First, each curve is re-parametrized by one's arc length and its Cartesian coordinates are plotted over the normalized arc length (see the left graphs in Fig. 4b). By the re-parametrization, we can purely compare the geometries of the two curves without any temporal effects. Next, each x, y, and z axis component of the two trajectories along the normalized time is compared (see the right graphs in Fig. 4b). As shown in those detailed figures, the proposed method closely approximates both the geometry and the spatio-temporal aspects of the actual hand path. It can be noticed that deviations between the two trajectories are relatively more significant when they are compared along the normalized time than along the normalized arc length. This can be explained by different temporal characteristics reflected in their speed profiles (see Fig. 4c). Since the approximated model trajectory is formulated by the analytical solution of the minimum jerk model in joint space, it shows smoother speed profile than the experimentally obtained profile. Here, it should be noted, that despite the deviations during the reaching, the approximated and the obtained hand paths overlap in the vicinity of the two positions where the higher motion constraints have been defined.

7 Conclusion and Future Work

Hand path formulation in a point-to-point reaching is a highly redundant mapping problem in mathematics which is easily resolved by the CNS almost unconsciously. In order to explain such an efficient and optimal redundancy resolution scheme, we propose that contact conditions at the reaching hand are taken into account. These contact conditions are related to higher order kinematic task constraints such as velocities and accelerations. In order to validate our viewpoint, an elbow constrained reaching motion profile is approximated using recently developed kinematic synthesis techniques.

As a preliminary result, an approximation model output is generated for geometrical and spatio-temporal comparisons with a selected experimental data. From qualitative analysis, the model output closely follows the experimental data. Therefore, it can be considered that the CNS takes the hand contact conditions into account when it plans reaching profiles. We expect this finding can be extensively applied to approximate healthy arm reaching movements, which are highly patterned, by assuming that characteristics of governing rules within the CNS can be modeled as imaginary hand contact conditions. In addition, the future work includes conducting quantitative and statistical analysis to model the shoulder and wrist constrained paths, as well as extending the model to finger motion and object manipulation.

Acknowledgments The work has been partially supported by the NSF Grant, Award Id: IIS-1208412, sub-award Id: 2013-2908.

References

1. Biess, A., Liebermann, D.G., Flash, T.: A computational model for redundant human three-dimensional pointing movements: integration of independent spatial and temporal motor plans simplifies movement dynamics. J. Neurosci. **27**(48), 13,045–13,064 (2007)
2. Craig, J.J.: Introduction to robotics: mechanics and control. Prentice Hall, Upper Saddle River (2004)
3. Flash, T., Hogan, N.: The coordination of arm movements: an experimentally confirmed mathematical model. J. Neurosci. **5**(7), 1688–1703 (1985)
4. Moon, H., Hoang, N., Robson, N.P., Langari, R.: Human arm motion planning against a joint constraint. In: Biomedical Robotics and Biomechatronics (BioRob), 4th IEEE RAS and EMBS International Conference on 2012, pp. 401–406 (2012)
5. Moon, H., Robson, N.P., Langari, R., Buchanan, J.J.: Experimental observations on the human arm motion planning under an elbow joint constraint. In: Engineering in Medicine and Biology Society (EMBC), Annual International Conference of the IEEE, pp. 3870–3873 (2012)
6. Moon, H., Robson, N.P., Langari, R., Shin, S.: An experimental study on redundancy resolution scheme of postural configuration in human arm reaching with an elbow joint kinematic constraint. In: IEEE/EMBS Middle East Conference on Biomedical Engineering (MECBME) (2014)
7. Morasso, P.: Spatial control of arm movements. Exp. Brain Res. **42**(2), 223–227 (1981)
8. Rimon, E., Burdick, J.W.: A configuration space analysis of bodies in contact-i. 1st order mobility. Mech. Mac. Theory **30**(6), 897–912 (1995)
9. Rimon, E., Burdick, J.W.: A configuration space analysis of bodies in contact-ii. 2nd order mobility. Mech. Mach. Theory **30**(6), 913–928 (1995)
10. Robson, N., Tolety, A.: Geometric design of spherical serial chains with curvature constraints in the environment. In: ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE) (2011)
11. Robson, N.P., McCarthy, J.M.: Kinematic synthesis with contact direction and curvature constraints on the workpiece. In: ASME 2007 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 581–588. American Society of Mechanical Engineers (2007)
12. Uno, Y., Kawato, M., Suzuki, R.: Formation and control of optimal trajectory in human multijoint arm movement. Biol. Cybern. **61**(2), 89–101 (1989)

Investigation of Error Propagation in Multi-backbone Continuum Robots

Long Wang and Nabil Simaan

Abstract Snake-like robots using multiple backbones have gained increased use in surgical robotics. These robots are essentially parallel robots with constrained flexible legs. Change in the equilibrium configuration of these robots is obtained by pushing/pulling on the robots' legs. This chapter presents an investigation of the effects of assembly and home position errors on the accuracy of these robots. The assembly errors considered in this chapter include twisting about the backbone curve and deviation of the equilibrium shapes from assumed circular shapes. A modeling framework for the inverse and direct kinematics is presented while taking into account twisting and shape deviation. The configuration space and the identification Jacobian matrices are derived and then used to investigate the effects of the assembly/modeling errors on inducing errors in configuration and task space.

Keywords Kinematics · Continuum robots · Calibration · Parallel robots

1 Introduction

Continuum robots (CRs) control their shape by bending elastic components embedded in their structure [5, 7]. This chapter focuses on a specific type of continuum robot we call *multi-backbone continuum robots* (MBCRs). Figure 1 shows an MBCR as proposed in [11]. These robots are built using serially stackable *segments*. The snake segment is essentially a parallel robot having constrained

This work was supported by NSF Career grant No. IIS-1063750.

L. Wang (✉) · N. Simaan
Department of Mechanical Engineering, Vanderbilt University, Nashville, TN 37212, USA
e-mail: long.wang@vanderbilt.edu

N. Simaan
e-mail: nabil.simaan@vanderbilt.edu

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_40,

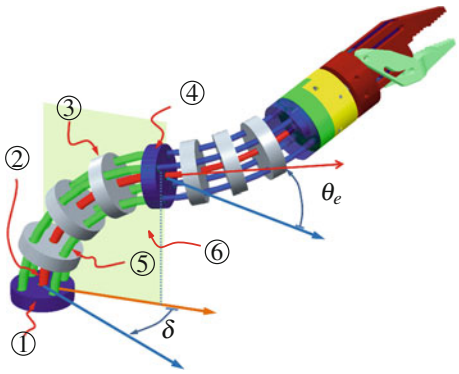


Fig. 1 Multi-backbone snake robot with two segments: *1* Base disk; *2* Primary backbone; *3* Spacer disk; *4* End disk; *5* Secondary backbone; *6* Bending plane

flexible legs in the form of superelastic NiTi beams actuated in push-pull mode. A *primary backbone* (PB) of the snake segment is surrounded by multiple *secondary backbones* (SBs) circumferentially distributed around it with equal separation angles and a constant pitch radius. The secondary backbones of each snake segment are kept equidistantly distributed around the primary backbone using spacer disks. The last disk in a snake segment is attached to all backbones while all other disks are attached only to the central backbone. Pushing or pulling on the secondary backbones allows controlled bending of the each segment in two degrees of freedom. Other design alternatives (e.g. [4]) replace the secondary backbones with wires. We call these robots *single-backbone continuum robots* (SBCRs). The simplified design of SBCR comes at a cost of limited miniaturization and payload carrying capabilities as was shown in [10]. Due to these advantages, MBCRs have been used extensively for dexterity enhancement in many surgical domains including otolaryngology, transurethral bladder surgery, and single port access surgery (see [2, 3, 12]).

Most modeling works assume *circular bending* of the continuum robot segments [15]. Few exceptions include [17] which addressed the exact kinematics and statics of MBCRs and [8, 9, 13] who addressed the exact statics and dynamics of SBCRs. However, all previous works on SBCRs and MBCRs ignore *twisting* along the central backbone curve. Additionally, they also ignore issues related to geometric assembly errors and potential bending shape deviations stemming from friction effects or modeling errors. There is a paucity of works addressing calibration of MBCRs and SBCRs.

The twisting typically stems from inexact assembly of the spacer disks on the primary backbone or from a wrong definition of the "home" position of the joints controlling the length of the secondary backbones. For example, if the SBs are assembled such that they are almost equal in length but longer than the PB then the CR segment will start from an almost straight configuration but with a twist of the secondary backbones about the primary backbone. In addition to these errors, the chapter also addresses calibration of a configuration-dependent bending shape of these robots.

Wang and Chirikjian [14] used convolution on the motion group $SE(3)$ and derived a second-order approximation using Lie algebras. Nahvi and Hollerbach [6]

presented a first-order error propagation based on error ellipsoid analysis. In this chapter, we adopt the ellipsoid analysis approach.

The contribution of this chapter stems from putting forward a generalized kinematic modeling framework allowing for evaluating errors in configuration and task space and providing the identification Jacobian matrix for calibration of MBCRs/SBCRs. A kinematic simulation of a single snake segment is presented as a case study for evaluating the pose errors caused by assembly and modeling errors.

2 Kinematic Modeling for Calibration

In this work we extend the modeling framework of [11, 16] to include *twisting* about the backbone curve and *non-circular bending* in shapes. The bending shape of each snake segment depends on the minimal energy of the elastic structure and it may deviate from circular bending depending on material distribution, friction and assembly errors. We will focus on capturing the bending behavior of a snake segment moving in *free* space (no external load).

Each snake segment has a *configuration vector* $\boldsymbol{\psi} = [\theta_e, \delta]$ where θ_e is the bending angle at the tip of the snake segment and δ is the angle characterizing the plane in which the snake segment bends, Fig. 1. We also use the arc length parameter s as measured from the base disk and we use L to denote the segment length.

Modeling non-circular bending shapes: When the segment is actuated, we assume that its exact bending shape changes as a function of its minimal energy, which may be configuration-dependent. To capture the general configuration-dependent bending behavior, we use a family of shapes given by a curvature profile function $\kappa(s, t)$. The interpolation parameter $t \in [0, 1]$ selects a specific bending shape that is interpolated from two shape generators $\kappa_0(s)$ and $\kappa_1(s)$ describing the limits of the bending workspace of the continuum segment.

$$\kappa(s, t) = t\kappa_1(s) + (1 - t)\kappa_0(s), \quad t \in [0, 1] \tag{1}$$

The relationship between t and θ_e is defined such that $t = 1$ when the snake assumes the most bent shape $\kappa_1(s)$ with a corresponding end disk angle θ_{e1} and $t = 0$ for the least bent shape $\kappa_0(s)$ with a corresponding end disk angle θ_{e0}, Fig. 2.

$$t = (\theta_e - \theta_{e0}) \,/\, (\theta_{e1} - \theta_{e0}), \quad \theta_e \in [\theta_{e1}, \theta_{e0}] \tag{2}$$

This mapping allows us to interchangeably use t instead of θ_e in the configuration space vector $\boldsymbol{\psi}$. We will therefore redefine $\boldsymbol{\psi}$ as $\boldsymbol{\psi} = [t, \delta]$.

To enable calibration of the shape behavior, the shape generator curves $\kappa_0(s)$ and $\kappa_1(s)$ are described using a modal representation:

$$\kappa_0(s) = \mathbf{a}^T \boldsymbol{\eta}, \quad \kappa_1(s) = \mathbf{b}^T \boldsymbol{\eta}, \quad \boldsymbol{\eta}(s) = [s^0, s^1, \ldots, s^n]^T \tag{3}$$

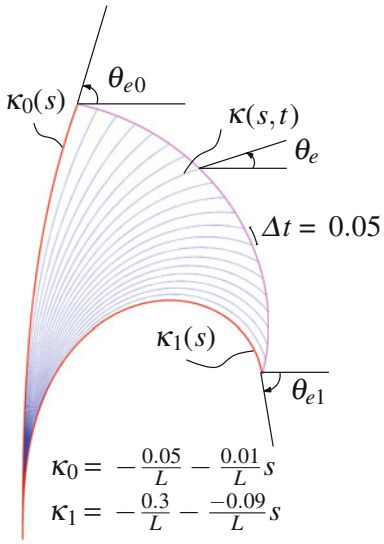


Fig. 2 Shape interpolation

where $\mathbf{a}, \mathbf{b} \in \mathbb{R}^{n+1}$ are vectors of modal factors, $\boldsymbol{\eta}(s)$ is a vector of modal functions and n is the highest degree in $\boldsymbol{\eta}(s)$. This use of the modal approach is inspired by [1] who first proposed this method for solving the inverse kinematics of hyper-redundant snake-like robots.

Modeling twist along the backbone curve: We assume that the segment has a twist angle γ_e describing the amount of twisting about PB at the end disk ($s = L$). This twist is configuration-dependent i.e. $\gamma_e = \gamma_e(\boldsymbol{\psi})$. The configuration-dependent behavior of γ_e is captured by an offset assembly error at the base disk γ_{e0} and two linear constants k_θ and k_δ describing the configuration dependence of the end disk twist angle. We also assume that the twisting about the PB is linearly distributed:

$$\gamma_e = \gamma_{e_0} + k_\theta t + k_\delta \delta, \qquad \gamma(s) = \gamma_e s/L \tag{4}$$

In the following sections we use these assumptions to evaluate kinematic sensitivity to calibration and geometric errors.

3 Inverse and Forward Kinematics

Figure 3 shows the PB and only the 1st SB for clarity. In the following we will use the notation {F} to designate a right-handed frame with unit vectors $\hat{\mathbf{x}}_f, \hat{\mathbf{y}}_f, \hat{\mathbf{z}}_f$ and $\mathbf{f}$ as its origin. Accordingly we define {B} as the base disk frame with $\mathbf{b}$ located at the center of the base disk, $\hat{\mathbf{x}}_b$ passing through the first SB and $\hat{\mathbf{z}}_b$ perpendicular to the base disk. Frame {1} characterizes the plane in which the snake segment bends and

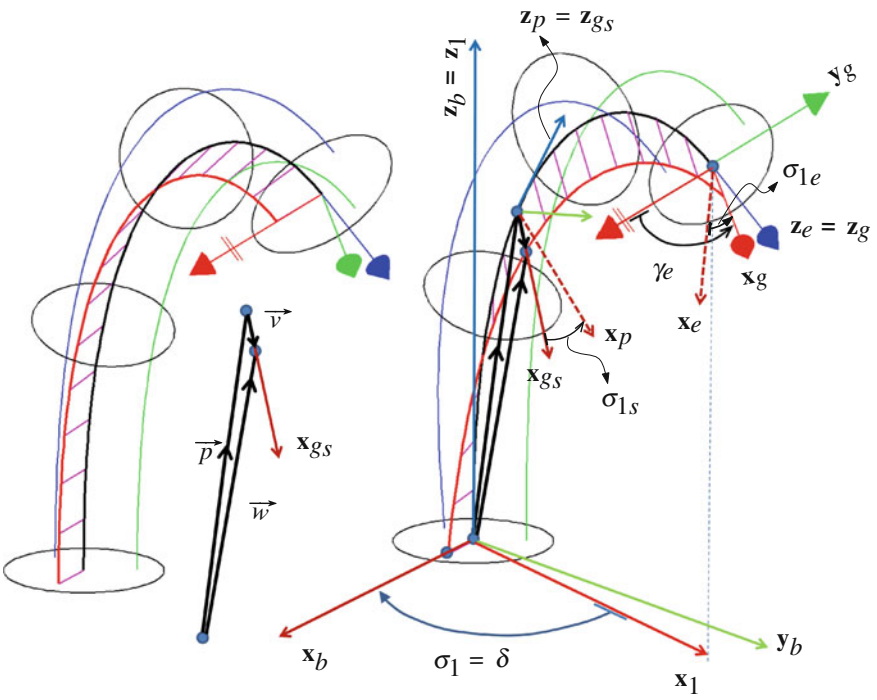


Fig. 3 Frame definitions for a single segment. *Left* shows a non twisted segment. *Right* shows a twisted segment

it is obtained by a rotation of $(-\delta)$ about $\hat{\mathbf{z}}_b$. Unit vector $\hat{\mathbf{x}}_1$ is along the projection of the PB on the plane of the base disk and $\hat{\mathbf{z}}_1 = \hat{\mathbf{z}}_b$.

To describe the end disk configuration we use frames {E} and {G}. Frame {E} is defined with $\hat{\mathbf{z}}_e$ as the normal to the end disk and $\hat{\mathbf{x}}_e$ is the intersection of the bending plane and the end disk top surface. Frame {G} is obtained by a rotation angle $(-\sigma_{1e})$ about $\hat{\mathbf{z}}_e$ which is the unit vector normal to the end disk. This angle is given by $\sigma_{1e} = \delta + \gamma_e$. Similarly, assuming three backbones, one could define $\sigma_{ie} = \delta + (i-1)\frac{2\pi}{3} + \gamma(s),\ i = 1, 2, 3$.

In addition, we define local frames {P} and $\{G_s\}$ associated with arc length s. These frames are defined in a manner similar to the definition of frames {E} and {G} but for a specific value of s as opposed to $s = L$. The origin of {P} is located at point $\mathbf{p}(s)$ and obtained by integration along the PB. We also define a vector $\mathbf{v}(s) = r\hat{\mathbf{x}}_{g_s}$ which points from $\mathbf{p}(s)$ on the PB to the corresponding point on the first SB. The location of the point on the first SB that corresponds to point $\mathbf{p}(s)$ on the PB is designated by $\mathbf{w}(s)$. We note that $\mathbf{w}(s) = \mathbf{p}(s) + \mathbf{v}(s)$.

The inverse kinematics of the snake segment solves the required lengths $L_i, i = 1, 2, 3$ of the SB's for a desired segment configuration $\boldsymbol{\psi} = (t, \delta)$. The length of the ith SB can be calculated using the tangential vector $\mathrm{d}\mathbf{w}(s)/\mathrm{d}s$,

$$L_i = \int_0^L \left\| \frac{\mathrm{d}\mathbf{w}(s)}{\mathrm{d}s} \right\| \mathrm{d}s, \quad \frac{\mathrm{d}\mathbf{w}(s)}{\mathrm{d}s} = \frac{\mathrm{d}\mathbf{p}(s)}{\mathrm{d}s} + \frac{\mathrm{d}\mathbf{v}(s)}{\mathrm{d}s} \tag{5}$$

Using the definition of curvature and expressing vectors in frame $\{1\}$, we have:

$$\frac{{}^1\mathrm{d}\mathbf{p}}{\mathrm{d}s} = [c_{\theta_s}, 0, s_{\theta_s}]^T, \quad \theta_s(s) = \frac{\pi}{2} + \int_0^s \kappa(\tau, t)\mathrm{d}\tau \tag{6}$$

where the left superscript designates the frame in which vectors are written, $c_x = \cos x$ and $s_x = \sin x$ and $\theta(s)$ designates the angle of the local tangent to the PB as measured from $\hat{\mathbf{x}}_1$ about $-\hat{\mathbf{y}}_1$ according to the right hand rule.

Since $\mathbf{v}(s)$ is coincident with $\hat{\mathbf{x}}_{g_s}$ and frame $\{G_s\}$ is rotated by σ_{1s} about the local tangent $\hat{\mathbf{z}}_{g_s} = \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}s}$, we can characterize its motion by two instantaneous rotations about $\hat{\mathbf{y}}_p$ and $\hat{\mathbf{z}}_p$. Then using Eqs. (4) and (6), we can derive the frame angular velocity ω_{g_s} and the velocity of vector $\mathbf{v}$ in Eqs. (7) and (8).

$${}^p\omega_{g_s} = {}^p\hat{\mathbf{z}}_p \frac{\mathrm{d}\gamma(s)}{\mathrm{d}s} - {}^p\hat{\mathbf{y}}_p \frac{\mathrm{d}\theta_s(s)}{\mathrm{d}s} = \left[0, -\kappa(s,t), \frac{\gamma_e}{L}\right]^T \tag{7}$$

$${}^1\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}s} = {}^1\mathbf{R}_p \left({}^p\omega_{g_s} \times {}^p\mathbf{v}\right), \quad {}^1\mathbf{R}_p = Rot\left(y, \left(\tfrac{\pi}{2} - \theta_s\right)\right), \quad {}^p\mathbf{v} = r\left[c_{\sigma_{is}}, s_{\sigma_{is}}, 0\right]^T \tag{8}$$

where $Rot\left(y, \left(\frac{\pi}{2} - \theta_s\right)\right)$ is a canonical rotation matrix about the y axis by an angle $\tilde{\theta}_s$. By substituting Eqs. (6) and (8) into Eq. (5) we obtain:

$$L_i = \int_0^L g_i(s)\,\mathrm{d}s, \quad g_i(s) = \sqrt{\left(r\kappa(s,t)c_{\sigma_{is}} + 1\right)^2 + \left(\tfrac{r\gamma_e}{L}\right)^2} \tag{9}$$

By defining the vector of joint values as $\mathbf{q} = [q_1, q_2, q_3]^T$ where $q_i = L_i - L,\ i = 1, 2, 3$ and assuming that the corresponding "zero" home positions for these joints are given by $\mathbf{q}_h = [q_{h1}, q_{h2}, q_{h3}]^T$ we can formulate the inverse kinematics problem as:

$$\mathbf{q} = \mathbf{f}(\boldsymbol{\psi}, \mathbf{k}), \quad \left\{\mathbf{f}, \mathbf{q} \in \mathbb{R}^{3\times 1}, \quad f_i = L_i(\boldsymbol{\psi}, \mathbf{k}) - L + q_{hi}\right\} \tag{10}$$

where $\mathbf{k}$ is a kinematic model parameter vector given by:

$$\mathbf{k} \triangleq [k_\theta, k_\delta, \gamma_{e0}, r, L, \mathbf{q}_h^T, \mathbf{a}^T, \mathbf{b}^T]^T \in \mathbb{R}^{(8+2(n+1))} \tag{11}$$

Equations (9) and (10) show us that the inverse kinematics of configuration space is solvable using numerical integration while it is difficult to find an analytical solution for the direct kinematics.

The direct kinematics of the snake segment requires finding $\boldsymbol{\psi}$ for a kinematically consistent value of $\mathbf{q}$. Due to its nonlinear nature of and because we are interested in investigating the sensitivity of the end disk pose to calibration errors we decided to use a numerical solution method. We therefore derive the *configuration space Jacobian* $\mathbf{J}_{\mathbf{q}\boldsymbol{\psi}} \in \mathbb{R}^{3\times 2}$ and the *identification Jacobian* $\mathbf{J}_k \in \mathbb{R}^{3\times(8+2(n+1))}$ such that $\mathbf{J}_{\mathbf{q}\boldsymbol{\psi}} \Delta\boldsymbol{\psi} = \Delta\mathbf{q}$ and $\mathbf{J}_k \Delta\mathbf{k} = \Delta\mathbf{f}\,(\boldsymbol{\psi}, \mathbf{k})$ using Eqs. (10) and (9).

$$\mathbf{J}_{\mathbf{q}\boldsymbol{\psi}} = \frac{\partial \mathbf{f}(\boldsymbol{\psi}, \mathbf{k})}{\partial \boldsymbol{\psi}} = \int_0^L \frac{\partial \mathbf{g}(s, \boldsymbol{\psi}, \mathbf{k})}{\partial \boldsymbol{\psi}}\, \mathrm{d}s, \quad \mathbf{g}(s, \boldsymbol{\psi}, \mathbf{k}) = [g_1, g_2, g_3]^T \tag{12}$$

$$\mathbf{J}_k = \frac{\partial \mathbf{f}(\boldsymbol{\psi}, \mathbf{k})}{\partial \mathbf{k}} = \int_0^L \frac{\partial \mathbf{g}(s, \boldsymbol{\psi}, \mathbf{k})}{\partial \mathbf{k}}\, \mathrm{d}s + \mathbf{g}(L, \boldsymbol{\psi}, \mathbf{k}) \frac{\partial L}{\partial \mathbf{k}} - \frac{\partial [L, L, L]^T}{\partial \mathbf{k}} \tag{13}$$

where $\frac{\partial g_i}{\partial \psi_j}$ and $\frac{\partial g_i}{\partial k_j}$ are obtained by differentiation of Eq. (9). Using $\mathbf{J}_{\mathbf{q}\boldsymbol{\psi}}$ the direct kinematics is solved using algorithm 1 based on Newton's method.

Algorithm 1 $\mathbf{f}_{\text{dir}}$ Configuration space direct kinematics

Input: $\{\mathbf{q}_{tg}, \quad \mathbf{k}\}$ **Output:** $\boldsymbol{\psi}_{tg}$ **Adjust Parameters:** $\boldsymbol{\psi}_0, \varepsilon_0 > 0, \mu > 0$

1: **START** Initialize: $\boldsymbol{\psi} \leftarrow \boldsymbol{\psi}_0, e_q \leftarrow 100\,\varepsilon_0$

2: **while** $e_q > \varepsilon_0$ **do**

3: Compute $\mathbf{q} = \mathbf{f}(\boldsymbol{\psi}, \mathbf{k}), \quad \mathbf{J}_{\mathbf{q}\boldsymbol{\psi}} = \mathbf{J}_{\mathbf{q}\boldsymbol{\psi}}(\boldsymbol{\psi}, \mathbf{k})$ using Eqs. 10 and 12.

4: Update $\{\boldsymbol{\psi}\}$

$$\mathbf{J}^+_{\mathbf{q}\boldsymbol{\psi}} = (\mathbf{J}^T_{\mathbf{q}\boldsymbol{\psi}} \mathbf{W} \mathbf{J}_{\mathbf{q}\boldsymbol{\psi}})^{-1} \mathbf{J}^T_{\mathbf{q}\boldsymbol{\psi}} \mathbf{W}, \quad \Delta\boldsymbol{\psi} = \mathbf{J}^+_{\mathbf{q}\boldsymbol{\psi}} \Delta\mathbf{q}, \tag{14}$$

5: $\Delta\mathbf{q} \leftarrow (\mathbf{q}_{tg} - \mathbf{q}), \quad \boldsymbol{\psi} \leftarrow (\boldsymbol{\psi} + \mu\Delta\boldsymbol{\psi}), \quad e_q \leftarrow \|\Delta\mathbf{q}\|$

6: **end while**

7: $\boldsymbol{\psi}_{tg} \leftarrow \boldsymbol{\psi}$ **END**

4 Evaluating Accuracy Effects of Geometric and Calibration Errors

We will evaluate the effect of calibration errors in three cases. In *case 1* we evaluate the effect of twist about the PB on the accuracy of the robot if this twist is not accounted for. In *case 2* we evaluate the effect of errors in (k_θ, k_δ) on the accuracy of the robot. Finally, in *case 3* we evaluate the effect of home position errors in $\mathbf{q}_h$ on the accuracy of the robot. In this section we used a single-segment robot with $L = 60\,\text{mm}$, $r = 4\,\text{mm}$, $\gamma_{e0} = 10°$, $\mathbf{q}_h = [1, -2, 3]\,\text{mm}$.

Case 1: effect of twist about PB. Figure 4 shows the flowchart of the simulation for evaluating the effect of an unknown twist angle γ_e on the task and configuration space errors. A joint actuation $\mathbf{q}$ is planned based on given desired configuration $\boldsymbol{\psi}_d$

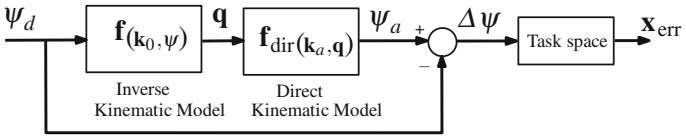


Fig. 4 Simulation flow chart

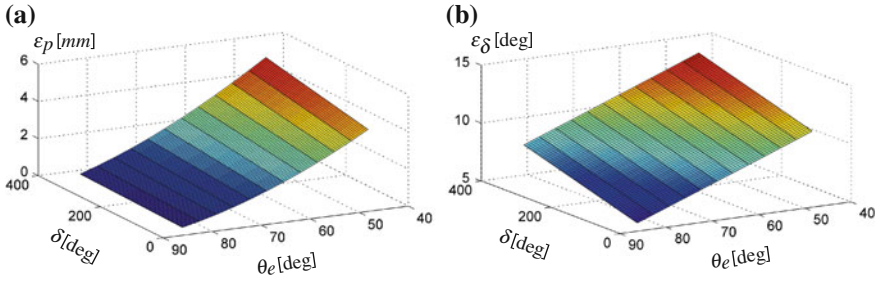


Fig. 5 Pose errors due to twist error: **a** end disk position error, **b** error in δ

and an ideal model $\mathbf{k}_0$ assuming non-twisted and circular bending shape. This joint actuation is fed into the forward kinematics model using $\mathbf{k}_{actual}$ which includes twist errors but satisfies the circular bending assumption. The error $\Delta\boldsymbol{\psi}$ is then propagated to the robot task space kinematics to evaluate the task space error. We assumed in this simulation $\gamma_e = 30°$, $k_\theta = 0.2327$, $k_\delta = 0.0185$, $a = -0.05$/L, $b = -0.8$/L. The errors between desired and actual robot pose are reported in configuration space and task space as shown in Fig. 5. For space limitations and because the maximal error in θ_e was less than 0.5° we did not include a separate for $\Delta\theta_e$. Figure 5a shows that the position error can be significant (almost 10 % of L). Figure 5b shows that the error in the bending plane direction can be as high as 14.3°.

Case 2: effect of twist gains error. To evaluate the effect of errors in k_θ and k_δ we assumed 10 % error over their nominal values as in case 1. The jacobian $\mathbf{J}_k$ was used to calculate the propagation of errors using $\Delta\mathbf{q} = \mathbf{J}_k\Delta\mathbf{k}$. The error $\Delta\boldsymbol{\psi}$ was bounded using the singular value decomposition (SVD) of $\mathbf{J}_{q\boldsymbol{\psi}}$:

$$\frac{1}{\sigma_1} \leq \left\| \frac{\Delta\boldsymbol{\psi}}{\Delta\mathbf{q}} \right\| \leq \frac{1}{\sigma_m}, \qquad [\sigma_1, \sigma_2, \ldots, \sigma_m] = \mathrm{SVD}(\mathbf{J}_{\mathbf{q}\boldsymbol{\psi}}) \tag{15}$$

Figure 6a shows the upper and lower bound for $\left\| \Delta\boldsymbol{\psi} \right\|$.

Case 3: effect of error in joint values at home position. Figure 6b shows the upper and lower bounds of the pose errors stemming from $\Delta\mathbf{q}_h = [0.1, 0.2, -0.3]^T$ [mm]. We estimated $\Delta\mathbf{q} = \mathbf{J}_k\Delta\mathbf{q}_h$ and used Eq. (15) to bound the error.

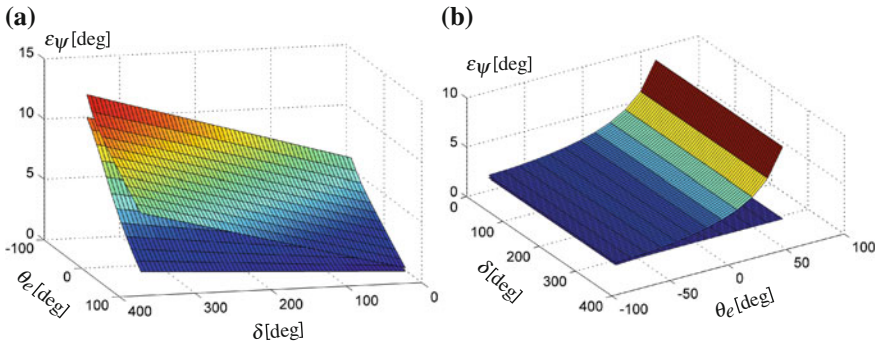


Fig. 6 Configuration error bounds: **a** error due to Δk_θ, Δk_δ, **b** error due to $\Delta \mathbf{q}_h$

5 Conclusions

This chapter presented a kinematic modeling framework that allows the derivation of generalized identification and configuration space Jacobians of MBCR's. The chapter assumed a configuration-dependant change in the twisting about the primary backbone and in the characteristic bending shape throughout the workspace. A generalized model for the inverse and forward kinematics was used along with the identification and configuration Jacobians to study the potential effect of errors including twisting about the primary backbone, error in the joint values at home position and error in the coefficients characterizing configuration-dependent shape behavior. A simulation study shows that these errors may be significant and emphasizes the need for calibration algorithms for these robots. Future work will include calibration of continuum robots using the identification Jacobian derived in this work.

References

1. Chirikjian, G., Burdick, J.: A modal approach to hyper-redundant manipulator kinematics. IEEE Trans. Robot. Autom. **10**(3), 343–354 (1994). doi:10.1109/70.294209
2. Ding, J., Goldman, R.E., Xu, K., Allen, P.K., Fowler, D.L., Simaan, N.: Design and coordination kinematics of an insertable robotic effectors platform for single-port access surgery. IEEE/ASME Trans. Mechatron. **18**, 1612–1624 (2013). doi:10.1109/TMECH.2012.2209671
3. Goldman, R.E., Bajo, A., MacLachlan, L.S., Pickens, R., Herrell, S.D., Simaan, N.: Design and performance evaluation of a minimally invasive telerobotic platform for transurethral surveillance and intervention. IEEE Trans. Biomed. Eng. **60**(4), 918–925 (2013). doi:10.1109/TBME.2012.2226031
4. Gravagne, I., Walker, I.: On the kinematics of remotely-actuated continuum robots. In: Proceedings of IEEE International Conference on Robotics and Automation, vol. 3, pp. 2544–2550. (2000). doi:10.1109/ROBOT.2000.846411
5. Hirose, S., Kado, T., Umetani, Y.: Tensor actutated elastic manipulator. In: Proceedings of the Sixth World Congress on Theory of Machines and Mechanisms, pp. 978–981 (1983)

6. Nahvi, A., Hollerbach, J.: The noise amplification index for optimal pose selection in robot calibration. In: Proceedings of IEEE International Conference on Robotics and Automation, vol. 1, pp. 647–654. (1996). doi:10.1109/ROBOT.1996.503848
7. Robinson, G., Davies, J.: Continuum robots—a state of the art. In: Proceedings of IEEE International Conference on Robotics and Automation, vol. 4, pp. 2849–2854. (1999). doi:10.1109/ROBOT.1999.774029
8. Rone, W.S., Ben-Tzvi, P.: Continuum manipulator statics based on the principle of virtual work. In: Volume 4: Dynamics, Control and Uncertainty, Parts A and B, p. 321. ASME, New York (2012). doi:10.1115/IMECE2012-87675
9. Rucker, D.C., Webster III, R.J.: Statics and dynamics of continuum robots with general tendon routing and external loading. IEEE Trans. Robot. **27**(6), 1033–1044 (2011). doi:10.1109/TRO.2011.2160469
10. Simaan, N.: Snake-like units using flexible backbones and actuation redundancy for enhanced miniaturization. In: Proceedings IEEE International Conference on Robotics and Automation, pp. 3012–3017. IEEE, Barcelona (2005). doi:10.1109/ROBOT.2005.1570572
11. Simaan, N., Taylor, R., Flint, P.: A dexterous system for laryngeal surgery. In: Proceedings of IEEE International Conference on Robotics and Automation, pp. 351–357 vol. 1. IEEE, New Orleans (2004). doi:10.1109/ROBOT.2004.1307175
12. Simaan, N., Xu, K., Wei, W., Kapoor, A., Kazanzides, P., Flint, P., Taylor, R.: Design and integration of a telerobotic system for minimally invasive surgery of the throat. Int. J. Robot. Res. **28**(9), 1134–1153 (2009). doi:10.1177/0278364908104278
13. Tatlicioglu, E., Walker, I.D., Dawson, D.M.: Dynamic modelling for planar extensible continuum robot manipulators. In: Proceedings of IEEE International Conference on Robotics and Automation, pp. 1357–1362. IEEE (2007). doi:10.1109/ROBOT.2007.363173
14. Wang, Y., Chirikjian, G.S.: Nonparametric second-order theory of error propagation on motion groups. Int. J. Robot. Res. **27**(11–12), 1258–1273 (2008). doi:10.1007/978-3-540-68405-3_10
15. Webster, R.J., Jones, B.A.: Design and kinematic modeling of constant curvature continuum robots: a review. Int. J. Robot. Res. **29**(13), 1661–1683 (2010). doi:10.1177/0278364910368147
16. Xu, K., Simaan, N.: An investigation of the intrinsic force sensing capabilities of continuum robots. IEEE Trans. Robot. **24**(3), 576–587 (2008). doi:10.1109/TRO.2008.924266
17. Xu, K., Simaan, N.: Analytic formulation for kinematics, statics and shape restoration of multibackbone continuum robots via elliptic integrals. ASME J. Mech. Robot. **2**, 11006–11013 (2010) doi:10.1115/1.4000519

Kinematics of Expansive Planar Periodic Mechanisms

Ciprian S. Borcea and Ileana Streinu

Abstract A flexible bar-and-joint framework is said to be moving expansively if the distance between any two of its joints either increases or stays the same. Expansive motions of finite 2D frameworks have been fully characterized. Here, we investigate their periodic counterparts. The key to their understanding is a family of one-degree-of-freedom mechanisms called *periodic pointed pseudo-triangulations.* Expansive infinitesimal motions for mechanisms with several degrees of freedom form a polyhedral cone whose extremal rays are obtained from different completions of the framework to pseudo-triangulations. We illustrate its structure on a framework associated to a stellated tiling of the plane.

Keywords Periodic framework · Repetitive assembly · Expansive deformation

1 Introduction

In this chapter we study the kinematics of a remarkable family of *planar periodic bar-and-joint frameworks*: those which possess periodic expansive deformations. For instance, we show that the periodic framework from Fig. 1 has locally a smooth 4-dimensional space of periodic deformations and all directions for expansive trajectories are contained in a polyhedral cone which has a natural geometric description and can be determined with precision. By definition, a one-parameter deformation of a flexible framework is *expansive* if the distance between any pair of joints either increases or stays the same. Taken in reverse, an expansive motion is *contractive*.

C. S. Borcea
Department of Mathematics, Rider University, Lawrenceville, NJ 08648, USA
e-mail: borcea@rider.edu

I. Streinu (✉)
Department of Computer Science, Smith College, Northampton, MA 01083, USA
e-mail: istreinu@smith.edu

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_41,

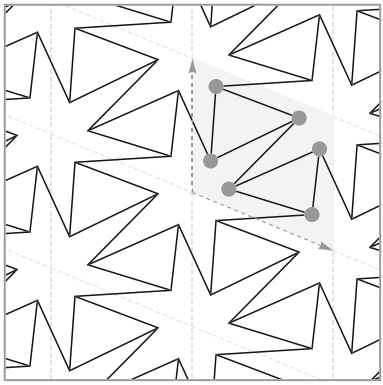


Fig. 1 A periodic bar-and-joint framework

Of particular interest are those one-degree-of-freedom (1dof) mechanisms which are expansive in some neighborhood of the initial configuration. We show that all expansive deformations are, infinitesimally, linear combinations with non-negative coefficients of underlying 1dof expansive mechanisms.

Expansion is a type of kinematic behavior with multiple applications which include deployable structure design and nano-mechanics. A popular example of a truss structure with reversible expansion/contraction properties is Hoberman's sphere [8]. Kovacs et al. [9] describe a kinematic model of a virus and argue that it has expansive properties. More recently, Tanaka et al. [17, 18] study *repetitive assemblies* with expanding properties. Expanding or auxetic features, as considered in the materials science literature [4–6, 10, 12] have been shown in [3] to be implied by the stronger *expansive* property as defined above. In dimension two, finite expansive framework deformations have applications to robot arm motion planning [14, 15] and are well-understood mathematically [13–15]. One-degree-of-freedom expansive frameworks arise from a planar pointed pseudo-triangulation with a convex hull edge removed [15]. In the expansive interval, such a framework has a smooth configuration space. Infinitesimal expansive motions of mechanisms with more than one degree-of-freedom form a polyhedral cone [13] whose extremal rays correspond to refinements with just 1dof of the given linkage. Our purpose here is to demonstrate a similar structure underlying planar *periodic* frameworks.

2 Preliminaries: Periodic Frameworks and Deformations

2.1 Planar Periodic Frameworks

A *periodic bar-and-joint framework* (G, Γ, p, π) in the plane is given by an infinite graph G, a periodicity group Γ acting on G, a placement p of the vertices of G in the Euclidean plane and a representation π of the periodicity group Γ by a lattice

of translations. The edges are viewed as rigid bars: they may rotate freely around their incident joints, and maintain their lengths during framework deformations. The graph $G = (\mathsf{V}, \mathsf{E})$ has an infinite set of vertices V and (unoriented) edges E and is connected. The *periodicity group* Γ is a free Abelian group of rank two acting on G without fixed vertices or fixed edges. We assume that the quotient multigraph G/Γ is *finite*, and denote the number of vertex and edge orbits by $n = |\mathsf{V}/\Gamma|$ and $m = |\mathsf{E}/\Gamma|$. For example, the periodic framework in Fig. 1 has 6 vertex orbits and 9 edge orbits. The function $p : \mathsf{V} \to \mathbb{R}^2$ gives a specific placement of the vertices as points in the plane, in such a way that any two vertices joined by an edge in E are mapped to distinct points. The placement is *periodic* in the obvious sense that the abstract action of the periodicity group Γ is replicated by the action of the periodicity lattice $\Lambda = \pi(\Gamma)$ on the placed vertices.

2.2 Periodic Deformations

A one-parameter *periodic deformation* is a family of placements and a family of lattices parametrized by time $(p(t), \pi(t))_t$, such that all bar lengths are maintained and the same abstract periodicity group Γ acts on all the frameworks of the deformed family. A periodic framework is *rigid* if it has no periodic deformations other than the trivial ones resulting from Euclidean isometries. The configuration space of the periodic framework is obtained from the placements of vertex orbits, subject to the algebraic constraints of prescribed (squared) lengths for edges. We factor out the 3-dimensional space of planar rigid transformations. This concept of periodic deformation was introduced in [1]. A framework is rigid when corresponding to an isolated point of the configuration space; otherwise it is flexible. After choosing vertex representatives for all vertex orbits and two generators for the periodicity lattice, the Jacobean matrix at a given placement p is a $(2n + 4) \times m$ matrix (called the *periodic rigidity matrix* and denoted by R) whose rank thus cannot exceed $2n + 1$. At a regular point the rank of the Jacobean equals the dimension of the configuration space in a small neighborhood. We say that a periodic framework is *infinitesimally rigid* if its periodic rigidity matrix has the maximum rank of $2n + 1$. In this case, the framework must have at least $2n + 1$ edges, properly placed. In [2], we have characterized the graphs which are periodically minimally rigid, when generically placed. A finite graph is said to be of "$2n - 2$"-sparsity type if it has exactly $2n - 2$ edges (where n is its number of vertices), and any of its subsets of $n' \leq n$ vertices spans at most $2n' - 2$ edges.

Theorem 1 [2] *A (multi)graph with* $2n + 1$ *edges (on n vertices) is the quotient graph of a minimally rigid periodic framework if and only if it contains a subgraph of* $2n - 2$ *sparsity type spanning all the vertices.*

A framework is *minimally rigid* if it is infinitesimally rigid and the removal of any edge turns it into a flexible framework. Classical arguments can be used to show

that infinitesimal (periodic) rigidity implies (periodic) rigidity and that, if a periodic framework is obtained from a minimally rigid one by the removal of k edges then the rank of its rigidity matrix is $2n + 1 - k$ and its deformation space has dimension k in a neighborhood of the given placement. We say, in this case, that the framework has k degrees of freedom.

2.3 Periodic Expansive Motions

A *one-parameter deformation* $(p(t), \pi(t))_{t\in(-\varepsilon,\varepsilon)}$ of a flexible framework is said to be *expansive* if, as the time t increases, all the distances between pairs of vertices increase or stay the same. Here, $(p(0), \pi(0))$ gives the initial framework and the corresponding *infinitesimal deformation* is the tangent vector to the deformation space given by the derivative at 0.

We describe now a family of planar periodic frameworks distinguished by two elementary and easy to verify properties.

2.4 Non-crossing Periodic Graphs

A periodic framework is *non-crossing* if all pairs of non-incident bars are disjoint (they do not cross, touch or overlap). All the frameworks illustrated in this chapter are non-crossing. A non-crossing framework subdivides the plane into two-dimensional regions, called faces. The periodicity group of the given framework also acts on its set of faces. The examples in Fig. 2 have 3, 2, 3 and 3 face orbits, colored distinctly. Euler's formula $n - m + f = 0$ for the torus relates the numbers n, m and f of vertex, edge and face orbits.

2.5 Periodic Pseudo-Triangulations

A *pseudo-triangle* is a simple closed planar polygon with exactly three internal angles smaller than π. A set of vectors is *pointed* if no subset allows a linear combination with strictly positive coefficients that sums to 0. Equivalently, for a pointed set of vectors, some consecutive pair (in the circular rotational order around the common vertex) induces an angle larger than π. A planar non-crossing periodic framework is a *periodic pointed pseudo-triangulation* when all faces are pseudo-triangles and the framework is pointed at every vertex. An illustration for $n = 3$ is given in Fig. 2d.

Proposition 2 [3] *A periodic pseudo-triangulation has $m = 2n$, that is, the number of edge orbits $m = card(\mathsf{E}/\Gamma)$ is twice the number of vertex orbits $n = card(\mathsf{V}/\Gamma)$.*

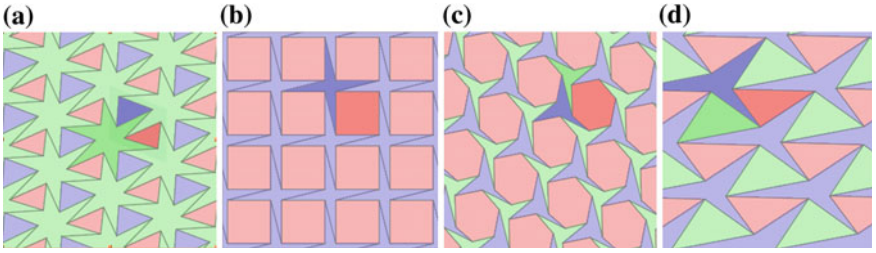


Fig. 2 Four pointed periodic tilings from [7]

Combined with Theorem 1, this proposition shows that periodic pointed pseudo-triangulations have the right number of edges to be smooth one-degree-of-freedom periodic mechanisms. The fact that this is indeed the case was proved in [3] based on our generalization to the periodic setting of Maxwell's Theorem [11] on liftings and stresses of planar bar-and-joint frameworks, where we showed that a periodic pseudo-triangulation cannot have nontrivial periodic stresses.

Proposition 3 [3] *The local deformation space of a periodic pseudo-triangulation is smooth and one-dimensional and continues to be so as long as the deformed framework remains a pseudo-triangulation. This is true for any relaxation of periodicity* $\tilde{\Gamma} \subset \Gamma$ *of finite index.*

Finally, we have proved the following, most remarkable property of periodic pseudo-triangulations.

Theorem 4 [3] *Let* (G, Γ, p, π) *be a planar periodic pseudo-triangulation. Then the framework has a one-parameter periodic deformation, which is expansive for the entire open interval where it remains a pseudo-triangulation.*

In the rest of this chapter, we extend this result from periodic pointed pseudo-triangulations to arbitrary pointed and non-crossing periodic frameworks and show how to *design* expansive trajectories for them. Ultimately, we obtain a complete characterization of the frameworks which support expansive motions. In addition, we give a precise procedure for calculating the set of *all* possible infinitesimal expansive motions of a given framework, which we show to be a polyhedral cone, called the *cone of expansive infinitesimal motions*. Expansive trajectories can be obtained by integrating an appropriate vector field of expansive infinitesimal motions belonging, at each point, to the corresponding cone.

3 Designing Expansive Trajectories: Examples

We illustrate the theory presented so far with the four periodic frameworks from Fig. 2, where they are depicted with colored face orbits and highlighted face representatives to facilitate the visual identification of the periodicity lattice. These

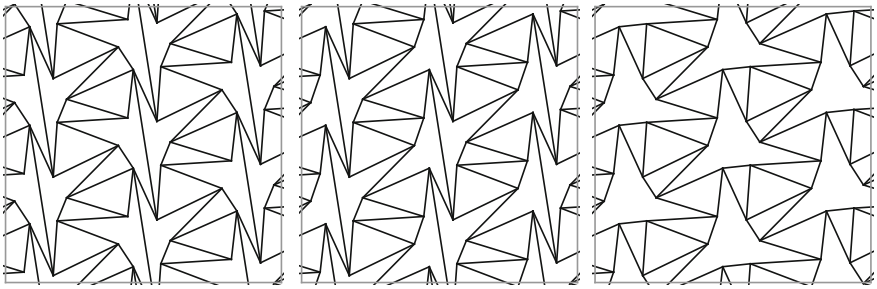


Fig. 3 The periodic framework from Fig. 2a can be turned into pointed pseudo-triangulations by the addition of three edges on the large stellated face, placed in 14 distinct ways (shown here up to symmetries)

examples are the stellated planar tilings of [7] (p. 239). They are all non-crossing and pointed, but only (d) is a pseudo-triangulation.

3.1 Counting Degrees of Freedom

An argument similar to the one used in [3] (based on our periodic extension of Maxwell's Theorem) can be applied to show that none of the examples in Fig. 2 (more generally, no periodic non-crossing and pointed framework) supports a periodic stress. In particular, this implies that Theorem 1 can be applied to compute their degrees of freedom, as follows.

Framework (a) has $n = 6, m = 9, f = 3$, and $k = 2n + 1 - m = 4$ dofs. Framework (b) has $n = 4, m = 6, f = 2$, and $k = 2n + 1 - m = 3$ dofs. Framework (c) has $n = 6, m = 9, f = 3$, and $k = 2n + 1 - m = 4$ dofs. Framework (d), which has $n = 3, m = 6, f = 3$, and $k = 2n + 1 - m = 1$, has a well-defined one-parameter expansive trajectory. This framework is a deformed configuration of the familiar Kagome framework [3, 16]. Our goal now is to explain our approach for designing expansive trajectories for the other three frameworks (a), (b) and (c).

3.2 Subdividing Faces

The faces of a periodic non-crossing and pointed framework which are not pseudo-triangles can be subdivided by new edges which maintain non-crossing and pointedness. This is always possible (Theorem 5 below), but not uniquely. Figure 3 illustrates three of the ways in which the large stellated face of the framework from Fig. 2a can be subdivided; the others are obtained by applying appropriate symmetries to these three types of constructions, for a total of 14 possibilities.

3.3 Designing Expansive Trajectories

For the example under discussion, each of the 14 pseudo-triangulations induces a distinct expansive trajectory of the original framework. However, these are not the only possibilities. One may imagine the following scenario: start with one pseudo-triangulation and deform the original framework for a small time step δt according to its induced trajectory. Since the points have not moved too much, there will still be 14 ways of pseudo-triangulating the deformed framework, so now we may choose a different one. This can be continued for as long as each of the intermediate pseudo-triangular frameworks retains its pointedness. We remark that no crossings of edges will occur during an expansive motion: joints move away from each other, by definition, and the same holds for arbitrary points on the edges.

This scenario can be further refined. If we make the time step δt infinitesimally small, we may look not just at the finite motions induced by the pointed pseudo-triangular completions of the original framework, but also at the corresponding infinitesimal expansive motions.

3.4 Cone of Infinitesimal Expansive Motions

Given an infinitesimal deformation and a pair of vertices, the condition expressing the infinitesimal increase of the squared distance between the vertices is a linear inequality: infinitesimal deformations on one side of the equality subspace produce infinitesimal increase and on the other side infinitesimal decrease. It follows that all expansive infinitesimal motions must lie in a polyhedral cone with the apex at the origin which is the intersection of all half-spaces determined by pairs of vertices. The extremal rays of this cone correspond to the possible refinements of the framework to periodic pseudo-triangulations. For the example in Fig. 2a there are 14 possibilities (Fig. 3), hence a section of the cone away from the origin will result in a convex polyhedron in R^3, with 14 vertices (Fig. 4). The combinatorics of this polyhedron can be explained as follows: each face corresponds to adding one more (non-crossing, pointedness respecting) edge-orbit to the given framework (in 9 ways); an edge of the polyhedron corresponds to adding two edge-orbits to the framework (21 ways), and a vertex of the polyhedron corresponds to adding three edges, i.e. to one of the 14 ways in which the framework can be completed to a pseudo-triangulation.

3.5 Convex Faces and Rigid Components

The frameworks (b) and (c) in Fig. 2 each have a convex face with more than 3 vertices. Such faces can be subdivided in many ways, but when all the possible edges have been added, the result is a triangulation of the convex face. Since each triangle is rigid, the

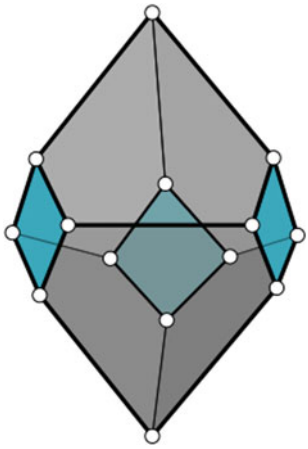


Fig. 4 A section of the cone of infinitesimal expansive motions for the 4dof framework from Fig. 2a

triangulated face becomes a (periodically repeated) *rigid component*. The pseudo-triangulation ultimately obtained will expand in a manner that does not depend on how the convex face was triangulated. Therefore, although the framework in Fig. 2b can be extended in 4 ways (two ways for the convex face and 2 ways for the non-convex face) to a pseudo-triangulation, only two of them lead to distinct expansive trajectories. The framework in Fig. 2c has only one face that can be further subdivided, and it is convex: this framework supports exactly one expansive trajectory, in spite of the fact that it can also be pseudo-triangulated in 14 ways.

4 Kinematics of Periodic Expansive Motions

We now present a complete characterization of the frameworks which support expansive motions. First, we show that any periodic non-crossing and pointed framework can be extended to a pointed pseudo-triangulation by subdividing faces. The next goal is to understand the rigid components, as we have already seen in the previous examples that they play a role in determining the expansive behavior of a periodic non-crossing and pointed framework.

4.1 Extending a Periodic Non-crossing Pointed Framework to a Pseudo-Triangulation

There is a simple procedure for *designing* frameworks similar to those shown in Fig. 3: start with an arbitrary periodic point set in *general position* (i.e. triplets of points are collinear only when they belong to the same orbit). Then insert edge

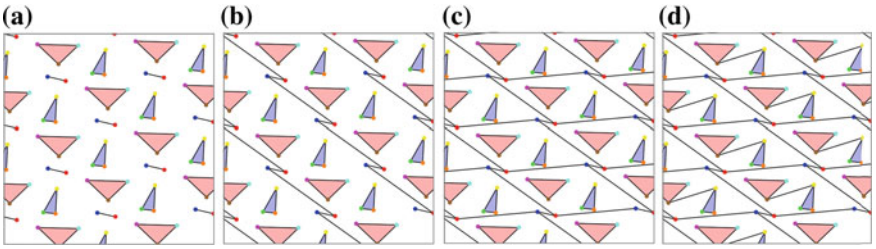


Fig. 5 Extending a periodic pointed non-crossing framework which is not a pseudo-triangulation. **a** All graph cycles are trivial homology cycles. **b** Cycles span a one-dimensional homology subspace. **c**, **d** Cycles span the full (rational) homology group of the torus

representatives, one by one, maintaining non-crossing and pointedness, and replicate them periodically. The following theorem proves the correctness of this procedure.

Theorem 5 *Let G be a non-crossing and pointed periodic framework which is not a pseudo-triangulation. Then there exists a new edge orbit which can be added, while maintaining the non-crossing and pointedness of the framework.*

Proof Compared to the proof for finite pseudo-triangulations [15], in the periodic setting we have to show that it is not possible that the only non-crossing edges that could be inserted have endpoints in the same orbit. The proof proceeds through a case analysis of three possible situations, differentiated by the nature of the cycles of the quotient graph G/Γ, when viewed as a graph embedded on the (flat) torus: (a) all graph cycles are trivial homology cycles; (b) cycles span a one-dimensional homology subspace and (c) cycles span the full homology group of the torus. We now reason in the Euclidean plane and for the infinite framework G. In case (a), G is disconnected and all its connected components (which repeat periodically) are finite frameworks (Fig. 5a). Vertices in the same orbit do not belong to the same component, and the planar subdivision induced by G has exactly one unbounded face F. If two such connected components are *visible* to each other, then a standard geometric argument as in [15], based on (piecewise linear) geodesic paths, shows the existance of a *tangent* edge. Since it lies in the unbounded face, the tangent does not cross any existing edge, and since it is tangent, its endpoints are pointed, hence it satisfies the conclusion. In case (b) (illustrated in Fig. 5b), G is still disconnected but at least one of its connected components (as a subgraph) is infinite (and so are all of its periodically repeated copies). The existence of a tangent segment, with endpoints lying on two different connected components, and not in the same vertex orbit, follows by taking a geodesic path between two non-adjacent, inner convex vertices on the infinite face: such a path is piecewise linear and contains at least one tangent edge. Finally, in case (c) (illustrated in Fig. 5c, d) all the face cycles of the periodic graph G are finite and thus enclose polygonal regions (which may have holes). However, such a face cycle may contain vertices belonging to the same orbit (Fig. 5c, d). If such a face is not an empty pseudo-triangle, then it will have an internal tangent along the geodesic path joining two inner convex vertices. All that remains

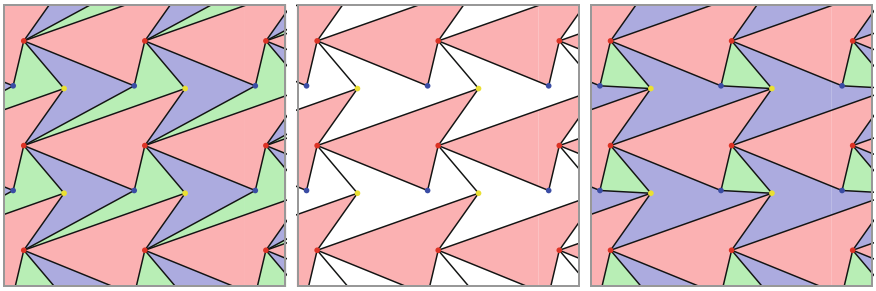


Fig. 6 A flip in a periodic pointed pseudo-triangulation

to be shown is that the two *endpoints* of one such tangent are not in the same vertex orbit. This follows from two observations: (a) on any simple polygonal cycle, the vertices in the same orbit appear with an inner reflex angle at most once, and (b) if an edge is tangent to a polygonal chain, then the inner angles of the polygonal cycle at these endpoints are both reflex.

The examples in Fig. 5 (where vertices of the same color indicate that they are in the same orbit) have been chosen to illustrate the properties used in the proof of the theorem. Each subsequent framework is obtained by inserting a *tangent* in the previous one.

As a corollary we obtain:

Corollary 6 (Flips in pseudo-triangulations) *If we remove an edge orbit from a periodic pointed pseudo-triangulation, then there always exists a different edge orbit that can be added to obtain another pointed pseudo-triangulation.*

Proof We use an idea from finite pointed pseudo-triangulations [15], namely that the removal of one edge creates a face with four inner convex angles (as in Fig. 6, middle), which can be pseudo-triangulated in two ways by two distinct tangents (Fig. 6, left and right). The argument from case (c) of Theorem 5 shows that the endpoints of these tangents belong to distinct vertex orbits.

4.2 Kinematically Equivalent Frameworks

A flexible framework decomposes into rigid parts called *rigid components* (Fig. 7). Two periodic frameworks on the same point set are *kinematically equivalent* if one is obtained from the other by placing differently the bars inside rigid components, while maintaining them rigid. Such frameworks have the same configuration space. An example is illustrated in Fig. 7. Since adding a bar to a rigid component does not

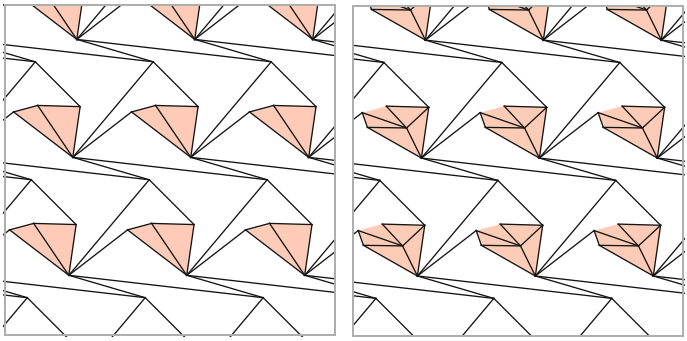


Fig. 7 A rigid component in a periodic pseudo-triangulation and a kinematically equivalent framework

change the deformation space, we assume that our frameworks are *non-redundant*, i.e. they use the minimum number of bars on all rigid components.

Proposition 7 *A rigid component of a periodic pseudo-triangulation is a finite pointed pseudo-triangulation; in particular, it is contained in its convex hull.*

Proof We only need to prove that a rigid component is finite. Then we apply a result from [15] about finite pointed pseudo-triangulations, which are contained in their convex hulls. An infinite rigid component must be, as an induced graph, periodically rigid, hence its quotient graph (on n' vertices) has $2n'+1$ edges. As an induced graph, it is still pointed and non-crossing, hence it has at most $2n'$ edges, a contradiction. Hence there can't be any infinite rigid components.

Theorem 8 *A non-redundant periodic framework is an expansive 1dof mechanism if and only if it is a periodic pointed pseudo-triangulation, or is kinematically equivalent to one. A periodic framework has an expansive deformation if and only if it is pointed and non-crossing or is kinematically equivalent to one.*

Proof Indeed, if a set of points belong to a rigid component, the way they are interconnected does not matter (they stay at the same distance anyway), so we can replace the interconnecting pseudo-triangulation with any finite minimally rigid (Laman) graph. The construction may violate pointedness and it may be self-intersecting, but only in the interior region of the rigid component convex hull. Figure 7 illustrates the idea.

4.3 The Cone of Infinitesimal Expansive Motions of a Periodic Non-crossing and Pointed Framework

It is possible to compute with precision how many, and which edges can be used for subdividing a face into pseudo-triangles. This information is related to a

combinatorial characterization of the polyhedral cone of expansive motions, and is a natural generalization of a similar result for the finite case [13].

Theorem 9 *A planar periodic non-crossing and pointed framework with n vertex orbits and $m = 2n - k$ edge orbits has a smooth local deformation space of dimension $k + 1$ and allows expansive deformation trajectories. The set of all possible directions for these expansive trajectories forms a polyhedral cone in the infinitesimal deformation space of the given framework. This cone of expansive infinitesimal motions has dimension at most $k + 1$ and all its extremal rays are obtained from completions of the framework to periodic pointed pseudo-triangulations.*

Proof (Sketch) The first statement is a direct consequence of the previous discussion and of the results in [2]. The cone of expansive motions is given as the half-space intersection of the set of all linear inequalities that express the property of infinitesimal expansiveness for a pair of vertices in the periodic framework. Corollary 6 characterizes the edges of the cone. The extension to faces of all dimensions is a direct generalization of the argument used in the finite case [13].

As a final remark, we point out that if a non-crossing and pointed periodic framework has convex faces, these faces must be rigid in any *expansive* deformation. Any triangulation of a convex face will serve the purpose of rigidifying it.

5 Conclusion

In this chapter, we have characterized those planar non-crossing periodic bar-and-joint frameworks which allow an expansive deformation trajectory. Those with a single degree of freedom are the periodic pointed pseudo-triangulations or are obtained from them by simple replacement operations on rigid components. For several degrees of freedom, we investigated all completions of the framework to 1dof expansive mechanisms (up to kinematic equivalence). Infinitesimally, they provide the extremal rays of the cone of infinitesimal expansive motions.

Acknowledgments The authors are partially supported by National Science Foundation grant CCF-1319366. This research was conducted while visiting Technische Universität München, with funding for the second author from the DFG-Collaborative Research Center TRR109, *Discretization in Geometry and Dynamics.*

References

1. Borcea, C.S., Streinu, I.: Periodic frameworks and flexibility. Proc. Roy. Soc. Lond. A **466**, 2633–2649 (2010)
2. Borcea, C.S., Streinu, I.: Minimally rigid periodic graphs. Bull. Lond. Math. Soc. **46**, 1093–1103 (2011)

3. Borcea, C.S., Streinu, I.: Liftings and stresses of periodic frameworks. In: Proceedings of the 30th Annual Symposium on Computational Geometry (SoCG'14), Kyoto, June (2014) (To appear)
4. Greaves, G.N., Greer, A.I., Lakes, R.S., Rouxel, T.: Poisson's ratio and modern materials. Nat. Mater. **10**, 823–837 (2011)
5. Grima, J.N., Alderson, A., Evans, K.E.: Auxetic behaviour from rotating rigid units. Phys. Status Solidi B **242**, 561–575 (2005)
6. Grima, J.N., Chetcuti, E., Manicaro, E., Attard, D., Camilleri, M., Gatt, R., Evans, K.E.: On the auxetic properties of generic rotating rigid triangles. Proc. Roy. Soc. A **468**, 810–830 (2012)
7. Grünbaum, B., Shephard, G.C.: Tilings by regular polygons. Math. Mag. **50**, 227–247 (1977)
8. Hoberman, C.: Reversibly expandable doubly-curved truss structure. US Patent no. 4–942-700 (1991)
9. Kovacs, F., Tarnai, T., Guest, S.D., Fowler, P.W.: Double-link expandohedra: a mechanical model for expansion of a virus. Proc. Roy. Soc. Lond. A **460**, 3191–3202 (2004)
10. Lakes, R.: Foam structures with a negative Poisson's ratio. Science **235**, 1038–1040 (1987)
11. Maxwell, J.C.: On reciprocal figures, frameworks and diagrams of forces. Trans. Roy. Soc. Edinb. **26**, 1–40 (1870)
12. Mitschke, H., Robins, V., Mecke, K. Schröder-Turk, G.E.: Finite auxetic deformations of plane tessellations. Proc. Roy. Soc. A **469**, 20120465 (2013)
13. Rote, G., Santos, F., Streinu, I.: Expansive motions and the polytope of pointed pseudo-triangulations. Discrete Comput. Geom. **25**, 699–736 (2003)
14. Streinu, I.: A combinatorial approach to planar non-colliding robot arm motion planning. In: Proceedings of ACM/IEEE Symposium on Foundation of Computer Science (FOCS), pp. 443–453 (2000)
15. Streinu, I.: Pseudo-triangulations, rigidity and motion planning. Discrete Comput. Geom. **34**, 587–635 (2005)
16. Sun, K., Souslov, A., Mao, X., Lubensky, T.C.: Surface phonons, elastic response, and conformal invariance in twisted kagome lattices. Proc. Nat. Acad. Sci. **109**, 12369–12374 (2012)
17. Tanaka, H.: Bi-stiffness property of motion structures transformed into square cells. Proc. Roy. Soc. A **469**, 20130063 (2013)
18. Tanaka, H., Shibutani, Y., Izumi, S., Sakai, S.: Planar mobility modes of 8-bar-jointed structures with a single degree of freedom. Int. J. Solids Struct. **49**, 1712–1722 (2012)

From Inverse Kinematics to Optimal Control

Perle Geoffroy, Nicolas Mansard, Maxime Raison, Sofiane Achiche and Emo Todorov

Abstract Numerical optimal control (the approximation of an optimal trajectory using numerical iterative algorithms) is a promising approach to compute the control of complex dynamical systems whose instantaneous linearization is not meaningful. Aside from the problems of computation cost, these methods raise several conceptual problems, like stability, robustness, or simply understanding of the nature of the obtained solution. In this chapter, we propose a rewriting of the Differential Dynamic Programing solver. Our variant is more efficient and numerically more interesting. Furthermore, it draws some interesting comparisons with the classical inverse formulation: in particular, we show that inverse kinematics can be seen as singular case of it, when the preview horizon collapses.

Keywords Optimal control · Inverse kinematics · Differential dynamic programming

P. Geoffroy (✉) · M. Raison · S. Achiche
Ecole Polytechnique de Montréal, Montréal, Canada
e-mail: perle.geoffroy@polymtl.ca

M. Raison
e-mail: maxime.raison@polymtl.ca

S. Achiche
e-mail: sofiane.achiche@polymtl.ca

N. Mansard
LAAS-CNRS, University of Toulouse, Toulouse, France
e-mail: nicolas.mansard@laas.fr

E. Todorov
University of Washington, Seattle, USA
e-mail: todorov@cs.washington.edu

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_42,

1 Introduction

Both inverse geometry [11] and inverse kinematics[1] [21] can be viewed as the resolution of an optimization problem: non-linear from the configuration space to the special Euclidean group $SE(3)$ for the first one [3], quadratic in the tangent space to the configuration space for the second [5]. This is only one view of the problem, but it helps to formulate efficient solvers and to understand their convergence properties, by using some powerful results of numerical optimization [14]. For controlling a robot, inverse kinematics is nowadays a standard technique, due to its simplicity and the limited computation cost (e.g. 1 ms is enough to invert the kinematics of a 40DOF humanoid robot [5]). Moreover, the structure of the problem is well understood and problems are easy to diagnose.

On the other hand, model predictive control (MPC) is an advanced technique to control a given system by optimizing its predicted evolution [1]. It relies on the systematic evaluation of the control of the system with respect to a reference cost function, while only the first few steps of the optimal trajectory are executed before its complete re-evaluation. The main interest of MPC is the ability of dealing with non-linear systems whose instantaneous linearization is not meaningful.

Like for inverse kinematics, MPC can be formulated as the resolution at each control cycle of a numerical optimization problem depending on the estimated state. However, the typical size of the problem generally makes it difficult to obtain real-time performance [12]. Moreover, this kind of formulation is difficult to interpret. It is typically difficult to quantify the robustness of such controllers [1], or even to explain the reasons that have led to the chosen trajectory.

In this chapter, we consider an optimal-control solver named Differential Dynamic Programming [8]. This numerical scheme provides a simple yet efficient solver of direct implicit (shooting) optimal-control problems, that makes it possible to control complex systems, like humanoid robots [18], despite the inherent complexity of this class of problems. We propose a reformulation that provides numerical advantages and, more importantly, gives a better understanding of the structure of the optimal trajectory. In particular, when only the robot kinematics are considered, we show that every iteration of the algorithm amounts to a sequence of Jacobian pseudo-inversions along the trajectory. Classical pseudoinverse-based inverse kinematics is then equivalent to the optimization of a single-step trajectory. Consequently, once the ratio between the size of the system and the CPU load are sufficiently low, any inverse-kinematics should be considered with several steps ahead rather than with only a single one. The same observation seems valid for inverse dynamics [9].

[1] The problem we name *inverse geometry* is sometimes referred as *inverse kinematics*, the second being referred as *differential* (or closed-loop) *inverse kinematics*. We use 'geometry' when only static postures are implied and keep the word 'kinematics' when a motion is explicitly implied.

2 Model Predictive Control

2.1 Principles and Model

Consider generic dynamical system, with state x and control u:

$$x_{t+1} = f(x_t, u_t, t) \tag{1}$$

f is the evolution function and the time variable t is discrete. x is typically a finite sequence of derivatives of the configuration q, e.g. $x = (q, \dot{q})$. Optimal control computes the control and state trajectories that minimize a given cost function:

$$\min_{X,U} \sum_{t=0}^{T-1} l_t(x_t, u_t) + l_T(x_T)$$

subject to the constraint (1), where T is the preview-interval length (fixed here), $U = (u_0 \dots u_{T-1})$ and $X = (x_0, \dots, x_T)$ are the control and state trajectories and l_t and l_T are the running and terminal cost functions. Linear dynamics and quadratic cost lead to the linear-quadratic regulator, given by Riccati equations.

In practice, the information contained in X and U is somehow redundant. The problem is reformulated as a problem only on X or only U (the other variable being deduced from the dynamic equation). The formulation is said explicit when computing X [13] (designated also by *collocation* [16]) and implicit when computing U [17] (designated also by *shooting* [10]). Both formulations have pros and cons [2]. We consider in the following the implicit formulation, cheaper to solve in practice, without the drawback that it might involve more local minima. For each formulation, the solution to the numerical problem is then approximated using any optimization solver, typically using Newton or quasi-Newton [6] descent.

2.2 Differential Dynamic Programming

Differential Dynamic Programming (DDP) is an iterative algorithm to solve a non-linear optimal control problem using implicit formulation [17]. It is nearly equivalent to the application of a Newton descent algorithm [15]. As in the Newton descent, it approaches a local optimum by iteratively modifying a candidate solution. It starts with initial state and control trajectories (e.g. obtained by integration of the zero control) and then iterates in two stages. It first computes a quadratic model of the variation of current candidate trajectory and computes the corresponding linear-quadratic regulator (LQR—backward loop). The candidate is then modified following the LQR (forward loop).

Quadratic model: We denote v_t the cost-to-go function defined by:

$$v_t(X_t, U_t) = \sum_{k=t}^{T-1} l_k(x_k, u_k) + l_T(x_T)$$

where $X_t = (x_t \dots x_T)$ and $U_t = (u_t \dots u_{T-1})$ are the trajectory tails. To simplify, we drop the t variable and denote the next quantity at $t+1$ by a prime: $v' \equiv v_{t+1}$. DDP relies on the Bellman principle. It proceeds recursively backward in time using the following equation:

$$v^*(X, U) = \min_{x,u,X',U'} \left(l(x, u) + v'^*(X', U') \right)$$

building a quadratic model of v from the quadratic models of l and v'^*:

$$\begin{aligned} v(x + \Delta x, u + \Delta u) = & v(x, u) + v_x \Delta x + v_u \Delta u + \frac{1}{2} \Delta x^T v_{xx} \Delta x + \Delta u^T v_{ux} \Delta x \\ & + \frac{1}{2} \Delta u^T v_{uu} \Delta u + o(||\Delta x||^2 + ||\Delta u||^2) \end{aligned}$$

The quadratic model is defined by the quadratic coefficients v_x, v_u, v_{xx}, v_{ux} and v_{uu}, functions of the derivatives of l, f and v' (see [17] for details).

Backward pass: The optimum Δu can be computed for any Δx. It is obtained at the zero of the derivative of the quadratic model:

$$\Delta u^* = \lambda + \Lambda \Delta x \tag{2}$$

where $\lambda = v_{uu}^{-1} v_x$ and $\Lambda = v_{uu}^{-1} v_{ux}$ are the open-loop and close-loop gains. From the optimal change Δu^*, the quadratic model of v^* can be computed:

$$v_x^* = v_x - \Lambda^T v_{uu} \lambda \tag{3}$$

$$v_{xx}^* = v_{xx} - \Lambda^T v_{uu} \Lambda \tag{4}$$

The backward pass starts from the quadratic model of l_T and then recursively computes the optimal gains of all the control cycles from $T-1$ down to 0.

Forward pass: The forward pass then computes the new candidate trajectory and control schedule. For each control cycle, a new control schedule $\tilde{u}$ is established using (2). For each new $\tilde{u}$, the changes is x are obtained by integrating (1) from x_0 and then propagated through the closed-loop gains of the next time:

$$\Delta x' = x' - f(x, \tilde{u}), \quad \tilde{u} = u + \lambda + \Lambda \Delta x$$

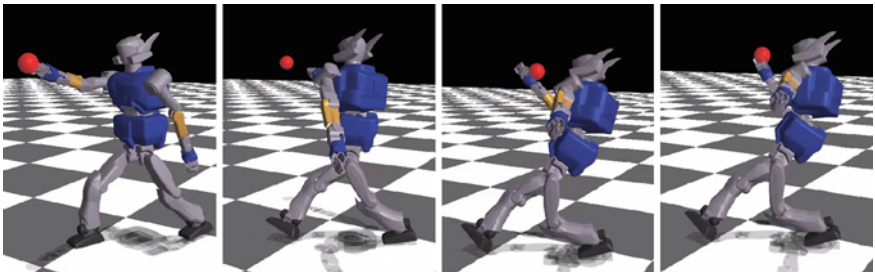


Fig. 1 Snapshots of a whole-body grasping movement on a 25-DOF humanoid robot. The control is computed in real-time. Courtesy from [19]

Performance: The interest of DDP is that its simple formulation can be easily implemented in an efficient way, taking into account the inherent sparsity of a numerical optimal control problem. For example, in [18], a dedicated solver was demonstrated to animate a humanoid virtual avatar in real-time in interaction with a user through a haptic device. It was used to control a simulated 25-DOF HRP2 robot in real-time [19]. In that case, the preview horizon was 0.5 s. The preview control was computed in 50 ms and then interpolated using the underlying LQR at 5 ms, enabling effective real-time control (see Fig. 1).

3 Square-Root Differential Dynamic Programming

In this section we present our proposed modification the the DDP algorithm. The key idea is to propagate the Value Hessian in square-root form. Reminiscent of the square-root Kalman Filter, this formulation ensures positive definiteness and confers numerical stability.

3.1 Algorithm Derivation

The Gauss–Newton approximation: Very often in practice, both the running and terminal costs have sum-of-square form, with the residuals $r(x, u)$:

$$l(x, u) = r(x, u)^T r(x, u)$$

This specific shape is interesting in practice as it leads to a cheap approximation of the second-order derivatives of l in neglecting the second order derivative of r. This is referred as the Gauss–Newton approximation.

$$l_{xx} = r_x^T r_x \ , \quad l_{ux} = r_u^T r_x \ , \quad l_{uu} = r_u^T r_u$$

where r_x and r_u are respectively the derivatives of r by respect x and u. The approximation converges to the real Hessian when the residuals r converge to 0, which in general ensures a good convergence. On the other hand, the approximated Hessian is always positive, which prevents the algorithm from violently diverging, as happens when the true Hessian is non-positive. Moreover, the particular shape of the approximated Hessian can be taken into account when inverting it, since we have:

$$l_{xx}^{-1} l_x^T = \left(r_x^T r_x\right)^{-1} r_x^T = r_x^+$$

where r_x^+ denotes the Moore–Penrose pseudoinverse of r_x and can be efficiently computed without explicitly computing the matrix product $r_x^T r_x$, using for example the SVD or other orthogonal decompositions [7].

In the literature, the Gauss–Newton approximation of the DDP algorithm is referred as the iterative LQR (iLQR) algorithm [20]. In this section, we take advantage of the square shape of the cost and derivatives to propose a more efficient formulation of this algorithm. This shape will also be used to make some correlations with the classical inverse kinematics.

Square-root shape of: v^* In the DDP backward loop, we have to invert the derivatives of v. Being a sum of squares, the cost-to-go v can be expressed as the square of some vector $v^* = s^{*^T} s^*$. However, DDP does not explicitly compute s_x but rather directly propagates the derivatives v_{xx}^* from $v_{xx}'^*$. In the following, we formulate the same propagation while keeping the square shape, by searching the vector $\hat{s}^*$ and matrix $\hat{s}_x^*$ such that

$$v_x^* = \hat{s}_x^{*^T} \hat{s}^*, \qquad v_{xx}^* = \hat{s}_x^{*^T} \hat{s}_x^*$$

At the beginning of the backward pass, the square shape is trivially given by $s(T) = r(T)$ and $s_x(T) = r_x(T)$. During the backward pass, the previous square-root shapes are written s' and s'_x. The derivative v_{xx}^* is given by the recurrence (3), (4). The square shape of (4) is not trivial since it appears as a difference, that we can prove to be positive. We denote by s, s_x and s_u the square root of v, v_{xx} and v_{uu}:

$$s = \begin{bmatrix} r \\ s' \end{bmatrix}, \quad s_x = \begin{bmatrix} s_x \\ s_x^{*'} f_x \end{bmatrix}, \quad s_u = \begin{bmatrix} r_u \\ s_x^{*'} f_u \end{bmatrix}$$

It is easy to show that $v'_{xx} = s_x'^T s'_x$, $v'_{xu} = s_x'^T s'_u$ and $v'_{uu} = s_u'^T s'_u$. In that case, the gains are given by the pseudoinverse of s_u:

$$\lambda = s_u^+ s, \qquad \Lambda = s_u^+ s_x$$

Thanks to the Moore–Penrose conditions, we can reduce $\hat{s}^*$ and $\hat{s}_x^*$ to:

$$\hat{s}^* = s, \qquad \hat{s}_x^* = (I - s_u s_u^+) s_x$$

3.2 Advantages and Discussion

Keeping the square shape of v_{xx} avoids some numerical trouble. In particular, decomposing s_x instead of v_{uu} offers much better numerical behavior. Moreover, it avoids the complexity of a big matrix multiplication. This is formalized below.

Comparison of the costs: To evaluate the complexity of this algorithm, sizes of x, u and r are supposed all equal to n. The cost for one iteration of the loops is $8n^3$, against $11n^3$ for the classical DDP. Moreover, most operations are due to the QR decompositions and could be performed when computing the derivatives, that leads to a total cost of roughly $3n^3$.

Pseudo inverse and projection: The gains and propagation closed forms also provide a better understanding of the nature of the inversion. As in the derivation, we consider only the current time of the backward loop. The Jacobian s_u is the derivative of the cost-to-go. The open-loop gain $s_u^+ s$ only tries to find the current control that minimizes the cost-to-go evolution. In most of cases, s_u has more rows than columns. The pseudoinverse will only provide the control that has the maximum efficiency in the least-square sense. What remains is a part of the cost that can be nullified. This is given by the orthogonal part to the image of s_u, i.e. the kernel of s_u^T, whose projector can be computed by $P_u = I - s_u s_u^+$.

The backward loop then propagates backward the part of the cost that was not accomplished, and that is selected using the projector. The trajectory optimization then corresponds to a sequence of virtual configurations, each of them being moved to optimize its own cost r and to help the configurations ahead in the trajectory by optimizing their residual cost $s^{*'}$.

4 Kinematic Simulation

Three-rotations planar (3R) Robot: Due to a lack of space, we only present some analytical results in simulation with a 3R kinematic model. The dynamic evolution function is reduced to a trivial integration scheme $f(x, u) = x + \Delta t u$, with $x = q$ and $u = \dot{q}$. The robot task is to reach a position p^{ref} with the robot end effector $p(q)$ while minimizing the velocities:

$$r_t = \begin{bmatrix} w_p(p(q) - p^{ref}) \\ w_u u \end{bmatrix}$$

with w_p and w_u the weights of the two cost components. In this case, the derivative r_x is the robot Jacobian J_q while $r_u = w_u I$ is a regularization term. At the first step $T - 1$ of the backward loop, the pseudoinverse is:

$$s_u(T-1)^+ = \begin{bmatrix} w_u I \\ w_p \Delta t J_q \end{bmatrix}^+ = \frac{1}{w_p \Delta t} J_q^{\dagger\eta}$$

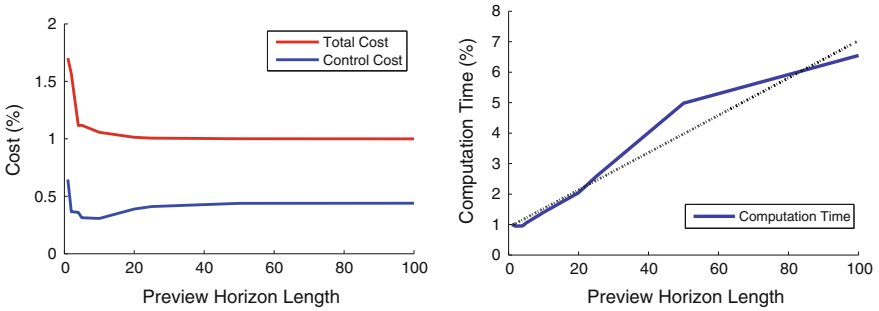


Fig. 2 Performance and computation ratio with respect to the preview length. (*left*) Evolution of the cost when increasing the preview horizon: the total cost is plotted as a ratio with respect to the infinite-horizon optimum. Indicatively, the percentage of the control term of the cost (integral of the velocity norm) is also given. *(right)* Computation load, plotted as a ratio of the load needed to compute the trajectory with a single-step horizon (i.e. cost of an inverse kinematics). The cost increases linearly with the size of the horizon

where $J^\dagger$ denotes the damped inverse [4] with damping $\eta = \frac{w_u}{w_p \Delta t}$. The last term of the trajectory indeed moves following an inverse-kinematics scheme. The same interpretation can be done on the other samples, with a similar regularization and a task that makes a trade-off between going to the target and helping the next sample in the trajectory to accomplish its residual.

Results: The Square Root algorithm on the simulated 3R Robot was implemented in C++. The control sampling frequency is 1 kHz and the cycle of the robot lasted 0.1 s (100 timesteps). We chose $w_u = 0.01$ and $w_p = 1$. The control reaches easily the target with a proper smoothing of the control, as expecting with such a simple system. With this setting, the robot needs 0.1 s to reach the target i.e. 100 control cycles.

We mainly focus the discussion on the comparison with inverse kinematics. As explained above, inverse kinematics is obtained when the horizon T is reduced to 1. On the opposite extreme, the optimum of the infinite-horizon problem is approximately obtained for a preview horizon of 0.1 s (which is the time to the goal). We consider the performance in both the obtained cost and the computation load for T varying from 1 to 100. A summary of the results is given by Fig. 2.

On the left figure, we consider the total cost for the overall executed trajectory. This cost is computed a posteriori, after the execution by the robot. The cost is minimal when T is maximal and vice-versa. Most of the cost increase when T is small is due to the increase of the control term (with an artificial apparent minimum for some $T = 10$ that is due to the ratio over a changing total quantity). On the other hand, the computation load (right figure) increases linearly with the horizon length (as expected). After a certain threshold on the horizon length, the obtained trajectories are the same, with same costs. As always with MPC, the horizon length has to be carefully adjusted: not to small, to find the best trajectory and not to big to limit computation times.

Inverse kinematics is obtained for $T = 1$. The cost is the lowest, but a poor resulting cost. For only a small expense, (e.g. $T = 4$), better trajectories are obtained.

5 Conclusion

In this chapter, we described a square-root formulation of the DDP algorithm. The formulation is numerically more efficient, improving both the computation load and the numerical conditioning. It also makes apparent the relation between MPC and other optimization-based robot algorithms. In particular, it makes use of a sequence of pseudo-inverses of the cost Jacobian along the trajectory. In the particular case where the time evolution function is reduced to the robot kinematics, this sequence is equal to the pseudo-inverse of the cost Jacobian, with the first term of the trajectory following exactly an inverse-kinematics scheme for the final cost.

This study reveals that inverse kinematics is nothing but an MPC scheme with a singular horizon, and that the robot behavior might be very much improved by simply considering a few samples ahead of the current robot position when computing an inverse-kinematics scheme.

The same principle should apply with more complex time-evolution function. For example, when considering the robot dynamics (the state being the configuration and velocity, and the control being joint torques), MPC should meet the operational-space inverse dynamics when the preview horizon collapses.

References

1. Alamir, M.: Stabilization of Nonlinear Systems Using Receding-Horizon Control Schemes. Springer, Lecture Notes in Control and Information Sciences (2006)
2. Biegler, L.: Nonlinear programming: concepts, algorithms, and applications to chemical processes. SIAM, Philadelphia (2010)
3. Das, H., Slotine, J.J., Sheridan, T.: Inverse kinematic algorithms for redundant systems. In: IEEE International Conference on Robotics and Automation (ICRA'88), pp. 43–48. Philadelphia (1988)
4. Deo, A., Walker, I.: Robot subtask performance with singularity robustness using optimal damped least squares. In: IEEE ICRA, pp. 434–441. Nice (1992)
5. Escande, A., Mansard, N., Wieber, P.B.: Hierarchical quadratic programming. Int. J. Robot. Res. (2012) (in press)
6. Goldfarb, D.: A family of variable-metric methods derived by variational means. Mathematics of computation **24**(109), 23–26 (1970)
7. Golub, G., Van Loan, C.: Matrix Computations, 3rd edn. John Hopkins University Press, Baltimore (1996)
8. Jacobson, D.H., Mayne, D.Q.: Differential Dynamic Programming. Elsevier, Amsterdam (1970)
9. Khatib, O.: A unified approach for motion and force control of robot manipulators: the operational space formulation. Int. J. Robot. Res. **3**(1), 43–53 (1987)

10. Leineweber, D.B., Schäfer, A., Bock, H.G., Schlöder, J.P.: An efficient multiple shooting based reduced sqp strategy for large-scale dynamic process optimization: part II: software aspects and applications. Comput. Chem. Eng. **27**(2), 167–174 (2003)
11. McCarthy, J.: Introduction to Theoretical Kinematics. MIT Press, Cambridge (1990)
12. Mombaur, K.: Using optimization to create self-stable human-like running. Robotica **27**(03), 321 (2008). doi:10.1017/S0263574708004724
13. Mordatch, I., Todorov, E., Popović, Z.: Discovery of complex behaviors through contact-invariant optimization. In: ACM SIGGRAPH'12. Los Angeles (2012)
14. Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer, New York (2006)
15. Pantoja, D.O.: Differential dynamic programming and newton's method. Int. J. Control **47**(5), 1539–1553 (1988). doi:10.1080/00207178808906114. http://www.tandfonline.com/doi/abs/10.1080/00207178808906114
16. Schulman, J., Lee, A., Awwal, I., Bradlow, H., Abbeel, P.: Finding locally optimal, collision-free trajectories with sequential convex optimization. In: Robotics: Science and Systems (2013)
17. Tassa, Y., Erez, T., Todorov, E.: Synthesis and stabilization of complex behaviors through online trajectory optimization. In: IROS'12, Portugal
18. Tassa, Y., Erez, T., Todorov, E.: Synthesis and stabilization of complex behaviors through online trajectory optimization. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'12), pp. 4906–4913 (2012). doi:10.1109/IROS.2012.6386025
19. Tassa, Y., Mansard, N., Todorov, E.: Control-Limited Differential Dynamic Programming (Under Review)
20. Todorov, E., Li, W.: A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems. In: Proceedings of the American Control Conference (ACC'05), pp. 300–306. Portland (2005). doi:10.1109/ACC.2005.1469949
21. Whitney, D.: Resolved motion rate control of manipulators and human prostheses. IEEE Trans. Man Mach. Syst. **10**(2), 47–53 (1969)

New Gravity Balancing Technique and Hybrid Actuation for Spatial Serial Manipulators

Basilio Lenzo, Antonio Frisoli, Fabio Salsedo and Massimo Bergamasco

Abstract The problem of gravity balancing of robotic systems has been investigated for a long time. A big amount of different designs has been developed so far, but with several limitations: applicability only to planar kinematics or some particular spatial ones, encumbrance and reduced workspace of the robot, complication of both theoretical and practical implementation. This chapter deepens a new simple technique for gravity balancing any spatial serial manipulator with rotational joints, using a number of springs less or equal to the number of degrees of freedom of the manipulator. Then, such technique is extended to a concept of actuation for robotic systems. Given a robotic manipulator and a force to exert at the level of the end effector, there is no energy consumption regardless of the configuration of the system (like in passive systems), as long as magnitude and orientation of the required force are fixed. Changes in magnitude and/or orientation of the exerted force require some energy to be achieved (like in active systems). Such combined features make an hybrid system with several benefits: low energy consumptions, simplified control and intrinsic safety of the system, with wide prospects in robotics.

Keywords Gravity balancing · Hybrid actuation · Energy efficiency · Springs

B. Lenzo (✉) · A. Frisoli · F. Salsedo · M. Bergamasco
PercRo Lab, TeCIP Institute, Scuola Superiore Sant'Anna, Pisa, Italy
e-mail: b.lenzo@sssup.it

A. Frisoli
e-mail: a.frisoli@sssup.it

F. Salsedo
e-mail: f.salsedo@sssup.it

M. Bergamasco
e-mail: m.bergamasco@sssup.it

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_43,

1 Introduction

High amounts of the required torque in robotic manipulators are used to balance gravity. As an example, industrial robots manipulating large payloads have massive links in order to be able to accomplish their tasks and, consequently, they have relatively big motors. Gravity balancing techniques reduce actuators efforts. A system is said to be gravity balanced if no joint actuator inputs are needed to maintain the system in equilibrium, at any configuration. This happens when the potential energy of the system is invariant, e.g. if the centre of mass of the machine is inertially fixed, or whether some elastic elements compensate suitably the variations of the potential due to changes of configuration (motion of masses).

Several methods of gravity balancing have been proposed over the years, exploiting clever designs using different techniques. In [10, 16] gravity balancing is achieved with counterweights. In [12] and [23] systems using cams and springs are proposed. Focusing on spring balanced mechanisms, examples of architectures are in [2, 5, 19, 20]. An example from daily life is given by the Anglepoise lamp, a desk lamp designed in 1932 by George Carwardine [8]. There have been also some studies (e.g. [4]) addressing the effect of the added weight of the springs.

Planar spring balanced mechanisms are proposed in [2, 8]. Extensions to spatial mechanisms have been object of numerous efforts in literature. For instance, [19] proposes a method applying the one-link solution to n links connected in series with each joint having one dof about the vertical axis, but kinematics is constrained. A hybrid method for gravity balancing spatial serial manipulators is presented in [1], but it is complicated and too bulky to be practically implemented. The technique [14] theoretically guarantees the balance of any general spatial n degrees of freedom (dof) serial-type manipulator using n zero free length springs and some auxiliary links defining n revolute-spherical-spherical-revolute (RSSR) mechanisms.

Not uncommonly, robotic systems have to generate constant forces for relatively long times [6, 11, 17]. The issue of energy managing in robots is crucial and, together with new needs in terms of weight, encumbrance and safety, it is gaining more and more importance in robotics. A concept for a new energy efficient actuator is introduced in [22] based on the principle that, during any force profile generating negative work, energy is stored (and not wasted) and it can be used subsequently whenever a positive work is needed. Energy efficiency issues are also addressed in [3, 5, 18, 24].

In this chapter a new gravity balancing method for spatial serial manipulators is deepened. With respect to [14] we propose a simpler theoretical approach and we address additional considerations on the number of springs needed and on the implementation of auxiliary links. A new concept of actuation is then conceived and spelled out with an example of application on an anthropomorphic arm. The proposed technique features low energy consumption, simplified control and intrinsic safety of the system, opening new scenarios for robotic actuators.

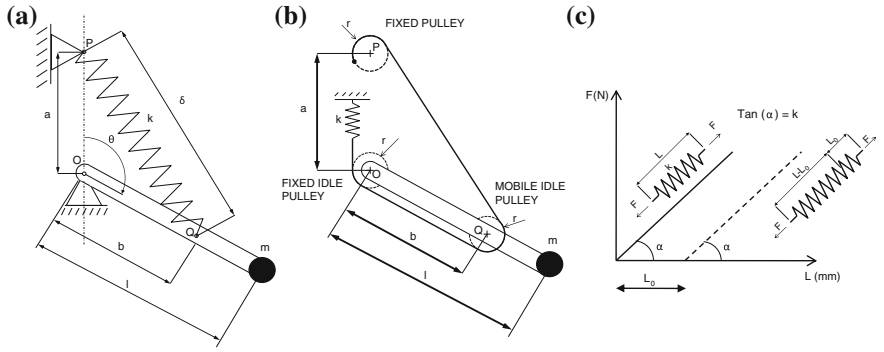


Fig. 1 Single link configuration (**a**), possible real implementation (**b**), spring's free length in ideal case (*continuous line*) and real case (*dashed line*) (**c**)

2 Gravity Balancing: New Technique

Consider a weightless link (Fig. 1a) able to rotate towards point O, having an attached mass m at the other extremity, distant l from O. Its configuration is defined by angle θ. This system can be balanced using a zero free length spring having extremities in point Q (distant b from point O along the link) and in point P (distant a from point O in the vertical direction, opposite to gravity). The total potential energy of the system is $E = mgl\cos\theta + \frac{1}{2}k\delta^2$ due to the contribution of the mass m (g being the gravity acceleration) and of the spring, long $\delta = \sqrt{a^2 + b^2 - 2ab\cos\theta}$. To ensure the equilibrium for any configuration, it must be $\frac{\partial E}{\partial \theta} = 0\ \forall\theta$, thus once fixed m, l, a and b, the stiffness of the zero free length spring must be $k = \frac{mgl}{ab}$ [5].[1]

A zero free length spring exerts a force proportional to its length, rather than its elongation. Such a spring is not available off the shelf (Fig. 1c), but there are several techniques to obtain the same behaviour [5]: for instance, it can be perfectly realized by means of a cable-pulleys system and a standard nonzero free length spring (Fig. 1b). Alternatively, it is possible to use a preloaded spring, namely a spring which elongates when the applied force is greater or equal to its preload. Such preload must be equal to the product of the spring stiffness and its free length.

Consider a generic n dof spatial serial manipulator with rotational joints (Fig. 2) and denote with $\Theta = [\theta_1\theta_2 \ldots \theta_n]^T$ the set of lagrangian coordinates describing the system. Assume the following notation [13]:

- $\mathbf{g}$ the gravity acceleration vector;
- $\mathbf{r}_0$ a vertical unit vector, opposite to $\mathbf{g}$;
- O_{i-1} the point connecting link $i-1$ and link i;
- O_i the point connecting link i and link $i+1$;

[1] Note that the projection of the length of the link l along the direction of the force to balance, involved in the gravitational potential energy, is $l\cos\theta$, and the nonconstant term of the potential energy of the spring is also proportional to $\cos\theta$.

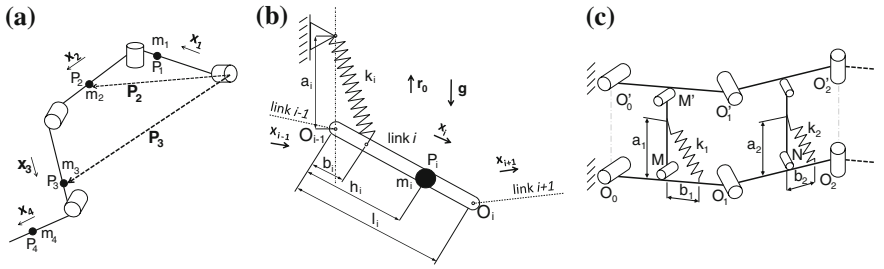


Fig. 2 Part of a generic balanced spatial manipulator (**a**), its generic link i (**b**) and a possible implementation of springs (**c**)

- $\mathbf{x}_i$ a unit vector defining the spatial direction of link i, oriented from point O_{i-1} to point O_i, according to the Denavit–Hartenberg convention [9];
- m_i the mass of link i, concentrated in point P_i distant h_i from O_{i-1};
- l_i the overall length of link i;
- $\mathbf{P}_i$ the vector connecting point O_0 with point P_i.

A spring of stiffness k_i is attached to link i, as to form a triangle, the other two sides being $\mathbf{b}_i = b_i\mathbf{x}_i$ and $\mathbf{a}_i = a_i\mathbf{r}_0$. The vector $\mathbf{a}_i$ must be oriented as $\mathbf{r}_0$ independently of the configuration of the manipulator, differently from $\mathbf{b}_i$ which is configuration dependent ($\mathbf{x}_i$). It is not mandatory that O_{i-1} is a vertex of said triangle. A possible implementation using 4-bar mechanisms is shown in Fig. 2c.[2]

The gravitational potential energy is $E_m = -\sum_{i=1}^{n} \mathbf{P}_i \cdot m_i\mathbf{g} = \sum_{i=1}^{n} \mathbf{P}_i \cdot m_i g\mathbf{r}_0$ and the potential energy of the springs is $E_s = \sum_{i=1}^{n} \frac{1}{2}k_i(||\mathbf{b}_i - \mathbf{a}_i||)^2 = \sum_{i=1}^{n}\left(\frac{1}{2}k_i(a_i^2 + b_i^2) - k_i\mathbf{a}_i \cdot \mathbf{b}_i\right)$. Being $\mathbf{P}_i = \sum_{j=1}^{i-1} l_j\mathbf{x}_j + h_i\mathbf{x}_i$ $(i > 1)$ and $\mathbf{b}_i = b_i\mathbf{x}_i$ and $\mathbf{a}_i = a_i\mathbf{r}_0$, they can be rearranged in the compact forms $E_m = \sum_{i=0}^{n}\sum_{j=0}^{n} \mathbf{x}_i^T k_{m_{ij}}\mathbf{x}_j$ and $E_s = \sum_{i=0}^{n}\sum_{j=0}^{n} \mathbf{x}_i^T k_{s_{ij}}\mathbf{x}_j$ where $\mathbf{x}_0 = \mathbf{r}_0$ and the terms $k_{m_{ij}}$ and $k_{s_{ij}}$ are constant. So, the total potential energy can be written as $E = E_m + E_s = \sum_{i=0}^{n}\sum_{j=0}^{n} \mathbf{x}_i^T \hat{k}_{ij}\mathbf{x}_j$ where $\hat{k}_{ij} = k_{m_{ij}} + k_{s_{ij}}$ is constant as well. So, denoting X as a set of $n+1$ unit vectors $X = [\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_n]^T$, it is $E = X^T\hat{K}X$ where $\hat{K}$ is called *stiffness matrix*. At the level of joint k, the torque τ_k due to conservative force fields is given by

$$\tau_k = \frac{\partial E}{\partial \theta_k} = \sum_{i=0}^{n}\sum_{j=0}^{n} \hat{k}_{ij}\left(\mathbf{x}_i^T \frac{\partial \mathbf{x}_j}{\partial \theta_k} + \frac{\partial \mathbf{x}_i^T}{\partial \theta_k}\mathbf{x}_j\right) \tag{1}$$

and if the system is gravity balanced, it must be $\tau_k = 0\ \forall\ k$. It is easy to see that when $i = j$ the term multiplying $\hat{k}_{ii}$ is zero, $\frac{\partial \mathbf{x}_i}{\partial \theta_k}$ being perpendicular to $\mathbf{x}_i^T$. Thus, it must be $\hat{k}_{ij} = 0$ for $i \neq j$, in other words $\hat{K}$ must be a diagonal matrix, namely

[2] There have been some studies on balancing manipulators without using auxiliary links [7], but workspace may be reduced because of possible elements connected between non-consecutive links.

$\hat{k}_{ij} = m_i g h_i + \sum_{j=i+1}^{n} m_j g l_j - k_i a_i b_i = 0$. So, the generic spring stiffness k_i results

$$k_i = \frac{m_i g h_i + \sum_{j=i+1}^{n} m_j g l_j}{a_i b_i}. \tag{2}$$

The proposed technique is effective even if one or more of the joints of the manipulator is spherical. Indeed, potential energies do not change. A spherical joint can be imagined as a sequence of three rotational joints in which the two theoretical intermediate links have zero length, so there are not springs related to such two links.[3] Formally, the number of springs needed for balancing a general spatial serial manipulator is less or equal to the number of degrees of freedom.

3 From Balancing to Actuating

Consider the n-dof system depicted in Fig. 3a, supposed to have weightless links and a force $\mathbf{F}$ to exert at the end effector.[4] If $\mathbf{F}$ pointed upwards it would have the same effect of a mass $m = |\mathbf{F}|/g$, and the required task could be achieved with the technique described in Sect. 2. A force having non-vertical direction could also be exerted by means of said technique, just by orienting $\mathbf{a}_i$ as $\mathbf{F}$ instead of $\mathbf{r}_0$, as in Fig. 3b. Moreover, a change of magnitude of the load (masses in case of gravity balancing) can be attained, according to previously derived equations, by changing proportionally the magnitude of either $\mathbf{a}_i$ or $\mathbf{b}_i$. So, orientation and intensity of the exerted force can be adjusted acting on the attachment points of the springs. A system able to exert a force of any magnitude and spatial orientation is an *actuation system*.

Given a force to be exerted $\mathbf{F} = F\mathbf{p}$, a potential energy can be defined as $E_f = -\mathbf{P}_n \cdot \mathbf{F}$ (reducing to a gravitational potential energy when $\mathbf{F} = m\mathbf{g}$). Being $\mathbf{b}_i = b_i \mathbf{x}_i$, $\mathbf{a}_i = a_i \mathbf{p}$ and $\mathbf{P}_n = \sum_{i=1}^{n} l_i \mathbf{x}_i$, with the same method described in the previous Section (using $\mathbf{p}$ instead of $\mathbf{r}_0$), the result is that springs have to be designed according to

$$k_i = \frac{F_i l_i}{a_i b_i}. \tag{3}$$

Hence, a given desired force at the level of the end effector of a given spatial serial manipulator can be exerted passively. A suitable system of springs will exert torques that, in traditional actuation systems, are given by motors. The magnitude of the force is changed by modifying the magnitude of $\mathbf{a}_i$ or $\mathbf{b}_i$. The orientation of the force is changed by modifying the orientation of $\mathbf{a}_i$. This kind of actuation is defined as *hybrid actuation*: as long as magnitude and orientation of the desired force are fixed, the force is exerted regardless of the configuration without energy contribution, namely *passively*; if a change of magnitude or orientation of the force

[3] Equally, a two-dof rotational joint with incident axes would work too.

[4] According to classical robotics conventions [21], $\mathbf{F}$ in Fig. 3 is the force applied from the robot to the external world.

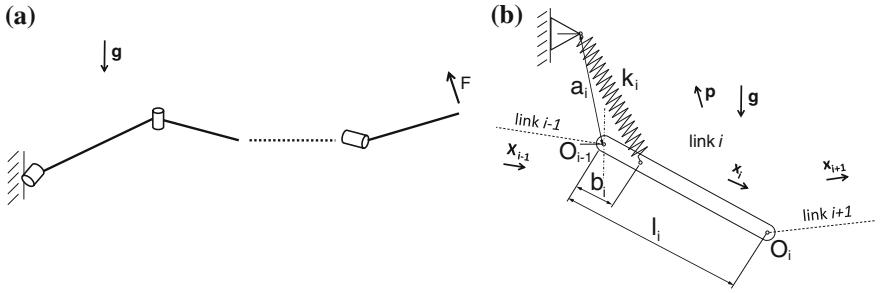


Fig. 3 Serial manipulator exerting F not aligned with gravity (**a**) and its generic link (**b**)

occours, magnitude of $\mathbf{a}_i$ or $\mathbf{b}_i$ or orientation of $\mathbf{a}_i$ need to be changed using some energy, namely *actively*. So, a relatively small power is required only to change the attributes of the force. Therefore, energy consumption can be dramatically reduced with respect to traditional actuation systems, and motors can be smaller since they need only to move some attachment points of springs. Smaller motors means motors less bulky, less heavy and less expensive. For a mobile device there would be more space for batteries, and at the same time batteries would last more due to the reduced energy consumption. Moreover, control is very simplified since it needs to be just a position control rather than a force control. No force sensors are required: the system is intrinsically safe.

4 Application Example: Actuated Anthropomorphic Robotic Arm

As an example of application, consider the anthropomorphic arm kinematics represented in Fig. 4a. Its dofs are: shoulder abduction/adduction, shoulder rotation, shoulder flexion, elbow flexion. The first three dofs have mutually orthogonal axes, incident in the center of the shoulder, ideally represented by a spherical joint. So, only two springs are needed to actuate such manipulator: one for the first three dofs and one for the fourth dof. Suppose that the force to exert $\mathbf{F}$ belongs to the $\mathbf{y}_0$-$\mathbf{z}_0$ plane (Fig. 4a), and it has orientation defined by $\mathbf{y}_0$ with an angle offset of $\pi/3$ towards $\mathbf{x}_0$. So the unit vector associated to $\mathbf{F}$ is, with respect to the $\mathbf{x}_0$-$\mathbf{y}_0$-$\mathbf{z}_0$ frame, $\mathbf{p} = [0\ \cos(\pi/3)\ \sin(\pi/3)]^T$. The following values are chosen: $a_1 = 10$ mm, $b_1 = 8$ mm, $a_2 = 10$ mm, $b_2 = 8$ mm (springs parameters), $l_1 = 30$ mm, $l_2 = 20$ mm (length of links), $|\mathbf{F}| = 50$ N. So, the stiffnesses of the springs k_1 and k_2 are computed according to Eq. (3). Assuming the configuration in Fig. 4a to be $\Theta = [\theta_1\ \theta_2\ \theta_3\ \theta_4] = [0\ 0\ 0\ 0] = \Theta_0$, the following steps are performed:

- starting with $\mathbf{F} = F\mathbf{p}$, $|\mathbf{F}| = 50$ N, the configuration changes in 20 s from Θ_0 to $\Theta = [\pi/6\ 3\pi/4\ -\pi/2\ \pi/2]$;

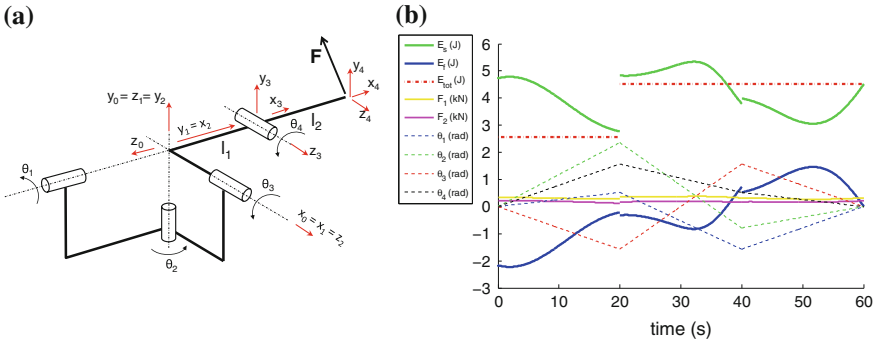


Fig. 4 Anthropomorphic arm kinematics (**a**) and result of the simulation (**b**)

- at time $t = 20$ s the force magnitude changes to 75 N, so $a_1 = 15$ mm and $a_2 = 15$ mm, then the configuration changes in 20 s to $\Theta = [-\pi/2 \ \ -\pi/4 \ \pi/2 \ \pi/6]$;
- at time $t = 40$ s the force orientation changes to $\mathbf{p} = \mathbf{y}_0$ (vertical force), then the configuration changes to Θ_0 in 20 s.

Results of the simulation are in Fig. 4b: energy of springs E_s, energy associated to the force E_f, sum of the two contributions E_{tot}, forces of springs F_1 and F_2, coordinates θ_i. During each 20 s step the total potential energy E_{tot} is constant regardless of the configuration. When the force magnitude increases ($t = 20$ s), a_i increase accordingly and some energy is introduced in the system. When the force orientation changes ($t = 40$ s), the contributions E_s and E_p change but a_i and b_i remain the same, as well as E_{tot}. Indeed, it can be shown easily that $E_{tot} = \sum_{i=1}^{n} \frac{1}{2} k_i (a_i^2 + b_i^2)$.

A prototype of arm exoskeleton exploiting this kinematics and actuation principle was designed at PercRo Lab. Practically, it was implemented by means of a remote actuation system located in the backpack of a user wearing the exoskeleton, like in [13]. Details are not presented here for space reasons.

5 Conclusion

A new simple approach to gravity balance a generic spatial serial manipulator was investigated. The technique is based on an energetic approach and it allows to balance gravity effects for serial spatial manipulators using a number of springs not greater than the number of degrees of freedom of the manipulator. A concept of hybrid actuation is derived extending said technique. During any movement of the manipulator, torques are given by springs and not by motors (as it is in traditional actuation systems). Energy is needed only when a change of force magnitude or force orientation is required. As a result, energy consumption is reduced, control is simplified and the system is intrinsically safe. A detailed example of application

of the concept is described, considering operating conditions including changes of magnitude and orientation of the exerted force.

The design of an anthropomorphic arm exoskeleton based on this concept was completed, and it was partially fabricated. It will be subject of future works.

Acknowledgments This work has been supported by the European project VERE (FP7, FET, 257695)

References

1. Agrawal, S., Fattah, A.: Gravity-balancing of spatial robotic manipulators. Mech. Mach. Theor. **39**(12), 1331–1344 (2004)
2. Agrawal, S.K., Gardner, G., Pledgie, S.: Design and fabrication of an active gravity balanced planar mechanism using auxiliary parallelograms. J. Mech. Des. **123**(4), 525–528 (2001)
3. Allison, J.T.: Plant-limited co-design of an energy-efficient counterbalanced robotic manipulator. Urbana **51**, 61801 (2012)
4. Arakelian, V., Ghazaryan, S.: Improvement of balancing accuracy of robotic systems: application to leg orthosis for rehabilitation devices. Mech. Mach. Theor. **43**(5), 565–575 (2008)
5. Barents, R., Schenk, M., van Dorsser, W., Wisse, B., Herder, J.: Spring-to-spring balancing as energy-free adjustment method in gravity equilibrators. ASME, New York (2009)
6. Colosky Jr, P.E., Ruttley, T.M., et al.: Gravity-Independent Constant Force Resistive Exercise Unit (2004) (US Patent 6685602)
7. Deepak, S.R., Ananthasuresh, G.: Perfect static balance of linkages by addition of springs but not auxiliary bodies. J. Mech. Robot. **4**(2), 021014 (2012)
8. French, M., Widden, M.: The spring-and-lever balancing mechanism, george carwardine and the anglepoise lamp. Proc. Inst. Mech. Eng. C: J. Mech. Eng. Sci. **214**(3), 501–508 (2000)
9. Hartenberg, R.S., Denavit, J.: Kinematic synthesis of linkages. McGraw-Hill, New York (1964)
10. Hayward, V., Gregorio, P., Astley, O., Greenish, S., Doyon, M., Lessard, L., McDougall, J., Sinclair, I., Boelen, S., Chen, X., et al.: Freedom-7: a high fidelity seven axis haptic device with application to surgical training. In: Experimental Robotics V, pp. 443–456 (1998)
11. Howell, L.L., Magleby, S.P., et al.: Substantially constant-force exercise machine (2006) (US Patent 7060012)
12. Koser, K.: A cam mechanism for gravity-balancing. Mech. Res. Commun. **36**(4), 523–530 (2009)
13. Lenzo, B., Frisoli, A., Salsedo, F., Bergamasco, M.: An innovative actuation concept for a new hybrid robotic system. Romansy 19-Robot Design, Dynamics and Control, p. 135. Springer, New York (2013)
14. Lin, P., Shieh, W., Chen, D.: Design of a gravity-balanced general spatial serial-type manipulator. J. Mech. Robot. (Trans. ASME) **2**(3) (2010)
15. Lin, P.Y., Shieh, W.B., Chen, D.Z.: A theoretical study of weight-balanced mechanisms for design of spring assistive mobile arm support (MAS). Mech. Mach. Theor. **61**, 156–167 (2013)
16. Massie, T., Salisbury, J.: The phantom haptic interface: a device for probing virtual objects. In: Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems vol. 55, pp. 295–300 (1994)
17. Nahar, D.R., Sugar, T.: Compliant constant-force mechanism with a variable output for micro/macro applications. In: Proceedings of IEEE International Conference on Robotics and Automation, 2003. ICRA'03. vol. 1, pp. 318–323. IEEE (2003)
18. Plooij, M., Wisse, M.: A novel spring mechanism to reduce energy consumption of robotic arms. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012, pp. 2901–2908. IEEE (2012)

19. Rahman, T., Ramanathan, R., Seliktar, R., Harwin, W.: A simple technique to passively gravity-balance articulated mechanisms. Trans. Am. Soc. Mech. Eng, J. Mech. Des. **117**, 655–657 (1995)
20. Rizk, R., Krut, S., Dombre, E.: Design of a 3D gravity balanced orthosis for upper limb. In: IEEE International Conference on Robotics and Automation, 2008. ICRA 2008. pp. 2447–2452. IEEE (2008)
21. Sciavicco, L.: Robotics: Modelling, Planning and Control. Springer, New York (2009)
22. Stramigioli, S., van Oort, G., Dertien, E.: A concept for a new energy efficient actuator. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2008. AIM 2008. pp. 671–675. IEEE (2008)
23. Ulrich, N., Kumar, V.: Passive mechanical gravity compensation for robot manipulators. In: Proceedings of IEEE International Conference on Robotics and Automation, 1991, pp. 1536–1541. IEEE (1991)
24. Wang, S., van Dijk, W., van der Kooij, H.: Spring uses in exoskeleton actuation design. In: IEEE International Conference on Rehabilitation Robotics (ICORR), 2011, pp. 1–6. IEEE (2011)

Analysis of Constraint Equations and Their Singularities

Rangaprasad Arun Srivatsan and Sandipan Bandyopadhyay

Abstract The identification of singularities is an important aspect of research in parallel manipulators, which has received a great deal of attention in the past few decades. Yet, even in many well-studied manipulators, very few reported results are of complete or analytical nature. This chapter tries to address this issue from a slightly different perspective than the standard method of Jacobian analysis. Using the condition for existence of repeated roots of the univariate equation representing the forward kinematic problem of the manipulator, it shows that it is possible to gain some more analytical insight into such problems. The proposed notions are illustrated by means of applications to a spatial 3-RPS manipulator, leading to the closed-form expressions for the singularity manifold of the 3-RPS in the actuator space.

Keywords Singularity · Parallel manipulator · Univariate equation

1 Introduction

This chapter attempts to revisit the relationship between the constraint equations inherent to a parallel manipulator , and the singularities thereof. It is well-established that singularities in a physical manipulator can be characterised in terms of a corresponding degeneracy in the mathematical equations defining its motion. In particular, in all the cases of singularities in *direct kinematics* (which has also been termed as a "singularity of the second type" [7], "gain-type" singularity [1], "constraint singularity", etc.), it is known that one or more pairs of branches of forward kinemat-

R. A. Srivatsan (✉)
Robotics Institute, Carnegie Mellon University, Pittsburgh 15213, USA
e-mail: arangapr@andrew.cmu.edu

S. Bandyopadhyay
Department of Engineering Design, Indian Institute of Technology Madras, Chennai 600036, India
e-mail: sandipan@iitm.ac.in

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_44,

ics merge—leading to the mathematical condition that the *loop-closure* equations defining the forward kinematics admit repeated roots. There exists a considerable variety in the development of the constraint equations, and the nature of variables included in them. However, notwithstanding the differences in the implementation details, many of these formulations are connected via a fundamental logical thread—namely, the application of *implicit function theorem* to a set of non-linear equations, leading to the analysis of the rank degeneracy of certain resulting Jacobian matrices. While a comprehensive discussion of the Jacobian-based formulations is out of scope of this chapter, the mention of two typical examples may help motivate the point better. In [7], the constraint equation is also the *input-output* equation of the manipulator, i.e., the equations are formed to connect the task-/output-space (i.e., dependent/unknown) variables directly to the input (i.e., independent) variables by eliminating the other unknown variables associated with the passive joints. On the other hand, in other works such as [1, 6], the passive variables are in focus, and they, along with the input variables, define the *configuration space* of the manipulator. Degeneracy of the Jacobian of the constraint equations with respect to the passive variables define the condition for the *gain-type* singularities in the configuration space. It is also understood, that these methods lead to similar results, since a gain of degree-of-freedom in the configuration space typically results in a corresponding gain in the task-space degree-of-freedom.

This chapter follows the same basic approach for the analysis of singularities. However, it differs in the fact that instead of the vanishing of the determinant of a certain Jacobian to identify singularities, it uses the derivative of a special scalar equation in conjunction with the equation itself. This equation is designated as the "forward kinematic univariate", or FKU for brevity. Typically, an FKU is derived in the process of solving the forward kinematic problem itself, by the process of systematic elimination of all the unknowns but one, which is then solved from the FKU itself. This observation motivates a very simple procedure/algorithm for deriving singularity conditions:

1. Derive the FKU depicting the forward kinematics;
2. Set its derivative w.r.t. the lone remaining unknown variable to zero;
3. Solve the above two equations simultaneously.

There are several advantages to this procedure in comparison to the standard Jacobian-based formulations. Firstly, in the case of a number of spatial manipulators of practical importance, the FKU can be derived in closed form—see, e.g., [3, 5]. This opens up the opportunity of deriving the scalar singularity condition also in the closed-form. Secondly, sometimes, it is possible to decompose the FKU into factors. In such cases, the algorithm can be applied separately to the individual factors, which further simplifies the task of the analytical computation. This aspect would be demonstrated in Sect. 3. Finally, often the FKU is either in the form of a polynomial in the unknown variable, or can be converted into one. In such cases, the problem reduces to the computation of the *resultant* of the FKU and its derivative w.r.t. the unknown, which can be accomplished easily provided the degree of the FKU (or its individual factors) is not too high in this variable. There are, however, several

limitations of the proposed formulation as well. Obviously, it cannot be applied to situations, where the FKU is not available in closed form. Also, since the FKU is typically the result of the elimination of a number of variables from the original set of constraint equations, it may accrue one or more *spurious solutions*. The zeros of the discriminant is only guaranteed to be a *super-set* of the singularities in the original system (see, e.g., [4]).

The remaining of the chapter is organised as follows: in Sect. 2, the mathematical formulation for the proposed method is described. In Sect. 3, the same is illustrated with the example of a spatial 3-R$\underline{\text{P}}$S manipulator, and the conclusions are presented in Sect. 4.

2 Formulation

Let, the loop-closure/kinematic constraint equations be defined as:

$$\boldsymbol{\eta}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbf{0}, \tag{1}$$

where $\boldsymbol{\theta}$ represents the set of *active* or known variables, and $\boldsymbol{\phi}$ the set of unknown variables, which could consist of any combination of the *passive* joint/configuration variables, and Cartesian-/task-space variables. At a *regular* point, these equations yield *unique* solutions for $\boldsymbol{\phi}$. At a singularity, however, one or more pairs of solutions merge. Applying the *implicit function theorem* to this situation, the corresponding condition emerges as:

$$\det(\boldsymbol{J}_{\boldsymbol{\eta\phi}}) = 0, \text{ where } \boldsymbol{J}_{\boldsymbol{\eta\phi}} = \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\phi}}. \tag{2}$$

The above observation forms the basis for the identification of the "*gain-type* singularities in the configuration space" [1, 6], as well as the "singularities of the *second type* in the task space" [7], albeit with different meanings for the variable $\boldsymbol{\phi}$. Thus, the said singularities can be identified as the set of points in the workspace, where Eqs. (1, 2) are satisfied simultaneously.[1] In practice, however, it is difficult to solve these equations together—particularly since the singularity condition involves a determinant. This observation motivates the following alternative approach.

Consider, that $\boldsymbol{\theta} \in \mathbb{R}^n$, and $\boldsymbol{\phi} \in \mathbb{R}^m$ (locally at least), where m, n are positive integers, such that n equals the degrees-of-freedom of the mechanism, and $\boldsymbol{\eta} \in \mathbb{R}^m$. The *forward kinematic* problem refers to the finding of solutions for $\boldsymbol{\phi}$ in terms of $\boldsymbol{\theta}$. To achieve this, it is fairly standard to eliminate the variables ϕ_i $(i = 1, \ldots, m-1)$,

[1] If constraint equations are written in, or, converted to, their algebraic (i.e., polynomial) forms, then an equivalent condition for such singularities would be that the *singularity condition* in Eq. (2) in its algebraic form, belongs to the *constraint ideal* generated by the algebraic form of Eq. (1). Thus, the determination of singularities in the constraint equations can also be posed as an *ideal membership* (see, e.g., [4]) problem.

i.e., the components of $\boldsymbol{\phi}$ from Eq. (1), till a single variable (which is ϕ_m in this context) remains in a single equation—which encapsulates into it *all* the kinematic characteristics captured by the original system (1). This final *univariate*, (defined as the FKU in Sect. 1) can be written as:

$$f(\boldsymbol{\theta}, \phi_m) = 0. \tag{3}$$

In the algebraic context, Eq. (3) would be the *resultant* of the algebraic form of Eq. (1). To find the singularities in Eq. (3), the implicit function theorem can be invoked again, leading to the new singularity condition:

$$\frac{\partial f(\boldsymbol{\theta}, \phi_m)}{\partial \phi_m} = 0. \tag{4}$$

Elimination of ϕ_m between Eqs. (3, 4) results in the singularity condition in terms of the actuator variables and the geometric parameters alone.

3 Illustrative Example: The 3-RPS Manipulator

The 3-RPS parallel manipulator was introduced by Lee and Shah in 1988, and has since been studied extensively by several researchers. Some of the important works on the singularity of this manipulator include [2, 8]. The manipulator, as shown in Fig. 1, consists of a fixed and a moving platform. The two platforms are connected by means of three "legs", each of which has a rotary, a prismatic, and a spherical joint. The prismatic joints are actuated, and all the other joints are passive. This gives rise to three-degrees-of-freedom at the moving platform. The coordinates of the (centres of the) spherical joints on the top platform are obtained in the fixed base reference frame as:

$$\begin{aligned}
{}^0\boldsymbol{p}_1 &= (b - l_1 \cos\phi_1, 0, l_1 \sin\phi_1)^T, \\
{}^0\boldsymbol{p}_2 &= \boldsymbol{R}_Z\,(2\pi/3)\,(b - l_2 \cos\phi_2, 0, l_2 \sin\phi_2)^T, \\
{}^0\boldsymbol{p}_3 &= \boldsymbol{R}_Z\,(4\pi/3)\,(b - l_3 \cos\phi_3, 0, l_3 \sin\phi_3)^T,
\end{aligned}$$

where $\boldsymbol{R}_Z(\alpha)$ denotes the rotation matrix for CCW rotation about axis $\boldsymbol{Z}$ through an angle α. Without any loss of generality, the base dimension, b, is scaled to *unity* in the following, which renders all the linear dimensions unit-less. All angles are expressed in radians (Fig. 1).

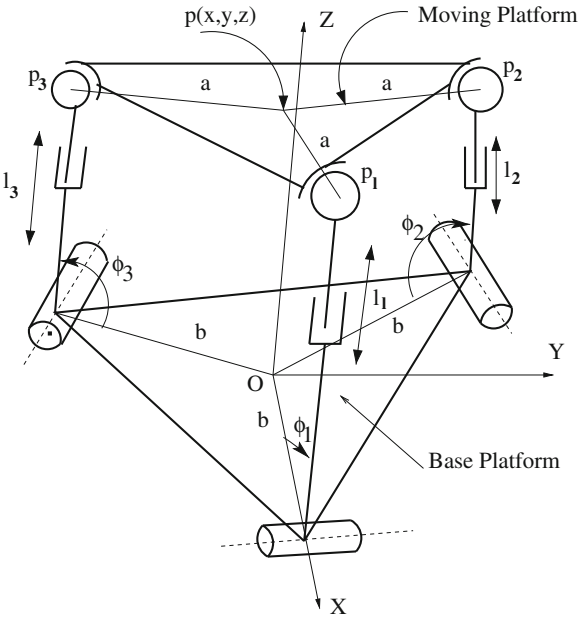


Fig. 1 The 3-RPS manipulator

3.1 Derivation of the FKU

Given the input variables $\boldsymbol{\theta} = (l_1, l_2, l_3)^T$, there are three passive joint variables $\boldsymbol{\phi} = (\phi_1, \phi_2, \phi_3)^T$, which are to be solved from the loop-closure equations denoted by $\boldsymbol{\eta} = \mathbf{0}$, where $\boldsymbol{\eta} = (\eta_1, \eta_2, \eta_3)^T$, and:

$$\eta_1 \triangleq ({}^0\boldsymbol{p}_2 - {}^0\boldsymbol{p}_1) \cdot ({}^0\boldsymbol{p}_2 - {}^0\boldsymbol{p}_1) - 3a^2 = 0,$$
$$\eta_2 \triangleq ({}^0\boldsymbol{p}_3 - {}^0\boldsymbol{p}_2) \cdot ({}^0\boldsymbol{p}_3 - {}^0\boldsymbol{p}_2) - 3a^2 = 0,$$
$$\eta_3 \triangleq ({}^0\boldsymbol{p}_1 - {}^0\boldsymbol{p}_3) \cdot ({}^0\boldsymbol{p}_1 - {}^0\boldsymbol{p}_3) - 3a^2 = 0.$$

The functions η_1, η_2, η_3 are first converted into polynomials in the variables $t_i = \tan(\phi_i/2)$, $(i = 1, 2, 3)$ using the standard *tangent half-angle* substitutions (see, e.g., [5]). After some manipulations, Eq. (1) transforms into a set of three simultaneous quadratic equations of the form $f_1(t_1, t_2) = 0$, $f_2(t_2, t_3) = 0$, $f_3(t_3, t_1) = 0$. The variable t_1 is then eliminated between $f_1 = 0$ and $f_3 = 0$, thereby leading to a new equation of the form $f_4(t_2, t_3) = 0$, which is quartic in t_2, t_3. The second unknown, t_2, is eliminated between $f_4 = 0$ and $f_3 = 0$, yielding the FKU $f(t_3) = 0$, which turns out to be of degree 8 in t_3^2 (see [9] for further details of the elimination scheme). On further analysis, it is found that is it possible to decompose $f(t_3)$ into two quartic factors, i.e., $f(t_3) = g_1(s_3)g_2(s_3)$, where both g_1, g_2 are of degree 4

in $s_3 = t_3^2$. The coefficients of s_3 in g_1, g_2 are functions of the platform dimension a and the inputs l_i only, and these have been obtained in closed-form. The coefficients reveal that g_2 becomes identical to g_1 when a is replaced by $-a$. The actual expressions of the coefficients are too big to be included in this chapter; for the sake of illustration, the coefficient of s_3 in g_1 is given below:

$$(9a^4 + 12a^3(l_3 + 3) - 3a^2(l_1^2 + l_2^2 - l_3^2 - 10l_3 - 15) - 2a(l_3 + 3)$$
$$(l_1^2 + l_2^2 + l_3^2 - 3) - l_1^2(-l_2^2 + 2l_3 + 3) - 2l_2^2 l_3 - 3l_2^2 - l_3^4 - 8l_3^3 - 18l_3^2 - 12l_3)^2.$$

3.2 Analysis of Singularities Using the FKU

Singularities in forward kinematics occur when the FKU, $f(t_3) = 0$, has repeated roots. Taking advantage of the factorisation, and using the formulation presented in Sect. 2, it can be seen easily that singularities can occur in one of two possible manners, and/or their combinations: $g_1(s_3)$ or $g_2(s_3) = 0$ has one or more repeated root(s); $g_1(s_3) = 0$ and $g_2(s_3) = 0$ share one or more common root(s). Both the cases are described below.

- **Case 1** $g_1(s_3) = 0$ **has a repeated root**
  For the numerical values $l_1 = 1, l_2 = 2, l_3 = 3, a = 0.851$, the equation $g_1(s_3) = 0$, written as a *monic polynomial*, becomes:

  $$s_3^4 + 1.83886s_3^3 - 0.23294s_3^2 + 0.00243s_3 + 0.00032 = 0.$$

  The roots of this equation are: $(-1.95839, -0.02950, 0.07452, 0.07452)$, and the passive variables corresponding to the repeated real root are: $\phi_1 = 0.68488$, $\phi_2 = 1.52654$, $\phi_3 = 0.52303$. The corresponding pose of the manipulator is shown in Fig. 2a.
- **Case 2** $g_1(s_3) = 0$ **and** $g_2(s_3) = 0$ **share a common root**
  For the inputs $l_1 = 1.57928, l_2 = 1, l_3 = 2$, and $a = 1/2$, the roots of $g_1(s_3) = 0$ are: $(-0.95589, -0.00028, 0.05276, 0.07163)$, and the roots of $g_2(s_3) = 0$ are: $(0.07163, 0.09818, 0.46876 \pm 0.05965i)$. Thus, the root $s_3 = 0.07163$ is shared between the two factors. The corresponding values of the passive variables are: $\phi_1 = 1.45197$, $\phi_2 = 1.27655$, $\phi_3 = 0.53296$. The manipulator is shown in this pose in Fig. 2b.

3.3 Special Cases

The loop-closure equations (1) suffer another type of degeneracy for "special" combinations of leg inputs, e.g., when two or more of the leg inputs are identical. For instance, consider the case when $l_2 = l_3$: obviously, in this case, $\phi_2 = \phi_3$, and hence, η_2 becomes identical to η_3. Proceeding as before with the equations $\eta_1 = 0$

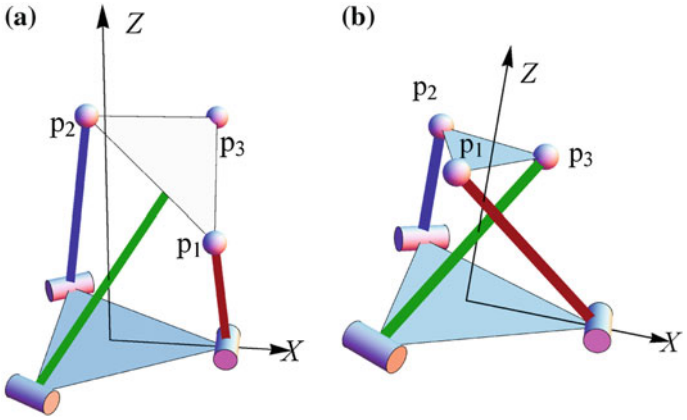


Fig. 2 Singular poses of the 3-RPS. **a** Case 1: $g_1(s_3) = 0$ has a repeated root. **b** Case 2: $g_1(s_3) = 0$ and $g_2(s_3) = 0$ share a common root

and $\eta_2 = 0$, the FKU is obtained in terms of t_1 in this case, which turns out to be a quadratic in t_1^2. Once again, the coefficients of the polynomial are obtained in close-form; e.g., the coefficient of t_1^2 is:

$$2\left(9a^4 - 18a^3 + a^2(9 + l_1^2 - 6l_2^2) + a(-14l_1^2 + 6l_2^2) + 4l_1^2 + l_1^4 - 6l_1^2l_2^2 + l_2^4\right).$$

In the following, the case of gain of one-degree-of-freedom derived in [9] following geometric reasonings, is studied again, albeit in the framework of analysis proposed in this chapter. For $a = 1/2$ and $l_2 = l_3 = 1$, the above-mentioned quadratic equation has a double root when[2] $l_1 = (\sqrt{37} - 3)/4$. The solutions for t_1 are obtained as: $\left(\pm\sqrt{\left(31 - 5\sqrt{37}\right)/6}, \pm\sqrt{\left(31 - 5\sqrt{37}\right)/6}\right)$, i.e., both the positive and negative solutions of t_1 are repeated (as they should, since they correspond to the poses *mirrored* at the base plane). The pose corresponding to the positive solutions is the same as in Fig. 7 of [9].

In the case where $l_1 = l_2 = l_3$, only one of the constraint equations, say, $\eta_1 = 0$, matters. Since $\phi_2 = \phi_1$ in this situation, this equation becomes a quadratic in $\cos\phi_1$. Setting the discriminant of this equation to zero, one obtains the final condition for singularity as $a^2l_1^4 = 0$—which can occur only if the top platform shrinks to a point, or coincides with the base platform.

[2] Note that numerically, $l_1 \simeq 0.770$, as noted in Sect. 5.5.1 of [9]. However, thanks to the proposed algorithm, it is now possible to compute this value *exactly*.

4 Conclusions

A new method for deriving the singularity condition of a parallel manipulator is presented in this chapter. The method depends upon the solution of the forward kinematic problem of the manipulator through a single univariate equation, which is a fairly common practice. The proposed computational scheme involves the elimination of a single variable between two equations, for any manipulator. Special structures in the final univariate equation, e.g., its decomposition into factors, or polynomial nature etc. can also be taken advantage of to reduce the computational complexity, while still obtaining analytical results. This is demonstrated by means of the spatial 3-RPS manipulator, leading to the description of its singularity manifold in terms of closed-form expressions in the general case, perhaps for the first time.

References

1. Bandyopadhyay, S., Ghosal, A.: Analysis of configuration space singularities of closed-loop mechanisms and parallel manipulators. Mech. Mach. Theory **39**(5), 519–544 (2004)
2. Basu, D., Ghosal, A.: Singularity analysis of platform-type multi-loop spatial mechanisms. Mech. Mach. Theory **32**(3), 375–389 (1997). doi:10.1016/S0094-114X(96)00033-X
3. Bonev, I.A.: Direct kinematics of zero-torsion parallel mechanisms. In: Proceedings of the 2008 IEEE International Conference on Robotics and Automation, pp. 3851–3856. Pasadena 19–23 May 2008
4. Cox, D., Little, J., O'Shea, D.: Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra. Springer, New York (1991)
5. Ghosal, A.: Robotics: Fundamental Concepts and Analysis. Oxford University Press, New Delhi (2006)
6. Ghosal, A., Ravani, B.: A differential-geometric analysis of singularities of point trajectories of serial and parallel manipulators. ASME Trans. J. Mech. Des. **123**(1), 80–89 (2001)
7. Gosselin, C., Angeles, J.: Singularity analysis of closed-loop kinematic chains. IEEE Trans. Robot. Autom. **6**, 281–290 (1990). doi:10.1109/70.56660
8. Schadlbauer, J., Walter, D., Husty, M.: The 3-RPS parallel manipulator from an algebraic viewpoint. Mech. Mach. Theory **75**, 161–176 (2014).doi:10.1016/j.mechmachtheory.2013.12.007. http://www.sciencedirect.com/science/article/pii/S0094114X13002504
9. Srivatsan, R.A., Bandyopadhyay, S., Ghosal, A.: Analysis of the degrees-of-freedom of spatial parallel manipulators in regular and singular configurations. Mech. Mach. Theory **69**, 127–141 (2013)

Shape Optimized Heliostats for Kinematic Sun Tracking

Li Meng, Zheng You and Steven Dubowsky

Abstract Sun tracking heliostat mirrors are key components of many solar power systems. They must track the Sun's position in the sky using a spatial 3-D robotic mount. In addition, their ideal shape is a section of a spatial paraboloid that changes as a function of time and the mirrors' positions in the system. Errors in this shape results in reduced system efficiency. The practical implementation of such an ideal heliostat is very difficult and expensive. Here, a novel compliant heliostat design concept is proposed to solve this problem by simply applying moments to the corners of a flat plate whose elastic properties have been tailored to the heliostat's position in the system. Analytical studies of concept are presented that show its performance closely approximates those produced by an ideal paraboloid and that are substantial improvements over current heliostats. Experimental results are presented to validate the approach.

Keywords Kinematics · Heliostats · Compliant mechanisms · Spatial tracking

1 Introduction

Many solar power systems use a central receiver surrounded by a large field of mirrors (heliostats) that are carried by robotic like spatial kinematic mounts. See Fig. 1. These mounts track the sun and concentrate and redirect the sun light to a fixed central receiver tower to generate thermal energy [5, 10, 11].

L. Meng · Z. You
Department of Precision Instrument, Tsinghua University, Beijing, China
e-mail: mengl04@gmail.com

Z. You
e-mail: yz-dpi@mail.tsinghua.edu.cn

S. Dubowsky (✉)
Department of Mechanical Engineering, MIT, Cambridge, MA, USA
e-mail: dubowsky@mit.edu

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_45,

Fig. 1 A central receiver solar power system with its heliostat field [11]

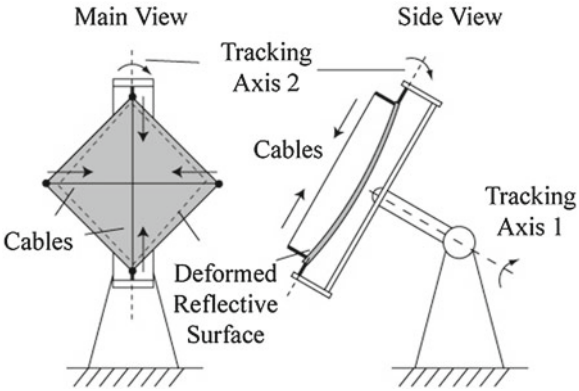


Fig. 2 Cable tensioned adaptive heliostat with its computer controlled spatial tracker [8]

To be most thermally efficient, the receiver should be as small as possible and the heliostat light should fall entirely on the receiver. Hence, the mirror's spot size on the receiver should be as small as possible. This spot size depends on the shape of the heliostat mirror, the tracking errors, and the sun shape effect [1]. Ideally, a heliostat mirror shape should be a precisely curved paraboloid [4]. This shape is difficult and costly to fabricate and control because it is a time varying function of the mirror's position with respect to the receiver tower and the sun's position in the sky that changes with the time of the day and the season. It can be a substantial component of a system's cost [6]. This chapter describes a new design approach to this problem based on using 3-D compliant mechanisms to achieve the required geometrical shape for high optical performance [8]. See Fig. 2.

Most CRS heliostat mirrors are either flat or composed of smaller flat or curved facets mounted on precise support structures [5]. The curved facets are difficult and

expensive to fabricate and need to adjust to different field positions. The performance of flat mirrors is generally poor. Design concepts have been suggested that use reflective compliant members to create adjustable focusing surfaces [6], such as those composed of stretched membranes, implemented with active air pressure to control the mirror's shape. However, these concepts are complicated and costly. Consequently, flat reflective surfaces are still widely used because of their simplicity [9]. In summary, the problem of a concept to practically shape high performance heliostats remains unsolved.

This work's objective is to devise a simple, low-cost concept to achieve an approximation of the ideal adaptive heliostat 3-D mirror shape for high optical performance heliostat mirrors with minimal spot sizes that would be practical and cost effective. In the concept, an ideal heliostat (a segment of offset paraboloid) is approximated by bending the corners of a layered compliant flat square plate in two directions. All the heliostats in a field would have identical physical designs and their shapes would be adjusted by the amount they are bent. The flat-layered mirrors could be easily fabricated, shipped to the site and adjusted to their shape required by their positions in the field. These adjustments can easily be made on site and real-time by automatic control using cable mechanism shown in Fig. 2. It has been shown that for realistic systems, a heliostat optimized for its field position does not need to be adjusted in real-time to achieve nearly ideal performance [7, 8]. This work addresses the mechanic and kinematics of producing a desired shape for an individual heliostat.

The results presented below show that the average efficiency difference between the proposed design and the ideal location-based fixed paraboloids is less than 1 %. The concept is experimentally validated.

2 Heliostat Analytical Models

A heliostat's optical efficiency is the ratio of solar energy that it receives to energy that it reflects on to a given size receiver. The spot size of a heliostat is defined as the circle at receiver aperture plane that receives 98 % of the energy reflected by the heliostat. The smaller a heliostat's spot size, the better is its efficiency.

To apply these concepts to a realistic heliostat a 3-D model is required. Here, the incidence light cones are treated as parallel rays. Geometric optics show that an ideal heliostat surface focusing all of its light to a single receiver focal point is a section of a paraboloid [13]. Referring to Fig. 3, the equation for the paraboloid is given in the cylindrical coordinate system (ρ, φ, Z') and Cartesian coordinates $O'X_0Y_0Z_0$ can be written as [7]:

$$x^2 + y^2 \cos^2 \frac{\theta}{2} + z^2 \sin^2 \frac{\theta}{2} - yz \sin\theta - 4Rz \cos \frac{\theta}{2} = 0 \tag{1}$$

Note f, the mirror focal length, is given by: $f = R(1 + \cos\theta)/2$.

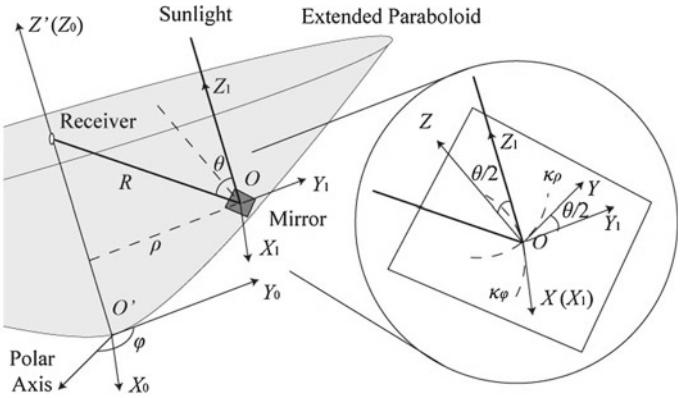


Fig. 3 An ideal heliostat paraboloid [8]

The ideal heliostat shape, given by Eq. (1), is not a simple symmetric surface. It is a function of location with respect to the receiver and the sun angle, which is a function of the time and the date. The two required principal curvatures of the surface can be calculated in (ρ, φ, Z') as:

$$\kappa_\rho = \frac{1}{2f}\left(1 + \frac{\rho^2}{4f^2}\right)^{-\frac{3}{2}} = \frac{1}{2R}\cos\frac{\theta}{2} \tag{2}$$

$$\kappa_\varphi = \frac{1}{2f}\left(1 + \frac{\rho^2}{4f^2}\right)^{-\frac{1}{2}} = \frac{1}{2R}\sec\frac{\theta}{2} \tag{3}$$

R and θ can be approximated as constants over the heliostat surface and the curvatures can also be assumed to be constants.

3 A Tailored Stiffness Heliostat Design Concept

It can be shown that the out-of-plane displacements of a heliostat are much smaller than its planar size. So here two-dimensional bending theory is used to find a mechanism that will compliantly deform a plate so that it approximates an ideal heliostat. In this method, the heliostat is formed from a flat square compliant plate by simple moments and/or forces applied to its four corners. The mirror plate's stiffness *D(x,y)* is divided into two regions, S_1 and S_2, where S_2 is a frame-like structure and S_1 the square center area. See Fig. 4.

S_2 is much stiffer than the square center area S_1. This stiffness distribution permits an approximate solution of the integral equations for required curvatures given by Eqs. (2) and (3). As shown in the figure, the diagonal lines of the plate are aligned

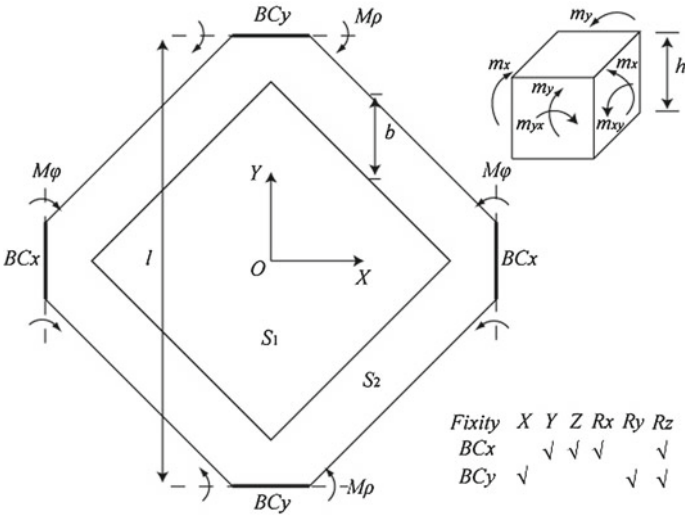


Fig. 4 Heliostat stiffness design [8]

with the body-fixed *OX* and *OY* axis. The corner bending moments (M_ρ, M_φ) create the required deformation. The plate boundary is assumed to be free of forces or moments except for the corners. The *X* corners are assumed to be simply supported with displacement constraints in the *OY* and *OZ* directions. The *Y* corners are simply supported with displacement constrained in the *OX* direction. In which case the plate bending equations can be shown from plate theory to be: For $x, y \in [0, l/2]$,

$$M_\rho = \int_{-(\frac{l}{2}-y)}^{\frac{l}{2}-y} m_y dx = \int_{-(\frac{l}{2}-y)}^{\frac{l}{2}-y} D(\kappa_\rho + \nu\kappa_\varphi)dx \tag{4}$$

$$M_\varphi = \int_{-(\frac{l}{2}-x)}^{\frac{l}{2}-x} m_x dy = \int_{-(\frac{l}{2}-x)}^{\frac{l}{2}-x} D(\kappa_\varphi + \nu\kappa_\rho)dy \tag{5}$$

$$D = \frac{Eh^3}{12(1-\nu^2)} \tag{6}$$

where: h is the thickness of the mirror element;
E is the Young's modulus;
ν is the Poisson's ratio of the material.

The exact solution of Eqs. (4) and (5) for the values of corner moments for a given required plate curvature is complex. However, an approximate solution can be constructed for the frame-like profile, where b is the width of the frame; c is the frame stiffness [7]. So that:

$$\int_{-(\frac{l}{2}-y)}^{\frac{l}{2}-y} D dx = \begin{cases} 2cb & |y| \le \frac{l}{2} - b \\ 2c(\frac{l}{2} - y) & \frac{l}{2} - b < |y| \le \frac{l}{2} \end{cases} \tag{7}$$

$$\int_{-(\frac{l}{2}-x)}^{\frac{l}{2}-x} D dy = \begin{cases} 2cb & |x| \le \frac{l}{2} - b \\ 2c(\frac{l}{2} - x) & \frac{l}{2} - b < |x| \le \frac{l}{2} \end{cases} \tag{8}$$

The stiffness distribution for a thin frame (bl) satisfies Eqs. (4) and (5) for most of the surface when simple moment loads are applied so the plates corners, yield a good approximation to the desired paraboloid shape. Note that $D(x,y)$ is independent of the exact shape, which permits the shape to easily configured to the one required for a mirror's location and the sun angle by only changing the external bending loads. Also paraboloids are not developable surfaces, so there will be some small stretching of the central region of the plate. These stretched membrane forces that can affect the plate's shape [12], as can gravity and wind loads. These effects are neglected here. However, they can be reduced if necessary by proper design choices [7]. This tailored-stiffness approach has the potential to approximate the ideal heliostat surfaces simply and at low cost. During operation, the proposed heliostat mechanism must adjust its orientation as the sun moves across the sky so that the corner loads and the curvatures of the deformed surface are aligned with the ideal paraboloid. This tracking motion can be achieved by target-aligned axis orientation [3] using the computer controlled robotic base. See Fig. 2.

4 Numerical Studies

The numerical analysis of the performance of a heliostat using the tailored stiffness compliant mirror approach is considered here and compared to ideal shaped and flat heliostats. The deformed shape of compliant mirrors are calculated using FEA two-dimensional plate elements. Optical ray tracing is used to calculate the flux distribution on the receiver aperture plane. The sun shape effect is modeled using the algorithm developed by Buie et al. [2]. In this study the assumed heliostat field is located in Los Angeles, California. Each of its heliostats is assumed to be 1 m by 1 m. The receiver tower height is taken as 50 m and the receiver area is 1.4 m^2. The field occupies an area of 200 × 100 m with the layout shown in Fig. 5. In this analysis,

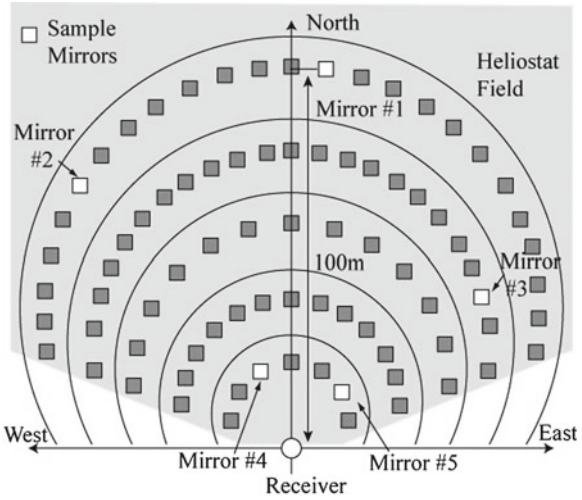


Fig. 5 Reference field configuration [8]

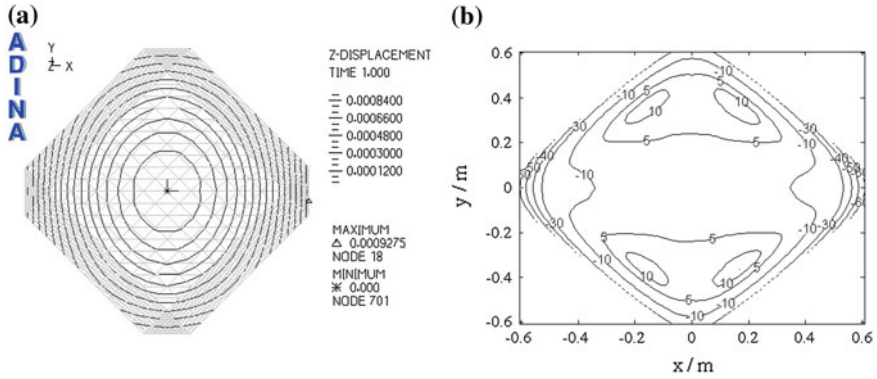


Fig. 6 **a** Contour and **b** errors of surface of a tailored-stiffness mirror [8]

the mirrors do not change their shapes as a function of season and time of the day, but instead use static shapes that are optimized for best optical overall efficiency, based on their location in the field. It has been shown that this approach is nearly as effective as time varying shapes [7, 8].

Figure 6a shows example result analytical surface of a tailored-stiffness mirror at position #1 in Fig. 5 for 15:32 on the spring equinox. The errors of this shape compared to the ideal paraboloid shape shown in Fig. 6b. These errors can be seen to be very small. The RMS value of the displacement errors is 17.74 μm, 1.9 % of the maximum displacement 0.928 mm.

A comparison of a tailored-stiffness design to that for a simple flat mirror, at the same time and same location as for Fig. 6 shows that the spot size of the tailored-

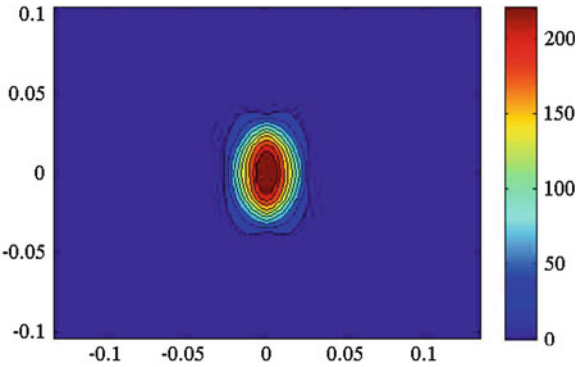


Fig. 7 Experimental optical performance base on ray tracing

stiffness heliostat is 1.87 m^2 while it is 3.55 m^2 for the flat mirror, a reduction of 47 %. For a 1.4 m^2 receiver aperture, the tailored-stiffness heliostat has an optical efficiency of 97.48 % while the flat mirror has 75.08 %, a very substantial improvement [8].

5 Experimental Studies and Key Results

To validate the proposed approach, a two-axis bending experimental heliostat was developed and fabricated. In it moments are manually applied to two opposing corners of the test plate with a calibrated torque wrench and locked into place and deflection of the plate surface is measured. A number of plates were tested, including a polymer honeycomb-sandwich panel with tailored stiffness built with the frame concept described above. For description of this experimental system, see [7].

Figure 7 shows the optical performance of the experimental honeycomb panel studied using numerical ray tracing. It showed that the majority of the light falls in a 20×20 cm region. The spot size is 0.046 m^2. By contrast, ray tracing analysis show that a flat mirror of the same size will have a spot size of 0.41 m^2, an increase of nearly a factor of 10.

6 Summary and Conclusions

This chapter presents an approach to produce high performance, low-cost, heliostat mirrors. It uses a tailored-stiffness compliant surface with simple adjustment mechanisms to form the required parabolic shape mounted on articulated tacking base. These mirrors can be shipped in flat stacks and then assembled and adjusted in the field and mounted on their robotic tracking supports. Results of numerical analy-

sis and experiments show that this tailored-stiffness compliant mechanism approach will significantly improve the efficiency of heliostats at a potentially low cost.

Acknowledgments The support of this work through the MIT Center for Clean Water and Clean Energy and the support of Li Meng during his studies at MIT by Chinese Scholarship Council and Tsinghua University, China, is acknowledged.

References

1. Bonanos, A., Noone, C., Mitsos, A.: Reduction in spot size via off-axis static and dynamic heliostat canting. In: Proceeding of ASME 6th International Conference on Energy Sustainability and 10th Fuel Cell Science Engineering and Technology Conference, San Diego, CA (2012)
2. Buie, D., Monger, A.G., Dey, C.J.: Sunshape distributions for terrestrial solar simulations. Solar Energy **74**(2), 113–122 (2003)
3. Chen, Y.T., Chong, K.K., Bligh, T.P., et al.: Non-imaging focusing heliostat. Solar Energy **71**(3), 155–164 (2001)
4. Igel, E., Hughes, R.: Optical analysis of solar facility heliostats. Solar Energy **22**(3), 283–295 (1979)
5. Kalogirou, S.A.: Solar thermal collectors and applications. Progressin. Energy and Combust. Sci. **30**(3), 231–295 (2004)
6. Kolb, G.J., Jones, S.A., Donnelly, M.W., et al.: Heliostat cost reduction study. In: Sandia National Laboratories: SAND2007-3293, Albuquerque, NM (2007)
7. Meng, L.: Design of compliant heliostats for solar power tower system. Ph. D. Thesis, Tsinghua University (2013)
8. Meng, L., You, Z., Arif, A.F.M., Dubowsky, S.: Shape optimized heliostats using a tailored stiffness approach. J. Solar Energy Eng. **136**(2):021,017 (2014)
9. Schell, S.: Design and evaluation of esolars heliostat fields. Solar Energy **85**(4), 614–619 (2011)
10. Slocum, A.H., Codd, D.S., Buongiorno, J., et al.: Concentrated solar power on demand. Solar Energy **85**(7), 1519C–1529 (2011)
11. Solcar, I., et al.: Final technical progress report: 10 mw solar thermal power plant for southern Spain. The German Space Agency (DLR) (2006)
12. Timoshenko, S., Woinowsky-Krieger, S. (eds.): Theory of Plates and Shells. McGraw-Hill, New York (1959)
13. Winston, R., Minano, J.C. (eds.): Nonimaging Optics. Elsevier Academic Press, Oxford (2005)

Efficient Resolution of Hyper-Redundancy Using Splines

Midhun Sreekumar Menon, B. Gurumoorthy and Ashitava Ghosal

Abstract Hyper-redundant systems such as snake robots, flexible manipulators, ropes and strings discretized as rigid links connected by joints can be reasonably assumed to length preserving during their motion. The resolution of the redundancy in such systems have been addressed by several researchers using least squares and other techniques in which the computation effort increases rapidly with the number of links and thus are not amenable to real time motion planning. In this chapter, we present a computationally efficient, tractrix based algorithm which appear more 'natural' with motion of links 'dying' down along the length of the hyper-redundant system. The hyper-redundant system is represented by splines and it is shown that an approximate length preserving motion of the hyper-redundant system can be obtained by employing the tractrix based algorithm on the control polygon which generate the spline. The deviation from the actual length is related to the configuration of the control polygon and it is shown that this approach reduces the dimension of the problem space leading to a very efficient resolution scheme. The approach also has the added advantages of better visualization of the motion due to the higher order continuities and capability of localized shape control available in splines.

Keywords Hyper-redundant system · Snake robots · Length preserving · Knot · Splines

M. S. Menon (✉) · B. Gurumoorthy · A. Ghosal
Indian Institute of Science, Bangalore, India
e-mail: midhun.sreekumar@gmail.com

B. Gurumoorthy
e-mail: bgm@mecheng.iisc.ernet.in

A. Ghosal
e-mail: asitava@mecheng.iisc.ernet.in

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_46,

1 Introduction

One dimensional flexible objects and algorithms for their natural motion are active areas of research owing to the increased application of snake like robots in various areas such as medical robotics and search and rescue. The main issue in such robots is that they are hyper-redundant and there exists infinite number of solution (or configurations) for a desired motion of the end-effector or the head. Various algorithms have been developed for the resolution of the redundancy and for navigation and motion planning of such robots and kinematic chains. One of the well-known approach uses the pseudo-inverse of the manipulator Jacobian matrix [3]. Obtaining the pseudo inverse explicitly requires $\mathcal{O}(n^3)$ operations where n is the number of joints and is not feasible if n is large. In another well-known approach with complexity $\mathcal{O}(n)$, continuous curves are used to approximate the backbone curve and motion planning is done on the backbone curve [1]. In this approach the length is approximately preserved. Recently an optimization based, length preserving algorithm for motion planning of redundant manipulators has been proposed [2]. Another algorithm by Su et al. [7] uses inverse kinematics and subdivision algorithms complementarily to generate length preserving motion of hyper-redundant manipulators. In both these algorithms, the computation complexity is $\mathcal{O}(n)$ where n is the number of links in the system. Even these efficient algorithms poses limitations for real-time motion visualization and planning of flexible objects such as ropes, strings, hair or snake robots. In such flexible objects, to obtain realistic motion visualization and higher order continuities of slope (C_1) or curvature (C_2), they need to be discretized into a very large number of linear segments/pieces resulting in a large n and an increase in computational costs. In this chapter, we propose a novel method of reducing the computational cost using a tractrix based approach together with splines and the method of sub-division. The tractrix based approach provides the 'natural' motion of the flexible object and the use of splines and sub-division yields a significant reduction in dimension of the joint space with a trade-off that the length of the hyper-redundant system is now approximately preserved.

2 Review of Tractrix Based Approach and Splines

In this section, we present a short review of the tractrix based approach and of splines for the sake of completeness. More details about splines are available in any geometric modeling textbook such as Piegl and Tiller [4] and more details on tractrix based approach are available in Refs. [5, 6].

2.1 Tractrix Based Resolution

Consider a single link of length L moving in the plane. If the head P moves parallel to the X axis and the motion of the tail is *along* the link at each instant, then the tail describes the well-known classical curve known as the tractrix. The equation of the tractrix is the solution of the differential equation

$$\frac{dy}{dx} = \frac{-y}{\sqrt{L^2 - y^2}} \tag{1}$$

given as

$$x = L \log \frac{y}{L - \sqrt{L^2 - y^2}} - \sqrt{L^2 - y^2} \tag{2}$$

Some of the main properties of the tractrix are as follows:

- The infinitesimal $dr = \sqrt{dx^2 + dy^2}$ is the local minimum of all possible infinitesimal displacements of the tail and for a motion of the head dp, $dr \leq dp$.
- The tractrix motion of the link can be extended to 3D space in terms of two differential equations of the form

$$\frac{dy}{dx} = \frac{y - y_e}{x - x_e}, \quad \frac{dz}{dx} = \frac{z - z_e}{x - x_e} \tag{3}$$

where the equations of the path followed by head are $y_e = m_1 x_e$, $z_e = m_2 x_e$ with $m_1 = y_p/x_p$, $m_2 = z_p/x_p$, and (x_p, y_p, z_p) is the destination point of the head.

Instead of numerically integrating the two differential equations, a computationally intensive process, the following algorithm can be used obtain the location of the tail for a given initial positions of head, tail and the destination point of the head $\mathbf{X}_p = (x_p, y_p, z_p)^T$.

Algorithm TRACTRIX3D

1 Define the vector $\mathbf{S} = \mathbf{X}_p - \mathbf{X}_h$ where $\mathbf{X}_h$ is the current location of the head.
2 Define the vector $\mathbf{T} = \mathbf{X} - \mathbf{X}_h$ where $\mathbf{X} = (x, y, z)^T$ is the tail of the link lying on the tractrix.
3 Define the new reference coordinate system $\{r\}$ with the X-axis along $\mathbf{S}$. Hence $\hat{\mathbf{X}}_r = \frac{\mathbf{S}}{|\mathbf{S}|}$.
4 Define the Z-axis as $\hat{\mathbf{Z}}_r = \frac{\mathbf{S} \times \mathbf{T}}{|\mathbf{S} \times \mathbf{T}|}$.
5 Define rotation matrix ${}^0_r[\ R\ ] = [\hat{\mathbf{X}}_r \quad \hat{\mathbf{Z}}_r \times \hat{\mathbf{X}}_r \quad \hat{\mathbf{Z}}_r]$.
6 The Y-coordinate of the tail (lying on the tractrix) is given by $y = \hat{\mathbf{Y}}_r \cdot \mathbf{T}$ and the parameter p can be obtained as $p = L\ \text{sech}^{-1}(\frac{y}{L}) \pm |\mathbf{S}|$.

7 Using the parametric form of the tractrix with p denoting the parameter, obtain the X and Y coordinate of the point on the tractrix in the reference coordinate system as

$$x_r = \pm|\mathbf{S}| - L \tanh\left(\frac{p}{L}\right) \quad y_r = L \operatorname{sech}\left(\frac{p}{L}\right) \tag{4}$$

8 Once x_r and y_r are known, the point on the tractrix $(x, y, z)^T$ in the global fixed coordinate system {0} is given by

$$(x, y, z)^T = \mathbf{X}_h + {}^0_r[\ R\ ](x_r, y_r, 0)^T \tag{5}$$

The algorithm *TRACTRIX3D* can be used for resolution of redundancy for any serial hyper-redundant system. Consider a hyper-redundant manipulator with n rigid links $l_1, l_2, ..., l_n$ with joints $j_1, j_2, ..., j_{n-1}$ where j_i is the joint connecting link i and link $i+1$. Consider the last two links l_n and l_{n-1}. The head of the link l_n, denoted by the point j_n, is required to be moved to a new position $j_{n_{\text{new}}}$ given by $(x_p, y_p, z_p)^T$. From the steps in *TRACTRIX3D*, we can obtain the new location of the tail point j_{n-1} denoted by $(x, y, z)^T$ as it follows a tractrix (see Eq. (5)). The link l_{n-1} is attached to the link l_n and hence the tail of the link l_n can be considered to be the head of the link l_{n-1}. The head of the link l_{n-1} should now be moved from its existing location to $(x, y, z)^T$. The location of the tail of link l_{n-1}, following a tractrix, can again be obtained from the steps given in algorithm *TRACTRIX3D*. Following similar steps, we recursively obtain the motion of the head and tail of all links down to the first link l_1.

We can make the following remarks about the above resolution scheme.

- The algorithm for resolution of redundancy has a complexity of $\mathcal{O}(n)$ where n is the number of rigid links. This fact makes the algorithm amenable for real time computations.
- Under a tractrix motion, when the head of the link l_n moves by dr_n the displacements of all the links obey the inequality $dr_0 \leq dr_1 \leq \ ... \ \leq dr_{n-1} \leq dr_n$, with the equality $dr_i = dr_{i-1}$ reached *only* when the line of motion of joint j_i coincides with link l_i. A consequence of this observation is that the motion of the links progressively gets smaller and appears to 'die' out as we move towards the first link. This feature gives a 'natural' looking motion of the hyper-redundant system.

2.2 Representation Using Splines

Splines are extensively used to represent smooth curves as they possess advantageous properties of local control, and numerical stability. In our approach, as a first step, a control polygon is generated for the smooth spline curve to represent the hyper-redundant system at its initial configuration. It maybe mentioned that the number of legs in the control polygon can be much smaller than the number of links in the hyper-redundant system. Consider a curve as shown in Fig. 1a and the dotted lines

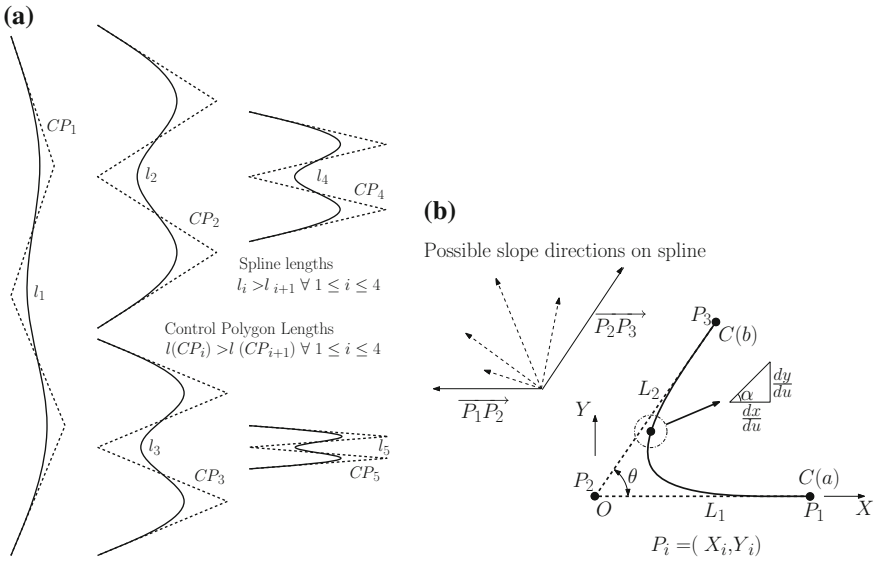


Fig. 1 Spline *curve* and its control polygon. **a** Spline length and length preserving motion of control polygon. **b** Quadratic spline length dependence on angle

which show the control polygon for the curve. As the control polygon is moved using the tractrix algorithm, the curve changes. As shown in Fig. 1a, the length of the curve will change even though the length of the control polygon remains constant due to the tractrix based motion. One can intuitively see that as the angle between the edges of the control polygon *decreases* the length of the curve deviates more from the length of the control polygon. This intuitive notion can be mathematically expressed using the equation

$$dl_1^2 = dl_0^2 \left(1 + 2N_3' L_2 du^2 \sin(\alpha - \theta) d\theta\right) \tag{6}$$

where u is the parameter for the spline curve, dl_0 is the initial curve length when the included angle is θ, dl_1 is the curve length when the included angle changes by $d\theta$, α is the slope at an arbitrary point on the spline curve, $N_3{}'$ is the derivative of the spline interpolation function for the point P_3, L_2 is the length of the segment P_2P_3. It is well known that the derivative of a B-Spline is again a B-Spline curve. Clearly, as the weighing coefficients are all positive, the slope vector of B-Spline at any point is always a linear combination of $p + 1$ adjacent Q_i's which are vectors directed along the control polygon legs with some scaling. In the case of a quadratic curve, slope at any point on the spline will be a convex combination of vectors along two adjacent control polygon segments P_1P_2 and P_2P_3. As shown in Fig. 1b, the slope angle is always in the range $\theta \leq \alpha \leq 180$. Hence, the second term in Eq. (6) is always positive for increasing angles and vice-versa. Thus, as the angle between segments increase, the length of the spline increases and decreases the other way

round. Furthermore, when the control polygon is fully stretched out as a straight line, $\theta = 180^\circ$ and $Y_i = 0 \forall\, i$. This can be used to conclude that the length of the curve is always less than or equal to the length of the control polygon.

As shown above, the length of spline changes as the included angles of the control polygon changes. This makes it unusable for redundancy resolution where the length must remain constant. In the next section, we present an algorithm to sub-divide the control polygon when the angle θ becomes too small and the length of the curve changes more than a user defined amount from the original length.

3 Approximate Length Preservation in Splines

From Sect. 2, the spline length and the control polygon length are related by

$$l(CP) = l(C(u)) + E(C(u)) \tag{7}$$

where $E(C(u)) \geq 0$ is length difference at any given stage of the motion. In subdivision algorithm, the control polygon is subdivided by inserting control points. Figure 2 illustrates the idea of subdivision.

In the subdivision, P_1P_1' and $P_1'P_2$ is replaced by P_1P_2 in $\Delta P_1P_1'P2$. By triangle inequality, the length of control polygon is reduced. On the other hand, the length of the spline remains the same as before. If original spline and control polygon are denoted by $C_0(u)$, CP_0 respectively and if the spline and control polygon after subdivision is denoted by $C_1(u)$, CP_1 respectively, then we can write from Eq. (7).

$$l(CP_1) \leq l(CP_0),\ l(C_1(u)) = l(C_0(u)) \Rightarrow E(C_0(u)) \geq E(C_1(u)) \tag{8}$$

After subdivision, the length difference between spline and control polygon decreases. Most importantly, the angles in the introduced edge increases and by Eq. (6), the sensitivity to change in spline length decreases, thereby effectively reducing error in the spline length. Hence, through subdivision, it is possible to control spline length error by setting a threshold angle value. If the angle between any two segments is less than this threshold ($\theta_i \leq \theta_{th}$), then the control point is subdivided into two new control points. One effect of subdivision is that the number of control points monotonically increases over time depending on the warping of the control polygon and this increases the computation requirement. To overcome this problem, we can also reduce the number of legs in the control polygon when parts of the control polygon stretches out and the included angle crosses a pre-defined threshold. When this is done, the number of legs in the control polygon will reduce and the dimension of the variable space will reduce. The insertion and deletion of links is schematically shown in Fig. 3.

In the next section, we present numerical simulation results which illustrate the theory presented in the sections above.

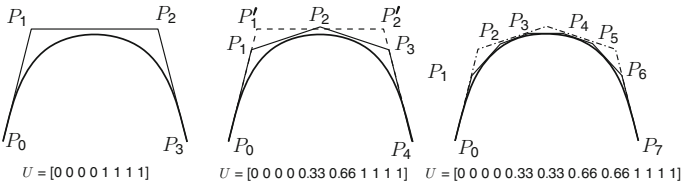


Fig. 2 Subdivision in splines

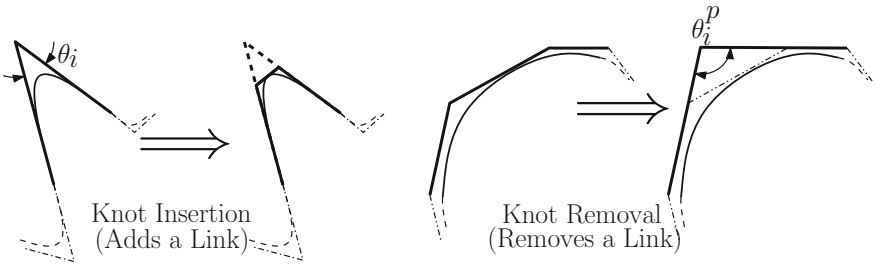


Fig. 3 Knot insertion and knot removal

4 Numerical Simulation

The tractrix algorithm and spline length preservation algorithm are both simulated on a four link planar kinematic chain with path as shown in Fig. 4a. The snapshots on various points on the path are shown in Fig. 4b. These reveal that one obtains a more natural motion using splines, in addition to savings in computation.

Figure 4d shows the variation of control polygon length and Fig. 4c shows number of control points over the simulation duration. As seen, the algorithm adapts to the characteristics of the perturbation path by adding control points as and when required to compensate for the warping of the chain. It also removes control points from the chain as and when the curve can be simplified and represented in terms of a lesser number of control points. In this simulation, threshold for inserting a control point/knot was when the included angle between two segments goes below 140°. Similarly, the threshold for removing a control point/knot was when the two consecutive included angles went above 160°.

Figure 5 shows the length variation of the splines for the algorithms. The variation in length of the spline has been brought down from 1.0 to 0.45, a reduction of 50 %, demonstrating that the algorithm works. Moreover, it is seen that the knot removal part of the algorithm is not affecting the length change much (a change of the order of ≈0.02), when compared with the gain by subdivision.

Finally, the algorithm is used to simulate a generic curve in 3D space moved along an arbitrary direction. The length of the curve is 30 units and is discretized into 20 links and perturbed in 2,650 steps of 0.1, thus making a total motion of 265 units. Figure 6 shows the various snapshots during the motion.

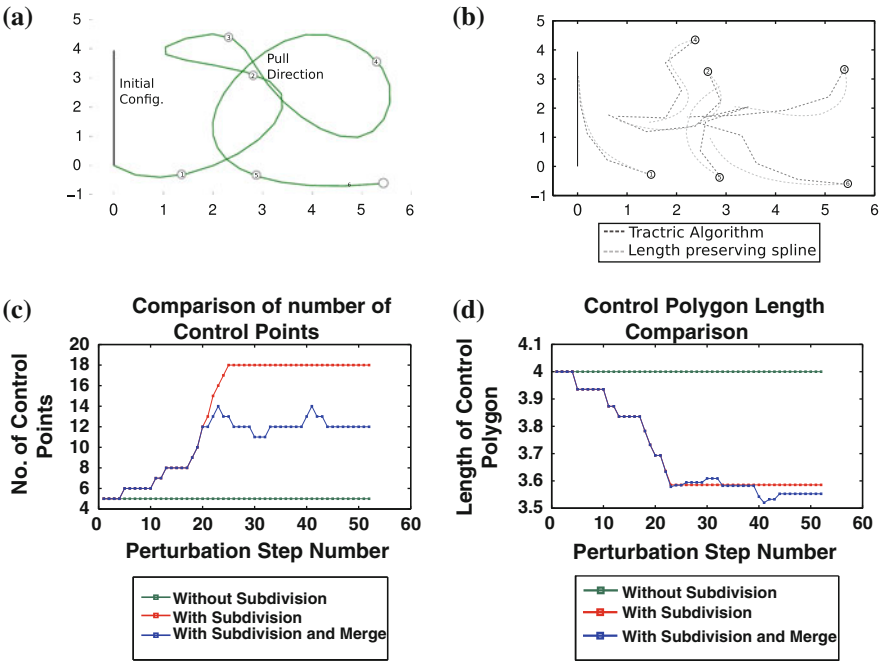


Fig. 4 Simulation of planar redundant system. **a** Kinematic simulation in 2D with approximately length preserved spline. **b** Tractrix based simulation. **c** Variation in number of control points. **d** Variation in control polygon length

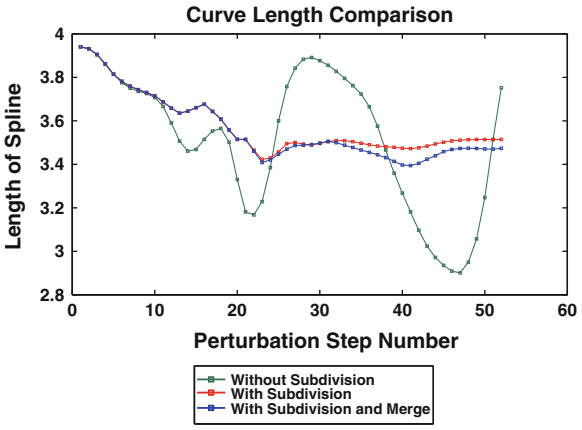


Fig. 5 Length variation of spline during the simulation

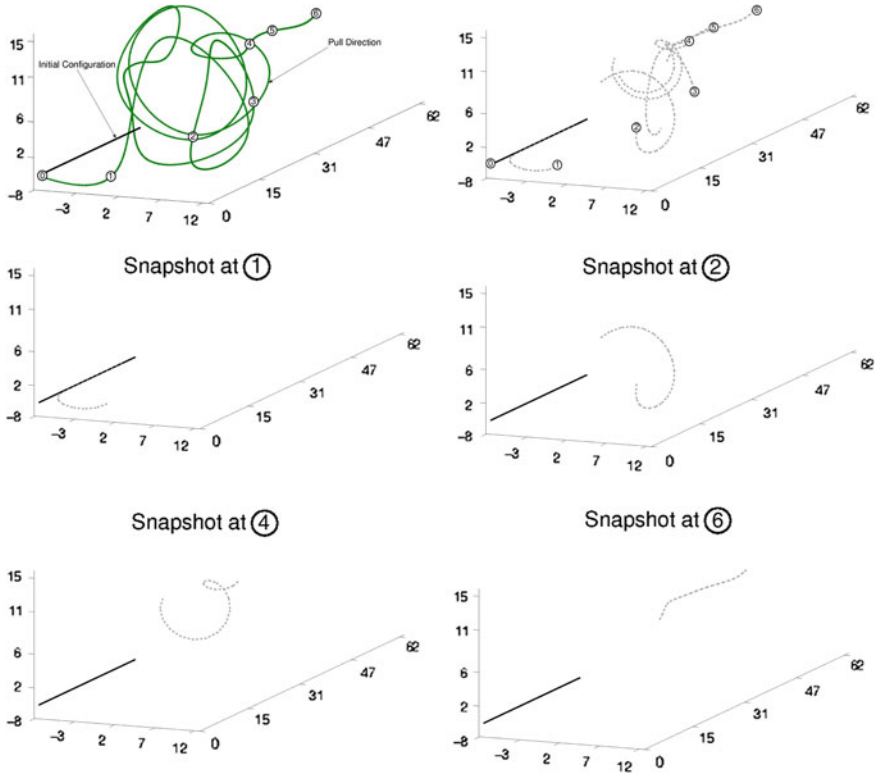


Fig. 6 Motion of an arbitrary *curve* in a generic direction in 3D

5 Conclusion

This chapter proposes a new paradigm for the simulation and visualization of the motion of one-dimensional flexible objects using a tractrix based approach and splines. The tractrix based approach yields a natural motion of the hyper-redundant system and the use of splines leads to efficient computation and more realistic visualization of the motion. An important feature of the proposed algorithm is that it is a purely *kinematics* and *geometry* based approach. The approach can be applied to simulation and realistic visualization of the motion of generic flexible objects such as snakes, chains, ropes and for redundancy resolution in hyper-redundant robotic manipulators.

Acknowledgments This work was funded in part by the Robert Bosch Centre for Cyber Physical Systems (RBCCPS) at the Indian Institute of Science, Bangalore.

References

1. Chirikjian, G.S., Burdick, J.W.: A modal approach to hyper-redundant manipulator kinematics. IEEE Trans. Robot. Autom. **10**(3), 343–354 (1994)
2. Menon, M.S., Ananthasuresh, G., Ghosal, A.: Natural motion of one-dimensional flexible objects using minimization approaches. Mech. Mach. Theory **67**, 64–76 (2013)
3. Nakamura, Y.: Advanced Robotics: Redundancy and Optimization. Addison-Wesley Longman Publishing Co., Inc., Boston (1990)
4. Piegl, L., Tiller, W.: The NURBS Book. Springer, Berlin (1997)
5. Reznik, D., Lumelsky, V.: Sensor-based motion planning in three dimensions for a highly redundant snake robot. Adv. Robot. **9**(3), 255–280 (1994)
6. Sreenivasan, S., Goel, P., Ghosal, A.: A real-time algorithm for simulation of flexible objects and hyper-redundant manipulators. Mech. Mach. Theory **45**(3), 454–466 (2010)
7. Su, Z., Li, L., Zhou, X.: Arc-length preserving curve deformation based on subdivision. J. Comput. Appl. Math. **195**(1–2), 172–181 (2006). doi:10.1016/j.cam.2005.03.092

Kinematic Modeling of an EAP Actuated Continuum Robot for Active Micro-endoscopy

Mohamed Taha Chikhaoui, Kanty Rabenorosoa and Nicolas Andreff

Abstract An active micro-endoscope based on concentric tubes, an emerging class of continuum robots, is presented hereby. It is designed to reach the digestive tube and the stomach for early cancer detection and intervention. The manipulator is constructed from three flexible, telescopic, and actuated tubes. The actuators are based on Electro-Active Polymer electrodes coated and patterned around the tube. A full multi-section kinematic model is developed; it is used to compare the existing constant curvature configuration to the proposed micro-endoscope. That comparison is established according to the reachable workspace and the performance indices. The results are used to prove the effectiveness of the embedded actuation method to reach the workspace more dexterously, which is very useful in medical systems, especially in surgical applications.

Keywords Continuum robot · Active cannula · Electro-active polymers (EAP) · Kinematic modeling

1 Introduction

Continuum robots are still enthralling researchers' interests, almost half a century after the first early prototype: the "Tensor Arm" of [1]. Exceptional usefulness of continuum robots appears in applications where it is restrictive to have joints and stiff links. They have the potential to suffer localized damage while still maintaining

M. T. Chikhaoui (✉) · K. Rabenorosoa · N. Andreff
Automatic Control and Micro-Mechatronics Department, FEMTO-ST Institute, Besançon, France
e-mail: mohamed.chikhaoui@femto-st.fr

K. Rabenorosoa
e-mail: kanty.rabenorosoa@femto-st.fr

N. Andreff
e-mail: nicolas.andreff@femto-st.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_47,

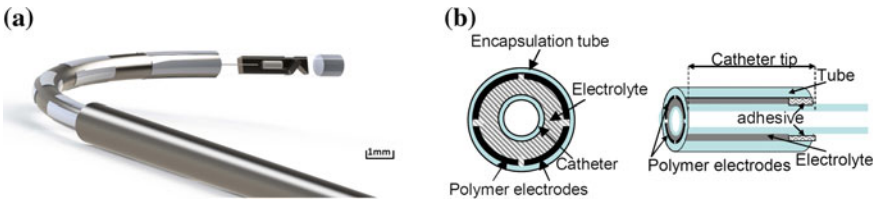


Fig. 1 Design of the considered continuum robot. **a** CAD design of the tubes, **b** Illustration of an EAP-based embedded actuation [8]

a healthy degree of functionality [4]. Moreover, continuum robots are able to navigate through complex anatomy. They may be considered as part of robots for MIS (Minimally Invasive Surgery) and NOTES (Natural Orifice Transluminal Endoscopic Surgery). Recent medical continuum robots include laparoscopic application, laser manipulators, catheters and micro-endoscopes, summarized in [9]. Concentric tubes developed by [7, 10] for endonasal skull base surgery are a major contribution to continuum robots category. Such manipulators are constructed from three precurved telescopic concentric tubes that have small diameters (less than 3 mm). They are actuated at their base by translation and axial rotation of each tube, and the overlapping builds the final shape.

Concentric tubes are the starting point of our study. We aim to change tubes curvatures by the means of embedded actuators, and thereby, provide additional degrees of freedom to the system. Monitoring the curvature and bending each tube in different directions are indeed expected to enhance the manipulator performances. Our considered manipulator is presented in Fig. 1a, showing a CAD design of a curved tube that holds a laser tool at its end-effector. The bending is performed with Electro-Active Polymer (EAP) electrodes coated around the tube based on [8] work and sketched in Fig. 1b.

We present in the next section the kinematic modeling of an active cannula robot, starting from a standard approach to achieve modeling of an EAP actuated continuum manipulator. In Sect. 3, both models will be used to analyze a part of the reachable workspace. Then we can establish a theoretical comparison between the existing configurations and our design, using also performance indices to confirm the manipulability improvement. Finally, in Sect. 4, we will conclude and present several future challenges that still need to be achieved in this field.

2 Kinematic Modeling

Before we can start modeling differential kinematics, a few assumptions need to be set up in the standard approach. This concerns the description used in previous work [10]. Furthermore, the modeling of a continuum robot with EAP-based embedded actuators will be described.

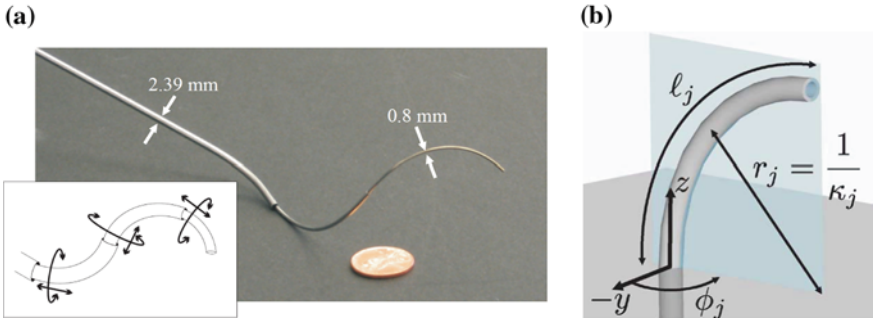


Fig. 2 Active cannula description: **a** three concentric tube configuration [9], and **b** arc parameter description [11]

2.1 Standard Approach

2.1.1 Arc Parameters

The standard approach [10] is based on a configuration of n concentric tubes presented in Fig. 2a. Each tube $i \in \{1 \dots n\}$ is made of a straight part (S_i) and a precurved distal one (C_i) and can translate (by ρ_i) and rotate (by θ_i) with respect to the z-axis of the robot base. Thus, depending on the translation of each tube, the concentric assembly can be decomposed into m successive links, defined by the concentric overlaps of straight parts, precurved parts or nothing (e.g. for three tubes, $C_3/S_2/S_1$ or $C_3/C_2/\emptyset$). Each link is modeled by an arc of a circle (constant curvature assumption), described by three parameters: its length ℓ_j, its curvature κ_j which is the inverse of the radius of curvature r_j, and the angle ϕ_j of the so-called equilibrium plane containing the arc (Fig. 2b).

Depending on the overlapping of the n tubes, the curvature of link $j \in \{1, \dots, m\}$ is given by:

$$\kappa_j = \sqrt{\kappa_{x_j}^2 + \kappa_{y_j}^2} \quad \text{with} \quad \kappa_{x_j} = \frac{\sum_{i=1}^{n} E_i I_i \kappa_{i,j} \cos\theta_{i,j}}{\sum_{i=1}^{n} E_i I_i}, \kappa_{y_j} = \frac{\sum_{i=1}^{n} E_i I_i \kappa_{i,j} \sin\theta_{i,j}}{\sum_{i=1}^{n} E_i I_i} \tag{1}$$

where E_i is the elastic modulus, I_i is the cross sectional moment of inertia, $\kappa_{i,j}$ is the intrinsic curvature of the ith tube in the jth link and $\theta_{i,j}$ denotes the ith tube angle about the jth link frame *z-axis*. Finally, the equilibrium plane angle is given by:

$$\phi_j = \arctan(\kappa_{y_j}/\kappa_{x_j}) \tag{2}$$

2.1.2 Specific Mapping and Independent Mapping

In [11], three spaces are defined:

- the actuator space: $\{\rho_i, \theta_i | i \in \{1 \dots n\}\}$
- the configuration space: $\{\kappa_j, \phi_j, \ell_j | j \in \{1 \dots m\}\}$
- the task space: $SE(3)$.

Two space transformations are thus defined:

1. The *specific mapping* from the actuator space to the configuration space (actuator dynamics). This mapping totally depends on the actuation of the tubes.
2. The *independent mapping* from the configuration space to the task space (forward kinematics). This mapping is the same for all concentric tube architectures, satisfying the assumption of constant curvature links and can be generically modeled.

Forward kinematics can be accomplished in a variety of ways: through Denavit-Hartenberg parameters [5], Frenet-Serret frames [5], integral formulation [3], and exponential coordinates [7, 11]. Using the latter convention, the transformation T_j from link $j-1$ to link j decomposes into a rotation of center $\mathbf{r_j} = [1/\kappa_j, 0, 0]^T$ about the *y axis* by α_j and a rotation about the *z axis* by ϕ_j:

$$T_j = \begin{bmatrix} R_z(\phi_j) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_y(\alpha_j) & \mathbf{p_j} \\ 0 & 1 \end{bmatrix} \tag{3}$$

where $\alpha_j = \kappa_j \ell_j$ and $\mathbf{p_j} = [r_j(1 - \cos\alpha_j), 0, r_j \sin\alpha_j]^T$.

2.1.3 Differential Kinematics

To compute constant curvature kinematics of a multi-section tube, one must compute single section tube kinematics. For the brevity of this chapter, the computation is omitted but details can be found in [9]. The velocity of link j with respect to link $j-1$ is given by:

$$\mathbf{V_j} = \underbrace{\begin{bmatrix} \cos\Delta\phi_j(\cos(\kappa_j\ell_j) - 1)/\kappa_j^2 & 0 & 0 \\ \sin\Delta\phi_j(\cos(\kappa_j\ell_j) - 1)/\kappa_j^2 & 0 & 0 \\ -(\sin(\kappa_j\ell_j) - \kappa_j\ell_j)/\kappa_j^2 & 0 & 1 \\ -\ell_j \sin\Delta\phi_j & 0 & -\kappa_j \sin\Delta\phi_j \\ \ell_j \cos\Delta\phi_j & 0 & \kappa_j \cos\Delta\phi_j \\ 0 & 1 & 0 \end{bmatrix}}_{J_j} \begin{bmatrix} \dot{\kappa}_j \\ \Delta\dot{\phi}_j \\ \dot{\ell}_j \end{bmatrix} \quad \text{where } \Delta\phi_j = \phi_j - \phi_{j-1} \tag{4}$$

Using the adjoint transformation introduced by [6], the full independent kinematic Jacobian can be deduced from the individual ones:

$$J_{indep} = \begin{bmatrix} J_0 & Ad_{T_0} J_1 & Ad_{T_{01}} J_2 \dots Ad_{T_{0(m-1)}} J_m \end{bmatrix} \tag{5}$$

where $T_{0j} = T_0 T_1 \ \ldots T_j$ is the jth transformation matrix at the jth link and

$$Ad_T = \begin{bmatrix} R & R[t]_\times \\ 0 & R \end{bmatrix} \tag{6}$$

with R and $\mathbf{t}$ the rotation and translation component of T and $[t]_\times$ the skew-symmetric matrix associated to the vector cross-product by $\mathbf{t}$. Thus, determining the number of links in a configuration is a preliminary task to understand full Jacobian matrix dimension. For a configuration with three totally curved concentric tubes, we obtain a 6 by 9 matrix, as there are 3 links.

The specific kinematic Jacobian maps actuator derivatives $\begin{bmatrix} \dot{\rho}_i & \dot{\theta}_i \end{bmatrix}^T$ into arc parameters derivatives $\begin{bmatrix} \dot{\kappa}_j & \Delta\dot{\phi}_j & \dot{\ell}_j \end{bmatrix}^T$. In the three totally curved concentric tube configuration, the full specific kinematic Jacobian J_{spec} is a 3 by 2 matrix. Consequently, this configuration is a non-holonomic robot: the whole space can be reached but, in a given state, only a subset of velocity directions is achievable.

2.2 Modeling of EAP Actuated Concentric Tubes

Adding an embedded actuation to the previous configuration is beneficial as it provides a direct control of the intrinsic curvatures $\kappa_{i,j}$ of each tube, whereas the standard approach only takes into account constant intrinsic precurvatures. This adds one component per tube in the actuator space $(\rho_i, \theta_i, \kappa_i)$ but does not change the two other spaces. Therefore, the independent mapping and the independent kinematic Jacobian are the same as those described above, and only the specific mapping and specific kinematic Jacobian need to be derived.

The curvature of a tube made of Electro-Active Polymer (Fig. 1) follows a linear law in terms of voltage according to the relation explained in [8]:

$$\kappa_{i,j} = 1/r_{i,j} = C_{PPY} V_i, \tag{7}$$

where C_{PPY} is considered as the Polypyrrole actuation constant and V_i is the applied voltage.

By appropriate low-level control, we foresee to be able to servo $\kappa_{i,j}$, the intrinsic precurvature of tube i in link j, to a desired value. Thereby, the specific mapping will have the same expression as in the standard approach, with only a change in the inputs (precurvatures). To compute the dependent Jacobian, for three totally curved concentric tubes, we first need to determine the number of links. With such configuration, there are three links: three tubes for the first link, two tubes for the second link, and one tube for the third link as shown in Fig. 3a.

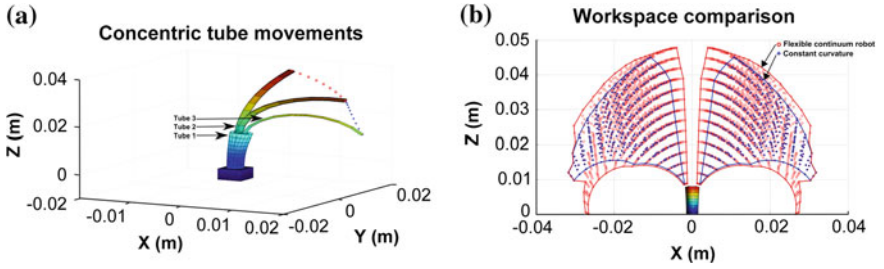


Fig. 3 End-effector workspace is marked by *blue* '+' for constant curvature active cannulas and by *red* 'o' for flexible continuum robot. **a** Schematic description of tube translation and second tube curvature $\kappa_{2,j}$ change from 50 to 10 m^{-1}. **b** Workspaces superimposition in the x-z plane

Differentiating (1) with respect to $\dot{\kappa}_{i,j}$ and $\dot{\theta}_{i,j}$ yields:

$$\dot{\kappa}_j = \frac{1}{\sqrt{\kappa_{x_j}^2 + \kappa_{y_j}^2}} \begin{bmatrix} \kappa_{x_j} & \kappa_{y_j} \end{bmatrix} \begin{bmatrix} \mathbf{A_j} & \mathbf{B_j} \\ \mathbf{C_j} & \mathbf{D_j} \end{bmatrix} \begin{bmatrix} \dot{\kappa}_{in,j} \\ \dot{\theta}_{in,j} \end{bmatrix} \tag{8}$$

where $\dot{\kappa}_{\mathbf{in,j}} = [\dot{\kappa}_{1,j}\ \dot{\kappa}_{2,j}\dot{\kappa}_{3,j}]^T$, $\dot{\theta}_{\mathbf{in,j}} = [\dot{\theta}_{1,j}\ \dot{\theta}_{2,j}\dot{\theta}_{3,j}]^T$, $\dot{\kappa}_i = C_{PPY}.\dot{V}_i$ and

$$\mathbf{A_j} = \begin{bmatrix} \frac{E_1 I_1 \cos\theta_{1,j}}{\Sigma_i E_1 I_1} & \frac{E_2 I_2 \cos\theta_{2,j}}{\Sigma_i E_2 I_2} & \frac{E_3 I_3 \cos\theta_{3,j}}{\Sigma_i E_3 I_3} \end{bmatrix} \tag{9}$$

$$\mathbf{B_j} = \begin{bmatrix} -\frac{E_1 I_1 \kappa_{1,j} \sin\theta_{1,j}}{\Sigma_i E_1 I_1} & -\frac{E_2 I_2 \kappa_{2,j} \sin\theta_{2,j}}{\Sigma_i E_2 I_2} & -\frac{E_3 I_3 \kappa_{3,j} \sin\theta_{3,j}}{\Sigma_i E_2 I_2} \end{bmatrix} \tag{10}$$

$$\mathbf{C_j} = \begin{bmatrix} \frac{E_1 I_1 \sin\theta_{1,j}}{\Sigma_i E_1 I_1} & \frac{E_2 I_2 \sin\theta_{2,j}}{\Sigma_i E_2 I_2} & \frac{E_3 I_3 \sin\theta_{3,j}}{\Sigma_i E_3 I_3} \end{bmatrix} \tag{11}$$

$$\mathbf{D_j} = \begin{bmatrix} \frac{E_1 I_1 \kappa_{1,j} \cos\theta_{1,j}}{\Sigma_i E_1 I_1} & \frac{E_2 I_2 \kappa_{2,j} \cos\theta_{2,j}}{\Sigma_i E_2 I_2} & \frac{E_3 I_3 \kappa_{3,j} \cos\theta_{2,j}}{\Sigma_i E_2 I_2} \end{bmatrix} \tag{12}$$

Similarly, differentiating (2) yields:

$$\dot{\phi}_j = \frac{1}{\sqrt{\kappa_{x_j}^2 + \kappa_{y_j}^2}} \begin{bmatrix} -\kappa_{y_j} & \kappa_{x_j} \end{bmatrix} \begin{bmatrix} \mathbf{A_j} & \mathbf{B_j} \\ \mathbf{C_j} & \mathbf{D_j} \end{bmatrix} \begin{bmatrix} \dot{\kappa}_{in,j} \\ \dot{\theta}_{in,j} \end{bmatrix} \tag{13}$$

For the jth link, the corresponding arc length derivative corresponds to $\dot{\ell}_j = \dot{\rho}_j$. This leads to the specific kinematic Jacobian at the jth link:

$$\begin{bmatrix} \dot{\kappa}_j \\ \dot{\phi}_j \\ \dot{\ell}_j \end{bmatrix} = \frac{1}{\sqrt{\kappa_{x_j}^2 + \kappa_{y_j}^2}} \begin{bmatrix} \kappa_{x_j}\mathbf{A_j} + \kappa_{y_j}\mathbf{C_j} & \kappa_{x_j}\mathbf{B_j} + \kappa_{y_j}\mathbf{D_j} & 0 \\ -\kappa_{y_j}\mathbf{A_j} + \kappa_{x_j}\mathbf{C_j} & -\kappa_{y_j}\mathbf{B_j} + \kappa_{x_j}\mathbf{D_j} & 0 \\ 0 & 0 & \sqrt{\kappa_{x_j}^2 + \kappa_{y_j}^2} \end{bmatrix} \begin{bmatrix} \dot{\kappa}_{in,j} \\ \dot{\theta}_{in,j} \\ \dot{\rho}_j \end{bmatrix} \tag{14}$$

The above specific kinematic Jacobian is a square matrix and does not contain any structural non-holonomic constraint. Moreover, it is a 3 by 3 matrix. Consequently,

only two tubes should be enough to reach any pose in the workspace. However, keeping three tubes provides us with a redundancy which is highly recommended, especially in medical and surgical manipulators.

3 Performance Analysis

3.1 Workspace Analysis

Workspace means the reachable zones of a manipulator end-effector. In traditional robotics, one can obtain the workspace by inverting the direct geometrical model. However, in continuum robotics, converting the effect of curvature change or bending angle into movements is significantly more complex. We restrict our analysis to planar movements for an intermediate step. This is sufficient to illustrate the key advantages of changing an additional arc parameter: the curvature.

We take into account a simple configuration case: three tubes with the same insertion angle α_i that are not rotated about their *z axis*. The total robot length is 45 mm and the tube outer diameters are respectively 3.05, 1.45 and 0.72 mm. The only possibility for a precurved concentric tube configuration to reach additional zones in the *x*-*z* plan is to combine tube translations. This would be controlled by the pre-curvatures already defined and thus, provides a reduced freedom to the end-effector. Moreover, path-controlling this movement would be noticeably challenging. However, monitoring all the curvatures yields more options to the manipulator to sweep even more space. Figure 3a shows that changing the second tube intrinsic curvature $\kappa_{2,2}$ in the second link, provides an additional reachable zone, without even changing first and third tube translation or insertion angle. The workspace was generated with a translation sampling of 3 mm, and a second tube curvature change sampling of $20\,\mathrm{m}^{-1}$. The workspace previously reachable by the existing active cannula type robot is hold, and additional set of movements is achieved each time one of the curvatures is changed. It is proven by both workspaces superimposition shown in Fig. 3b.

3.2 Performance Indices

This analysis is based on the full Jacobian J_{robot} singular values σ_i. Thus, we need a singular-value-decomposition (SVD) of the matrix. This study is based on the three most significant performance indices: manipulability, isotropy, and condition number. Mathematical exact definitions can be found in [2] and their expressions are :

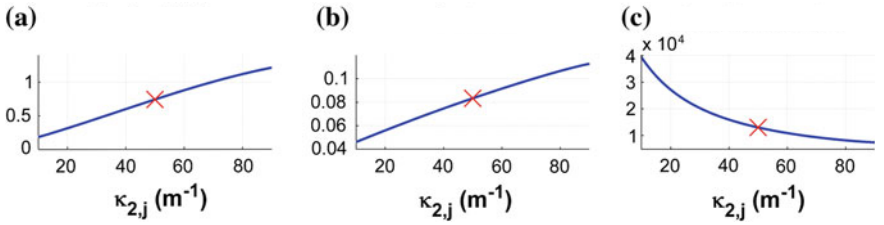


Fig. 4 Performance index variation according to second tube precurvature variation below and beyond $\kappa_{2,j} = 50\,\mathrm{m}^{-1}$ fixed for the configuration in the standard approach in Sect. 2: **a** Manipulability, **b** Isotropy, and **c** Condition number

$$\text{Manip} = \sqrt{\prod_i \rho_i},\ \text{Isotropy} = \frac{\prod_i \rho_i}{\sum_i \rho_i},\ \text{Condition number} = \frac{\rho_{max}}{\rho_{min}} \tag{15}$$

As shown in Fig. 4, monitoring the tube pre-curvature has a significant effect. Firstly, we notice that manipulability is enhanced when the tube bends beyond $\kappa_{2,j} = 50\,\mathrm{m}^{-1}$ (Fig. 4a). Otherwise, it decreases as the tube is straightened to a linear configuration. Nevertheless, the manipulator is able to reach additional zones, which was impossible with constant curvature tubes. The same phenomena is observed for the isotropy index. Straightening the tube draws the manipulator closer from a singular configuration as the isotropy reaches zero. In both cases, we notice that the isotropy measure is very low. It is due to the robot architecture that undeniably does not allow velocities in all directions similarly. Observing the third curve, the singular position is confirmed when $\kappa_{2,j}$ is near zero, with a condition number close to infinity (Fig. 4c). Beyond $\kappa_{2,j} = 50\,\mathrm{m}^{-1}$, the Jacobian matrix is well-conditioned. The straight position of the manipulator is a singular configuration; thus, it is more challenging to achieve velocities. The manipulability indices are increasing with the curvatures. It is owing to the easiness to generate radial movements in a bent position.

4 Conclusion

In this chapter, a novel EAP-based actuation technique of an active micro-endoscope was briefly described. Demonstrating the benefits of monitoring the tubes curvatures in contrast with a constant curvature existing configuration was the main contribution of this chapter. On the one hand, this has been proven in terms of dependent Jacobian analysis which changes from non-holonomic to holonomic. On the other hand, performances of both manipulators have been compared through a part of the workspace as well as via the three most significant performance index evaluation. It has proven that the variable curvature improves the continuum robot performances.

For the future works, additional mechanical constraints as shearing and torsion have to be included. Moreover, other tube materials that allow more flexibility need

to be explored. Another challenge is to improve the actuator design: modifying the electrodes patterned along the tubes would enable more bending directions and would provide more degrees of freedom to the manipulator.

Acknowledgments This work has been supported by the Labex ACTION project (contract ANR-11-LABX-01-01) and by μRALP, the EC FP7 ICT Collaborative Project no. 288663.

References

1. Anderson, V.C., Horn, R.C.: Tensor arm manipulator design. ASME Trans. **1**, 1–12 (1967)
2. Angeles, J., Park, F.: Performance evaluation and design criteria. In: Siciliano, B., Khatib, O. (eds.) Springer Handbook of Robotics, pp. 229–244. Springer, Berlin(2008)
3. Chirikjian, G.S., Burdick, J.W.: A modal approach to hyper-redundant manipulator kinematics. IEEE Trans. Robot. Autom. **10**(13), 343–353 (1994)
4. Gravagne, I.A., Walker, I.D.: Kinematic transformations for remotely-actuated planar continuum robots. In: IEEE International Coneference on Robotics and Automation (2000)
5. Hannan, M.W., Walker, I.D.: Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots. J. Robot. Syst. **20**(12), 45–63 (2003)
6. Murray, R.M., Li, Z., Sastry, S.S.: A Mathematical Introduction to Robotic Manipulation. CRC Press, Boca Raton (1994). DOI 9780849379819
7. Sears, P., Dupont, P.E.: A steerable needle technology using curved concentric tubes. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2850–2856 (2006)
8. Shoa, T., Munce, N.R., Yang, V., Madden, J.D.: Conducting polymer actuator driven catheter: overview and applications. In: Proceedings of SPIE 7287, Electroactive Polymer Actuators and Devices (EAPAD), vol. 7287, pp. 1–9 (2009)
9. Webster, R.J.I., Jones, B.A.: Design and kinematic modeling of constant curvature continuum robots: a review. Int. J. Robot. Res. **29**, 1661–1683 (2010)
10. Webster, R.J.I., Okamura, A.M., Cowan, N.J.: Toward active cannulas: miniature snake-like surgical robots. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2857–2863 (2006)
11. Webster, R.J.I., Romano, J.M., Cowan, N.J.: Mechanics of precurved-tube continuum robots. IEEE Trans. Robot. **25**(11), 67–78 (2009)

Kinematics Analysis and Singularity Loci of a 4-UPU Parallel Manipulator

Massimiliano Solazzi, Massimiliano Gabardi, Antonio Frisoli and Massimo Bergamasco

Abstract The chapter presents the kinematics and singularity analysis by screw theory of a novel 4-UPU fully parallel manipulator with four degrees of freedom. The kinematics is characterized by four UPU legs with the actuation on the prismatic joint and allows the three translations of the end-effector and one rotation along a direction orthogonal to the platform. In the chapter we show how by studying the singularity loci and analyzing both the constraint and actuation Jacobian by screw theory, it is possible to analytically determine the geometric conditions and kinematic parameters that lead to a null determinant of the Jacobian. Moreover, the singular configurations for the legs and the effects on the kinematics are investigated.

Keywords Parallel kinematics · Singularity analysis · Screw theory

1 Introduction

Parallel manipulators have been widely studied and utilized for many practical applications, where their characteristics of high stiffness, good dynamic performance and precise positioning are relevant. Among the class of lower-mobility parallel manipulators, the three degrees of freedom (3-DoF) fully-translational or fully-rotational parallel manipulators are the most common, while the number of 4-DoF devices is relatively small. The possible kinematic structures for lower-mobility parallel devices has been studied in [4], and in particular structure synthesis of 4-DoF parallel manipulators has been investigated in [1] and [5], while the analysis of specific 4-DoF fully parallel kinematics has been presented in [6, 7, 9]. The analysis of kinematics and singularity loci in parallel manipulators presents greater challenges, and the screw theory is a useful tool to perform the analysis in a more effective

M. Solazzi (✉) · M. Gabardi · A. Frisoli · M. Bergamasco
PERCRO Laboratory, TeCIP Institute, Scuola Superiore Sant'Anna, Pisa, Italy
e-mail: solazzi@sssup.it

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_48,

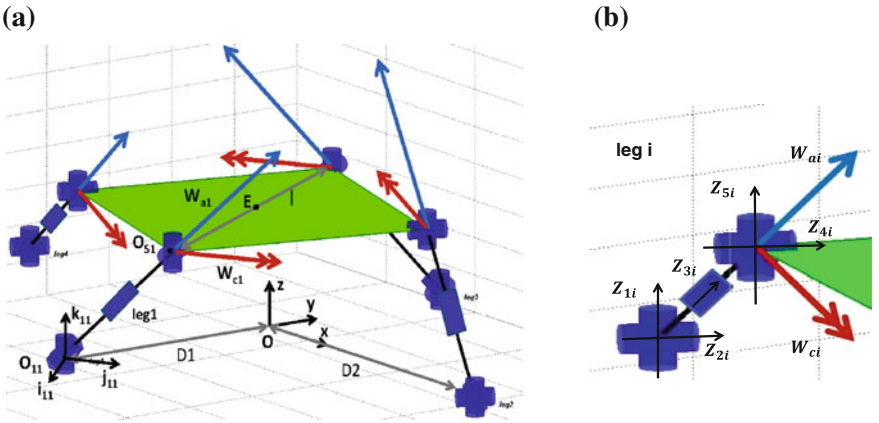


Fig. 1 Kinematics of the 4 degrees of freedom parallel manipulator. **a** 4-UPU kinematics, **b** single leg kinematics

way [2]. An example of the application of screw theory to the singularity analysis of a 4-DoF parallel manipulator can be found in [3]. A Schoenflies-type parallel manipulator with 4 UPU legs has been described in [8], and the kinematics and singularity analysis of the Jacobian has been performed.

In this chapter we extend the singularity analysis by screw theory of the 4-UPU parallel manipulator to the constraint wrenches, and the singular configurations of the legs, which can not be determined by the Jacobian analysis, has been described.

The proposed kinematics is of particular interest because, with a simple architecture, it allows to control independently the three translations of the end-effector and one rotation along a direction orthogonal to the platform. This makes it suitable for several applications, ranging from automatic machining up to haptic interfaces.

In particular, in Sect. 2 the inverse kinematics is solved, in Sect. 3 the singularities of the constraint wrenches are described, Sect. 4 concerns the singular configurations of the legs and finally Sect. 5 performs the analysis of the Jacobian.

2 4-UPU Kinematics

The proposed kinematics consists in a four legged parallel manipulator, shown in Fig. 1. Each leg is composed by an actuated prismatic joint between two universal joints, mounted in a UPU configuration (Fig. 1b). The manipulator is symmetric in respect of the x-z and y-z planes, with the base joints on the y axis at a distance D_1 from the center O and the base joints on the x axis at a distance D_2. All the base joints rotation axis are parallel to the z axis. The rotation axes of the second and fourth joints are parallel. The last joint of each leg is connected to the coupling part at a distance $l/2$ from the center E and are placed at the corner of a square of diagonal l. Moreover the rotation axis of these joints are perpendicularly connected to the

coupling part surface. According to the mobility criterion, for a generic orientation of the leg joints the manipulator is overconstrained and has not rotational degrees-of-freedom; if the last joints O_{5i} are mounted to keep always parallel to the z axis, each leg constrains a rotation in the x-y plane and the manipulator degrees-of-freedom are the three translations and the rotation around z.

The pose of the end effector E referred to the base reference system O_{xyz} is defined by the coordinates $\overline{OE} = [x_e \quad y_e \quad z_e]$ for the position and θ_e for the rotation around the z axis. The position of the points O_{5i} are calculated as $\overline{OO_{5i}} = \overline{OE} + \overline{EO_{5i}}$ with

$$\overline{EO_{5i}} = \frac{l}{2}[\sin(\theta + (i-1)\frac{\pi}{2}) \quad -\cos(\theta + (i-1)\frac{\pi}{2}) \quad 0], \qquad i = 1, \ldots, 4 \quad (1)$$

where i denotes the number of the leg. The position $\overline{OO_i}$ of the first joint of each leg is known by the geometry of the manipulator. The direction and the length of each leg is function of the pose of the end effector and results $\overline{O_{1i}O_{5i}} = \overline{OO_{5i}} - \overline{OO_{1i}}$.

Let's define the plane of the leg (defined by the versors $\hat{j}_{1i} \quad \hat{k}_{1i}$) as the plane perpendicular to the x-y plane and passing through $\overline{O_{1i}O_{5i}}$. $\hat{k}_{1i}$ is the versor parallel to the z axis, while $\hat{j}_{1i}$ can be calculated normalizing the projection $\overline{O_{1i}O_{5i_{xy}}}$ of $\overline{O_{1i}O_{5i}}$ on the x-y plane. The perpendicular to the leg plane is defined by the versor $\hat{i}_{1i} = \hat{j}_{1i} \wedge \hat{k}_{1i}$. The position and axis of each joint can be now defined by the pose of the end effector and the geometric dimensions using the above-mentioned relation.

By screw theory for each joint of a leg the twist $\$_{ji}$ is associated, using Plücker coordinates and considering the origin of the base system O as the pole for representation of screws; j denotes the number of the joint and i the number of the leg. All the twists of a leg can be arranged in the system of twists $\$_i$:

$$\$_i = \begin{pmatrix} \$_{1i} & \$_{2i} & \$_{3i} & \$_{4i} & \$_{5i} \end{pmatrix}. \quad (2)$$

The constraint wrench of the leg $\mathbf{W}_{ci}$ is then calculated as the reciprocal of the system $\$_i$, so that $\mathbf{W}_{ci} \otimes \$_{ji} = 0$, for $j = 1 \ldots 5$, where $\otimes$ denotes the product of reciprocity between screws.

Analogously the actuation wrench of the leg $\mathbf{W}_{ai}$ is calculated as the wrench reciprocal to the leg twists except the twist associated at the actuated joint:

$$\begin{cases} \mathbf{W}_{ai} \otimes \$_{ji} = 0 & for \ j = 1 \ldots 5, \quad j \neq 3 \\ \mathbf{W}_{ai} \otimes \$_{ji} \neq 0 & for \ j = 3 \end{cases} \quad (3)$$

with the actuation at the third joint.

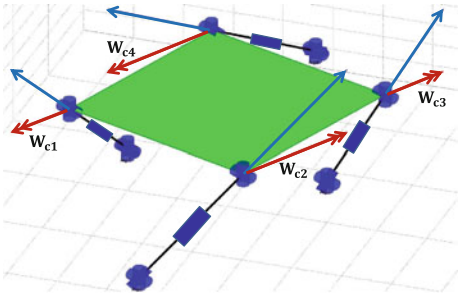


Fig. 2 Singular configuration of the kinematics with all the constraint wrenches parallel

3 Analysis of the Constraint Wrenches

Let's arrange the constraint wrenches of ∞ pitch $\mathbf{W}_{ci}$ in the matrix J_c, hereafter called constraint Jacobian:

$$J_c = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ x + \frac{l}{2}\sin\theta & x - D_2 + \frac{l}{2}\cos\theta & x - \frac{l}{2}\sin\theta & x + D_2 - \frac{l}{2}\cos\theta \\ y + D_1 - \frac{l}{2}\cos\theta & y + \frac{l}{2}\sin\theta & y - D_1 + \frac{l}{2}\cos\theta & y - \frac{l}{2}\sin\theta \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{4}$$

D_1 and D_2 are the distance of the base joints from the center, measured along y and x axis respectively (Fig. 1). Equation (4) holds only if any leg is not in singularity. Since the constraint wrenches always lay on the x-y plane and have ∞ pitch, the rank of J_c cannot be higher than two. If all the wrenches are aligned, as depicted in Fig. 2, the rank of J_c is equal to one and the manipulator presents a singularity.

This condition occurs when all the parameters satisfy the following system:

$$\left(x + \frac{l}{2}\sin\theta\right)\left(y + \frac{l}{2}\sin\theta\right) - \left(y + D_1 - \frac{l}{2}\cos\theta\right)\left(x - D_2 + \frac{l}{2}\cos\theta\right) = 0 \tag{5}$$

$$xl\cos\theta + yl\sin\theta - 2xD_1 = 0 \tag{6}$$

$$-xl\sin\theta + yl\cos\theta - 2yD_2 = 0. \tag{7}$$

The three equations above express the cross product between the first and second columns, the first and third columns, and the second and forth columns of J_c, respectively. By computing the terms xD_1 and yD_2 from Eqs. (6) and (7) and substituting into (5), we obtain the following singularity condition:

$$D_1 D_2 - D_1\frac{l}{2}\cos\theta - D_2\frac{l}{2}\cos\theta + \frac{l^2}{4} = 0. \tag{8}$$

The condition is verified if and only if

$$2D_1 \leq l \leq 2D_2 \text{ if } D_1 \leq D_2 \quad \text{or} \quad 2D_2 \leq l \leq 2D_1 \text{ if } D_2 \leq D_1. \tag{9}$$

This is a relevant result, since we can state that the constraint wrenches $\mathbf{W}_{ci}$ are not in singularity for any pose of the manipulator, when conditions (9) are not verified. On the other side when it holds one of the conditions in (9), the solution of Eq. (5) is given by D_1, D_2, l and θ that fulfill:

$$\theta = \pm \arccos\left(\frac{l^2 + 4D_1D_2}{2l(D_1 + D_2)}\right). \tag{10}$$

When Eq. (10) holds, Eqs. (6) and (7) describe the same line through the origin in the x-y plane. In a plane parallel to the x-y plane for every value of the z coordinate the singularity condition is then verified only at the origin or in the following cases:

- $D_1 = \frac{l}{2} \neq D_2, \Rightarrow \theta = 0$ and $y = 0$. The singularity locus is the x-z plane.
- $D_2 = \frac{l}{2} \neq D_1, \Rightarrow \theta = 0$ and $x = 0$. The singularity locus is the y-z plane.
- $D_1 = D_2 = \frac{l}{2}, \Rightarrow \theta = 0$. The singularity locus is the whole space.

In these particular cases at least one leg is in singularity because the first and the last joints are aligned. The orientation of the leg and then the constraint wrench cannot be univocally determined.

4 Legs Singularities

In this section the singular configurations of the legs of the manipulator are investigated. They cannot be detected analyzing the Jacobian of the manipulator, since for these configurations the kinematics of the manipulator is altered.

Single leg singularity. When a leg is in singularity with the first and last joint aligned, so that the leg is aligned to the z axis as shown in Fig. 3, one constraint wrench is added:

$$\mathbf{W}_b = \begin{bmatrix} a & b & 0 & 0 & 0 & c \end{bmatrix} \qquad a, b, c \in R. \tag{11}$$

The rank of the matrix J_c is then three, and the manipulator looses a degree of freedom. The terms a, b and c are not univocally determined because it is not possible to define a leg plane, so that the leg can arbitrarily rotate to constrain a translation in the x-y plane. If the end effector rotates around the axis of the leg, the manipulator remains in singularity.

Two legs singularity. Two opposite legs are in singularity if $D_1 = \frac{l}{2} \neq D_2$ or $D_2 = \frac{l}{2} \neq D_1$, $\theta = 0$ and $x = y = 0$, condition that already represents a singularity configuration for J_c. With two legs in singularity two constraint wrenches $\mathbf{W}_{bi}$, of the same type of the wrench in (11), are added.

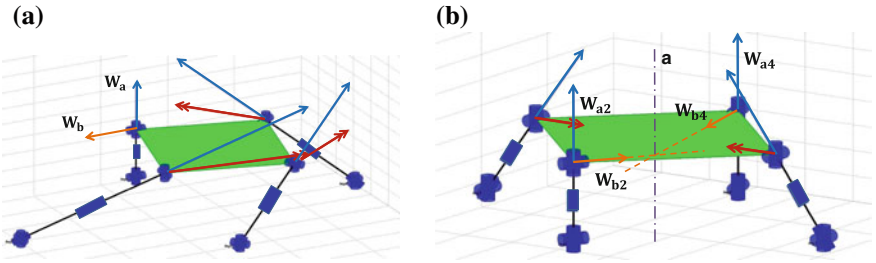


Fig. 3 Legs singularities. **a** One leg in singularity, **b** two legs in singularity

Even in this case the orientation of both the legs are not univocally determined. If the legs in singularity have the same orientation, but are not oriented as the other legs, the manipulator looses two degrees of freedom because a translation in the x-y plane and the rotation around z are constrained. If the legs in singularity are oriented as the other ones the rotation around the same direction of the constrained translation is allowed.

If the legs in singularity have not the same orientation, the manipulator looses two degrees of freedom because it can only translate along z and rotate around the axis a. a is the axis parallel to z passing through the intersection of the two additional wrenches $\mathbf{W}_{bi}$ (Fig. 3b).

Four legs singularity. Because of the geometry of the manipulator, if three legs can be aligned along z axis even the forth leg is aligned. This holds if $D_1 = D_2 = \frac{l}{2}$, $\theta = 0$ and $x = y = 0$, and causes the indetermination of the orientation of all the constraint wrenches. In the general case that all the legs have different orientations, four additional wrenches are added and the manipulator can only translate along z. If the additional wrenches $\mathbf{W}_{bi}$ intersect in one point, the end effector can rotate around an axis parallel to z and passing through the intersection point. In the particular case all the leg planes are parallel, the rotation around z and the translation parallel to $\mathbf{W}_{bi}$ are constrained, but the rotation around the direction of $\mathbf{W}_{bi}$ is not constrained.

5 Jacobian Analysis

Assuming the prismatic joints are actuated, we obtain four wrenches $\mathbf{W}_{ai}$ applied to the contact point between the leg and the end effector with the same direction of the leg, representing the forces that each leg applies to the end effector. We can define the actuation Jacobian J as the matrix that express the relation between the force at the actuated joints τ_i and the wrench at the end effector $\mathbf{W}_e$, with $\mathbf{W}_e = J^{-T}\tau$.

$$J^{-T} = \begin{pmatrix} \mathbf{W}_{a1} & \mathbf{W}_{a2} & \mathbf{W}_{a3} & \mathbf{W}_{a4} \end{pmatrix} \qquad \tau = [\tau_1 \quad \tau_2 \quad \tau_3 \quad \tau_4]^T \tag{12}$$

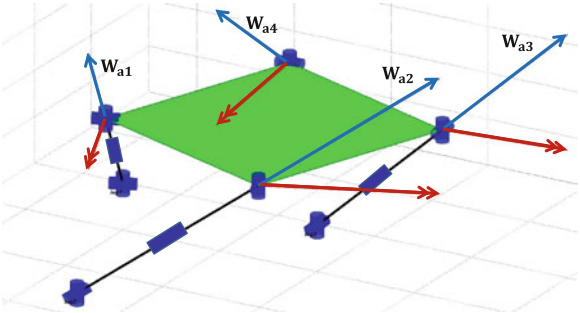


Fig. 4 Singular configuration of the kinematics leading to $Det\left(J^{-T}\right)= 0$

Since the fourth and fifth rows represent the reaction torques of the structure, described by the constraint wrenches, they are not relevant for the actuation analysis and can be neglected, obtaining a 4 by 4 square matrix. The determinant of the square matrix can be calculated as follows:

$$\mathrm{Det}\left(J^{-T}\right)= -z(D_1 - D_2)\frac{l}{2}\sin\theta\left(D_1 D_2 - D_1\frac{l}{2}\cos\theta - D_2\frac{l}{2}\cos\theta + \frac{l^2}{4}\right) \quad (13)$$

and setting $\mathrm{Det}\left(J^{-T}\right)= 0$ the singularity loci can be determined. If $\mathrm{Det}\left(J^{-T}\right)= 0$ means that exists a non null τ producing a null $\mathbf{W}_e$, i.e. a particular load at the end effector that cannot be balanced by the forces at the actuators .

Since for finite values of z, D_1, D_2 and l, $\mathrm{Det}\left(J^{-T}\right)$ is always upper bounded, it follows that $\mathrm{Det}\left(J^{T}\right)\neq 0$. This means that does not exist in any configuration a load $\mathbf{W}_e$ at the end effector that is balanced by the reaction of the structure and not by the actuators, or that the manipulator never looses mobility in the workspace.

The above-mentioned statements hold when any leg is not in singularity.

When the rank of J^{-T} is less then 4, the wrenches $\mathbf{W}_{ai}$ in general are disposed on the regulus of an hyperboloid, as shown in Fig. 4. $\mathrm{Det}\left(J^{-T}\right)= 0$ holds in the following cases: $z = 0$; $\theta = 0$; $D_1 = D_2$; $4D_1 D_2 - 2l\cos\theta(D_1 + D_2) + l^2 = 0$.

The latter condition, as for the analysis of J_c, is verified if and only if conditions given in Eq. (9) are verified, and so for values of θ fulfilling Eq. (10).

So the singular configurations for J_a include all the singular configurations for J_c. For the actuation Jacobian it is not possible to avoid the singularities in any configuration of the manipulator choosing proper values of the geometric parameters. In fact, by Eq. (13), $\mathrm{Det}\left(J^{-T}\right)$ is null for any value of D_1, D_2 and l if $z = 0$ or $\theta = 0$.

6 Conclusion

The kinematics of a 4-UPU fully parallel manipulator has been solved and the singularity loci has been investigated. By the analysis of the constraint Jacobian J_c, the singular conditions for the wrenches $\mathbf{W}_{ci}$ are determined: in these configurations the manipulator looses a constraint and it is not isostatic. Analogously, the determinant of the actuation Jacobian J_a has been studied to define the singular conditions for the wrenches $\mathbf{W}_{ai}$: in these configurations a particular load at the end effector, that cannot be balanced by any forces at the actuators, exists. As a relevant result, the singular configurations for J_a include all the singular configurations for J_c. Moreover the singular configurations for the legs of the manipulator, that cannot be determined by the Jacobian analysis, has been defined, finding further singularities for the manipulator.

Acknowledgments This work has been funded from the EU FP7 project n. 601165 WEARHAP.

References

1. Fang, Y., Tsai, L.W.: Structure synthesis of a class of 4-dof and 5-dof parallel manipulators with identical limb structures. Int. J. Rob. Res. **21**(9), 799–810 (2002)
2. Gallardo, J., Rico, J., Frisoli, A., Checcacci, D., Bergamasco, M.: Dynamics of parallel manipulators by means of screw theory. Mech. Mach. Theory **38**(11), 1113–1131 (2003)
3. Gallardo-Alvarado, J., Rico-Martínez, J.M., Alici, G.: Kinematics and singularity analyses of a 4-dof parallel manipulator using screw theory. Mech. Mach. Theory **41**(9), 1048–1061 (2006)
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**(11), 1395–1411 (2002)
5. Li, Q., Huang, Z.: Type synthesis of 4-dof parallel manipulators. In: Proceedings of ICRA 2003 IEEE International Conference on Robotics and Automation, vol. 1, pp. 755–760 (2003)
6. Pierrot, F., et al.: H4: a new family of 4-dof parallel robots. In: Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 508–513 (1999)
7. Wen-Jia, C., Ming-Yang, Z., Shu-Hong, C., Hong-Guang, W., Zhi-Gang, X., Li-Jin, F.: A novel 4-dof parallel manipulator and its kinematic modelling. In: Proceedings of ICRA 2001 IEEE International Conference on Robotics and Automation, vol. 4, pp. 3350–3355 (2001)
8. Zhao, J.S., Fu, Y.Z., Zhou, K., Feng, Z.J.: Mobility properties of a schoenflies-type parallel manipulator. Rob. Comput. Integr. Manuf. **22**(2), 124–133 (2006)
9. Zoppi, M., Zlatanov, D., Gosselin, C.M.: Analytical kinematics models and special geometries of a class of 4-dof parallel mechanisms. IEEE Trans. Rob. **21**(6), 1046–1055 (2005)

On the Kinematics of an Innovative Parallel Robot for Brachytherapy

Bogdan Gherman, Nicolae Plitea, Bogdan Galdau, Calin Vaida and Doina Pisla

Abstract The chapter presents the kinematics of a new parallel robot for brachytherapy. Brachytherapy (BT) is an innovative technique called also internal radiation, which enables the physician to deliver higher doses of radiation to very-specific areas of the body. Nowadays, BT usage is limited by the insufficient accuracy of the radioactive seeds placement devices. Thus, the authors propose an innovative modular parallel structure which overcomes these limitations, enabling the high accurate positioning of the BT needles in any parts of the patient's body. The kinematics of the new 5-DOF parallel robot is presented. The dextrous workspace of the robot is computed. Some specific advantages of this structure and the conclusions are presented in the end.

Keywords Kinematics · Parallel robot · Robotic assisted brachytherapy

B. Gherman · N. Plitea · B. Galdau · C. Vaida · D. Pisla (✉)
Research Center for Industrial Robots Simulation and Testing, Technical University of Cluj Napoca, Cluj-Napoca, Romania
e-mail: Bogdan.Gherman@mep.utcluj.ro

N. Plitea
e-mail: Nicolae.Plitea@mep.utcluj.ro

B. Galdau
e-mail: Bogdan.Galdau@mep.utcluj.ro

C. Vaida
e-mail: Calin.Vaida@mep.utcluj.ro

D. Pisla
e-mail: Doina.Pisla@mep.utcluj.ro

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_49,

1 Introduction

With the current ageing of the population, cancer is one of the main causes of the death, representing one of the most challenging, but rewarding fields of research. In brachytherapy (BT) doctors use radioactive sources with powerful but locally concentrated radiation which must be placed very close to or inside the tumor, [2]. The only drawback of the procedure consists in the high accuracy placement requirements which limits its current use. In BT, the doctor should introduce into the tumor a number of needles with a 1.6 mm diameter, through which radioactive seeds will be introduced later. These needles are straight, rigid and must be inserted on linear trajectories (defined based on CT/MRI radiologic data). For deeply located tumors, the needle placement is impossible without visual feedback and the accuracy required by the medical procedure (of 1 mm from the target point) is very hard to achieve due to tissue and needle elasticity. Several robotic devices have been developed for specific BT tasks. EUCLIDIAN [15] is a modular robot by Yu et al. comprised of a positioning module and a surgery module. Fichtinger et al. developed TRUS [3] Guided Robotic Approach-an assisting device with 4 DOF composed of two cartesian platforms mounted on each other. MrBot [14] by Stoianovici et al is built in the form of a platform supported by articulated linear actuators in a 5 DOF parallel link structure. BrachyGuide is a 4 DOF robot compound of two-axis wrist positioned by a translation stage presented by Salcudean et al. [13]. PROSPER [7] is modular robot which consists of a 5 DOF needle positioning module and a 2 DOF needle insertion module, designed only for prostate BT. MAXIO [6] is a 5 DOF robotic arm with integrated planning, navigation and guidance for tumor ablation. An MRI compatible needle manipulator concept for cancer treatment was developed by Dubowski et al. [1]. Glozman and Shoham developed in [4] the control algorithm and implementation for a flexible needle steering to achieve a planned trajectory. It is shown that only for needle insertion, 3-DOF are sufficient, without taking into account the needle positioning and orientation of the needle. A report presented in the AAPM (American Association of Physicists in Medicine) meeting in 2010, [12] shows that robotic BT is underdeveloped as most of the solutions target only the prostate, without any device capable of performing the BT tasks on larger areas of the body. In accordance with the current reported limitations, this chapter presents a *robot able to perform general BT procedures, capable of targeting any organs in the thoracic and abdominal areas, like liver, lungs, paravertebral areas, breast, kidney, etc.* The chapter is organized as follows: Section 2 presents the innovative parallel robot for BT, based on the medical specifications and its geometric model. Section 3 presents the kinematics and some numerical and simulation results for a specific trajectory of the BT needle. Section 4 shows and discusses its workspace. The conclusions and future work are presented in the Sect. 5.

2 An Innovative Parallel Robot for Brachytherapy

Figure 1 shows the kinematic scheme of the parallel robot BR1 [10, 11]. The robot consists of two parallel modules: one with M = 3 DOF and family F = 1, of type 2CRR (PRRR-PRRR), with three active joints, similar with the one studied in [8] by Kong and Gosselin and and the second module with M = 3 DOF and family F = 1, of type CR (PRR), having two active joints. The two modules are connected to the needle holder through two universal joints having the first axis of rotation around Z axis. Based on [5, 9], the following equation is introduced:

$$M = (6 - F) \cdot N - \sum_{i=1}^{5} (i - F) \cdot C_i, \tag{1}$$

where $(i - F), i = 1 \div 5$, cannot take negative values. In our case, $F = 1, N = 8, C_5 = 5, C_4 = 5, C_3 = C_2 = C_1 = 0$, so the number of DOF of the robot is: $M = 5$. The geometrical parameters of the robot are: $T_1 \div T_6, h, H, l_1 \div l_7, d$.

For the **inverse geometric model** the coordinates of the needle tip $E(X_E, Y_E, Z_E)$ and the orientation angles of the needle ψ (rotation angle around Z-axis) and θ (rotation angle around x^*-axis) are known. Knowing also the geometrical parameters of the parallel robot BR1, the coordinates of A and B points can be determined:

$$\begin{cases} X_A = X_E - (h + l_{AB}) \cdot \sin(\theta) \cdot \cos(\psi), \\ Y_A = Y_E - (h + l_{AB}) \cdot \sin(\theta) \cdot \sin(\psi), \\ Z_A = Z_E + (h + l_{AB}) \cdot \cos(\theta). \end{cases} \tag{2}$$

respectively,

$$\begin{cases} X_B = X_E - h \cdot \sin(\theta) \cdot \cos(\psi), \\ Y_B = Y_E - h \cdot \sin(\theta) \cdot \sin(\psi), \\ Z_B = Z_E + h \cdot \cos(\theta). \end{cases} \tag{3}$$

From Fig. 1 the translational active generalized coordinates of the robot can be determined:

$$q_1 = X_A, q_3 = Y_A + l_2, q_4 = Z_B - l_5 \tag{4}$$

$$\begin{aligned} q_2 = & atan2(H - (Z_A + l_3), Y_A - l_1) \\ & + atan2\left(\sqrt{1 - \left(\frac{T_1^2 + T_7^2 - T_3^2}{2 \cdot T_1 \cdot T_7}\right)^2}, \frac{T_1^2 + T_7^2 - T_3^2}{2 \cdot T_1 \cdot T_7}\right) \end{aligned} \tag{5}$$

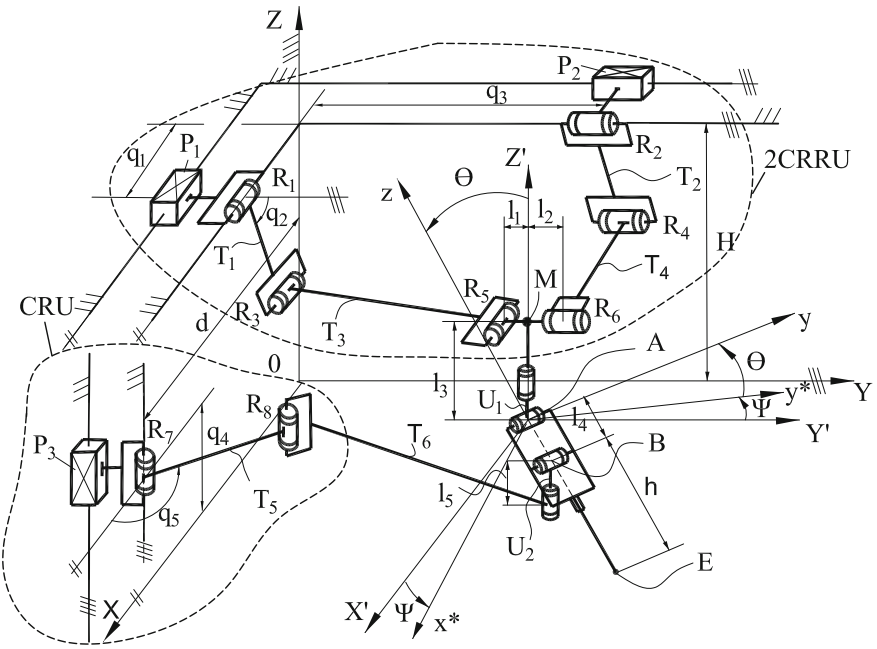


Fig. 1 Kinematic scheme of the parallel robot

$$T_7 = \sqrt{(H - Z_A - l_3)^2 + (q_3 - l_1 - l_2)^2} \tag{6}$$

$$q_5 = atan2\left(\sqrt{1 - \left(\frac{T_5^2 + l_5^2 - T_6^2}{2 \cdot T_5 \cdot l_5}\right)^2}, \frac{T_5^2 + l_5^2 - T_6^2}{2 \cdot T_5 \cdot l_5}\right) - atan2\left(\frac{Y_B}{l_5}, \frac{d - X_B}{l_5}\right) \tag{7}$$

In order to solve the **direct geometric model**, the active joints q_1, q_2, q_3, q_4, q_5 as well as the geometrical parameters are considered as known and the unknowns are the coordinates of the end-effector (the needle tip) and the needle orientation. Knowing the active joints coordinates, the T_8 and T_9 lengths can be determined :

$$T_8 = T_1 \cdot \sin\left(\frac{\pi}{2} - q_2\right), T_9 = q_3 - l_1 - l_2 - T_7 \tag{8}$$

Coordinates of A point are:

$$\begin{cases} X_A = q_1 \\ Y_A = q_3 \\ Z_A = H - T_1 \cdot \cos(\frac{\pi}{2} - q_2) - \sqrt{T_3^2 - T_9^2} - l_3. \end{cases} \tag{9}$$

θ angle results in a single solution with the equation:

$$\theta = atan2\left(\sqrt{l_4^2 - (Z_A - (q_4 + l_5))^2}, Z_A - (q_4 + l_5)\right) \tag{10}$$

Knowing the coordinates of active joints q_1, q_2, q_3, q_4, q_5, the ψ angle can be determined in the following way:

$$\psi = \psi_{\alpha_1\alpha_2} - \alpha_1 - \alpha_2 \tag{11}$$

where,

$$\psi_{\alpha_1\alpha_2} = atan2\left(\sqrt{1 - (\cos\psi_{\alpha_1\alpha_2})^2}, \cos\psi_{\alpha_1\alpha_2}\right) \tag{12}$$

$$\alpha_1 = atan2\left(\sqrt{1 - \cos\alpha_1^2}, \cos\alpha_1\right), \alpha_2 = atan2(Y_A, d - X_A) \tag{13}$$

$$\cos\psi_{\alpha_1\alpha_2} = \frac{T_{11}^2 + T_{10}^2 - T_5^2}{2 \cdot T_{10} \cdot T_{11}} \tag{14}$$

The T_{10} and T_{11} segments can be found in the following way:

$$\begin{cases} T_{10} = \sqrt{(d - T_5 \cdot \cos(q_5) - X_A)^2 + (Y_A + T_5 \cdot \sin(q_5))^2} \\ T_{11} = \sqrt{Y_A^2 + (d - X_A)^2} \end{cases} \tag{15}$$

The final equations of the needle tip coordinates are:

$$\begin{cases} X_E = X_A + (h + l_{AB}) \cdot \cos(\psi) \cdot \sin(\theta), \\ Y_E = Y_A + (h + l_{AB}) \cdot \sin(\psi) \cdot \sin(\theta), \\ Z_E = Z_A - (h + l_{AB}) \cdot \cos(\theta). \end{cases} \tag{16}$$

3 Kinematics

Kinematic equations are obtained starting from the geometric model. For BR1, we have the following implicit equations defining the tip of the BT needle:

$$\begin{cases} f_1 = q_1 - X_E + (h + l_{AB}) \cdot \sin(\theta) \cdot \cos(\psi) \\ f_2 = \cos(q_2) - \frac{1}{T_1} \cdot \left(\frac{Y_{R_5} - H + Z_{R_5}}{2} \right) \cdot \sqrt{4 \cdot \frac{T_1^2}{Y_{R_5}^2 + (H - Z_{R_5})^2} - 1} \\ f_3 = q_3 - Y_E + (h + l_{AB}) \cdot \sin(\theta) \cdot \sin(\psi) \\ f_4 = q_4 - Z_E - h \cdot \cos(\theta) + l_5 \\ f_5 = \sin(q_5) - \frac{1}{T_5} \cdot \left(\frac{d - Y_B - X_B}{2} \right) \cdot \sqrt{4 \cdot \frac{T_5^2}{d^2 + Y_B^2} - 1} \end{cases} \tag{17}$$

where using (16),

$$Y_{R_5} = Y_A - l_1, \, Z_{R_5} = Z_A + l_3 \tag{18}$$

Using the matrix representation, the kinematic model is:

$$A \cdot \dot{X} + B \cdot \dot{q} = 0, \tag{19}$$

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 (19), both the **direct kinematic model** and the **inverse kinematic model** were determined. After the calculations were achieved, an analytical solution for both geometric and kinematic model resulted due to its relatively simple mathematical model.

Some simulation results of the kinematics are presented in Fig. 2. This figure shows the displacements, velocities and accelerations of the active joints (the five motors) of the robot. The selected simulated trajectory is a linear one between the two points along the BT needle axis, so the orientation of the needle remains constant (namely the angles ψ and θ). The coordinates of the initial and final position of the end-effector are: $X_I = 465.65\,\text{mm}$, $Y_I = 222.18\,\text{mm}$, $Z_I = 1089.57\,\text{mm}$ and $X_E = 490.55\,\text{mm}$, $Y_E = 251.34\,\text{mm}$, $Z_E = 892.41\,\text{mm}$. The maximum velocity and acceleration of the needle tip are: $v_{\max} = 20$ mm/s and $a_{\max} = 10$ mm/s^2. The geometrical parameters fits to the experimental model of the robot which is going to be built: $T_1 = T_3 = 465\,\text{mm}$; $T_5 = T_6 = 400\,\text{mm}$, $h = 342.1\,\text{mm}$, $l_1 = 50\,\text{mm}$, $l_2 = 65$ mm, $l_3 = 60\,\text{mm}$, $l_4 = 112.36\,\text{mm}$, $l_5 = 40\,\text{mm}$, $d = 747\,\text{mm}$, $H = 2046.04\,\text{mm}$.

4 Workspace Modeling

In order to estimate the use of the BR1 structure for different BT procedures, an algorithm for generating the analytic workspace based on the inverse geometric model has been achieved. For the workspace generation, using the inverse geometrical model, a volume which integrates the robot workspace is initially defined. One considers that all the points in that volume are within the robot workspace, and for each point of the needle tip, the values of the active joints are determined. When a valid combination for the active joints is obtained, the set of coordinates which defined the target point is saved. For the active coordinates the following limits

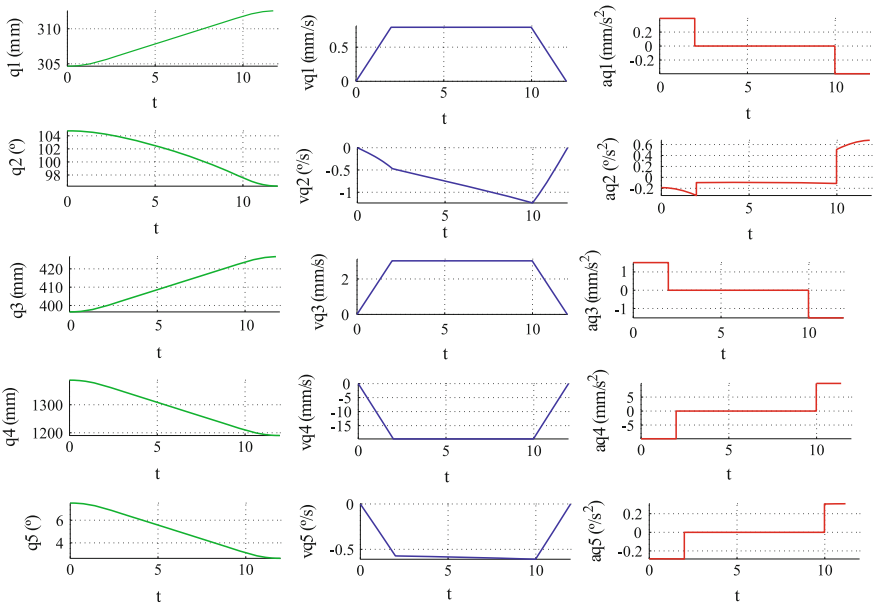


Fig. 2 Simulation results for the kinematic model of BR1

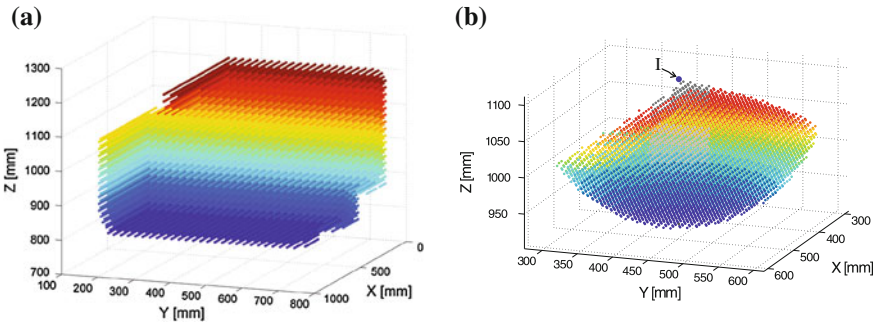


Fig. 3 Workspace of BR1 structure: **a** the dextrous workspace; **b** the workspace of BR1 structure for the insertion point I having the following coordinates: $X_I = 425\,\text{mm}$, $Y_I = 425\,\text{mm}$, $Z_I = 1102.2$ mm

were defined: $q_{i_{\min}} = 0\,\text{mm}$, $i = 1.3$, $q_{i_{\min}} = 0\,\text{rad}$, $i = 1.3$, $q_{4_{\min}} = 590\,\text{mm}$, $q_{i_{\max}} = 850\,\text{mm}$, $i = 1.3$, $q_{i_{\max}} = 2 \cdot \pi$ rad, $i = 2.5$, $q_{4_{\max}} = 1370\,\text{mm}$. The dextrous workspace has been obtained and shown in Fig. 3a, illustrating the capabilities of the robot for the given geometric parameters. In Fig. 3b, the BR1 workspace for a given insertion point $I(X_I, Y_I, Z_I)$ has been obtained, where all the robot joints are positioned in the midway of their strokes. The matrix of the validated needle tip coordinates represents the robot workspace. The color scale varies with the Z coordinate of the needle tip.

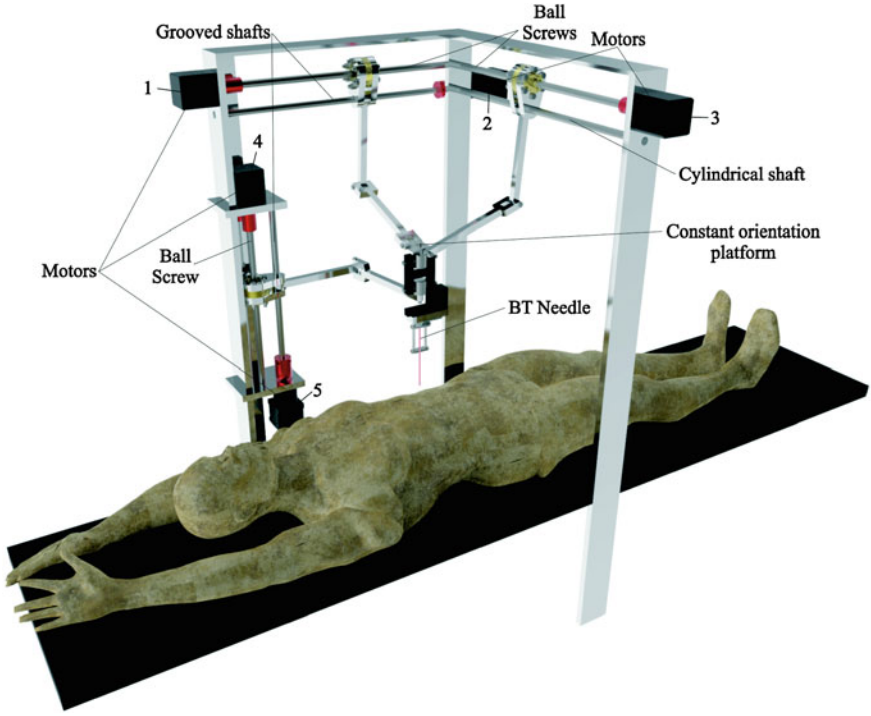


Fig. 4 CAD model of the parallel robot in the medical environment

A CAD model of the BR1 structure, integrated in the medical environment has been presented in Fig. 4, in order to illustrate its working possibilities. The position and orientation of the needle, as well as its insertion is obtained by the actuation of all five motors (motors 1, 3 and 4 for translation and motors 2 and 5 for rotation). The needle insertion motion will be obtained by rotating the ball screws or grooved shaft held by bearings connected to the fixed frame, necessary for a smooth motion at the active joints.

5 Conclusions

An innovative parallel robot designed for brachytherapy applications was presented. The structure can be used for general purpose brachytherapy procedures, to position accurately the needle, being able to cover most organs in the thoracic and abdominal areas. The geometric modeling and and some kinematic simulations are presented. The workspace analysis demonstrates the usability of the robot for the targeted medical task. Further research will focus on trajectory generation for multiple needles in

a singularity free workspace. The analytic kinematic model will be implemented in the control system of the experimental model of the robot.

Acknowledgments This chapter was supported by the Project no. 173/2012, code PN-II-PCCA-2011-3.2-0414, entitled "Robotic assisted brachytherapy, an innovative approach of inoperable cancers—CHANCE" financed by UEFISCDI and the Scopes International Grant IZ74Z0-137361/1, Creative Alliance in Research and Education focused on Medical and Service Robotics-CARE-Robotics.

References

1. Dubowsky, S., et al.: An MRI-compatible needle manipulator concept based on elastically averaged dielectric elastomer actuators for prostate cancer treatment: an accuracy and MR-compatibility evaluation in phantoms. J. Med. Devices **3**(3), 10 (2009)
2. Dueitt, B.: External beam radiotherapy, chap. Comparisons of External Beam Radiation Therapy, Brachytherapy, and Combination Therapy in the Treatment of, Prostate cancer, pp. 34–40 (2002)
3. Fichtinger, G., et al.: Robotic assistance for ultrasound guided prostate brachytherapy. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, vol. 4791, pp. 119–127 (2007)
4. Glozman, D., Shoham, M.: Image-guided robotic flexible needle steering. IEEE Trans. Robot. **23**(3), 459–467 (2007)
5. Gogu, G.: Structural Synthesis of Parallel Robots, Solid Mechanics and Its Applications, vol. 149. Springer, Dordrecht (2008)
6. http://www.perfinthealthcare.com/MaxioOverview.asp: (2012)
7. Hungr, N., et al.: A 3D ultrasound robotic prostate brachytherapy system with prostate motion tracking. IEEE Trans. Robot. **28**, 1382–1397 (2012)
8. Kong, X., Gosselin, C.: Kinematics and singularity analysis of a novel type of 3-crr 3-dof translational parallel manipulator. Int. J. Robot. Res. **21**(9), 791–798 (2002)
9. Manolescu, N.: For a united point of view in the study of the structural analysis of kinematic chains and mechanisms. J. Mech. **3**, 149–169 (1968)
10. Plitea, N., et al.: Parallel robot for brachytherapy with two kinematic guiding chains of the platform (the needle) type 2CRRU and CRU, patent pending, A/10004/2013 (2013)
11. Plitea, N., et al.: Structural analysis and synthesis of parallel robots for brachytherapy. D. Pisla et al. (eds.) New Trends in Medical and Service Robots, vol. 16, pp. 191–204. Springer, MMS, Dordrecht (2014)
12. Podder, T., Fichtinger, G.: Robotic brachytherapy: overview of robotic brachytherapy approaches and synergistic applications. AAPM Annual Meeting (2010)
13. Salcudean, S.E., et al.: A robotic needle guide for prostate brachytherapy. In: IEEE International Conference on Robotics and Automation, pp. 2975–2981 (2008)
14. Stoianovici, D., et al.: "MRI stealth" robot for prostate interventions. Minim. Invasive Ther. Allied Technol. **16**, 241–248 (2007)
15. Yu, Y., Podder, T., et al.: Robot-assisted prostate brachytherapy. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, vol. 9, pp. 41–49 (2006)

Reconfigurable and Deployable Platonic Mechanisms with a Variable Revolute Joint

Guowu Wei and Jian S. Dai

Abstract This chapter presents for the first time a variable revolute joint and a group of reconfigurable and deployable Platonic mechanisms. Structure of the variable revolute joint is presented and demonstrated by its application to the construction of a reconfigurable generic 4R linkage which is capable of converting itself to a planar parallelogram 4R linkage, a spherical 4R linkage and a Bennett linkage. Then, with a two-phase variable revolute joint, a group of reconfigurable and deployable Platonic mechanisms are constructed and mobility of the proposed reconfigurable Platonic mechanisms is investigated by formulating their corresponding constraint matrices. Finally, kinematic characteristics of the proposed mechanisms are illustrated.

Keywords Reconfigurable mechanisms · Deployable polyhedral mechanisms · Variable revolute joint · Generic 4R linkage

1 Introduction

Deployable polyhedral mechanisms (DPMs) have been raising interest from kinematicians and mathematicians since the pioneering work of Bricard on flexible polyhedrons [1] and of Verheyen [11] on the expandable polyhedral structures coined as "Jitterbug transformers". Wohlhart proposed different synthesis methods leading to the generation of various deployable polyhedral mechanisms including the regular polyhedral linkages [15] which are directly related to the mechanisms proposed in this chapter. Kiper et al. [9], and Wei and Dai [14] revisited Wohlhart's work and synthesized the polyhedral mechanisms with new approaches, and Röschel [10]

G. Wei (✉) · J. S. Dai
Centre for Robotics Research, King's College London, Strand, London WC2R 2LS, UK
e-mail: guowu.wei@kcl.ac.uk

J. S. Dai
e-mail: jian.dai@kcl.ac.uk

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_50,

investigated polyhedral mechanisms from the geometric point of view giving insight into the intrinsic geometric properties of the deployable polyhedral mechanisms. Reconfigurable mechanical systems including metamorphic mechanisms [3] satisfy the ever-growing market demands of adapting for various stipulations by changing topological configurations. In order to construct multifunctional mechanisms that are capable of changing mobility, mechanism topology and function through the metamorphic process to adapt themselves for different tasks and working environment without disassembling the mechanisms, Dai and Rees Jones [4] presented the concept of mechanism metamorphosis, Yan and Kuo [16] investigated variable topology mechanisms and kinematics pairs, Gan et al. [7] invented a reconfigurable Hooke (rT) joint, and Zhang et al. [17] proposed a variable axis (vA) joint.

In this chapter, a variable revolute (vR) joint is for the first time proposed which leads to the construction of reconfigurable mechanisms including a group of reconfigurable and deployable Platonic mechanisms.

2 A Variable Revolute Joint and a Reconfigurable Generic 4R Linkage

2.1 Structure of a Variable Revolute (vR) Joint

In the traditional mechanism design, a revolute joint is frequently employed to connect two links providing one degree of freedom relative motion and once a revolute joint is installed, direction of the axis of rotation as well as the corresponding structure parameters (e.g. D-H parameters [6]) between the two links are determined. However, in order to use revolute joints to construct reconfigurable/metamorphic mechanisms [3] having capability of changing topology structures and functions in different working stages, it is expected that the directions of their joint axes can be altered. Thus, inspired by the development of the reconfigurable Hooke joint (rT) [7] and the variable-axis (vA) joint [17], in this chapter, a variable revolute joint is invented and designed as illustrated in Fig. 1. It is denoted as vR joint, where R stands for a revolute joint and v stands for variable indicating that the orientation of the revolute joint is changeable. The joint consists of a reconfigurable connector (rC) which is rigidly attached to link *i* and contains a groove to accommodate a axis-variable revolute joint. As shown in Fig. 1a, the capacity of changing orientation of the joint axis is realized by adjusting the axis-variable revolute joint about the groove and connecting link *j* such that the relative geometric configuration between link *i* and link *j* can be consequently altered. After changing the direction of the revolute joint axis to a desired orientation, the joint is fixed by bolting it to the reconfigurable connector. Figure 1a illustrates one type of vR joint with the revolute joint embedded inside the groove, and Fig. 1b shows its variant in which the revolute joint is placed outside of the groove. Further, Fig. 1c gives the topological schematic diagram of the vR joint.

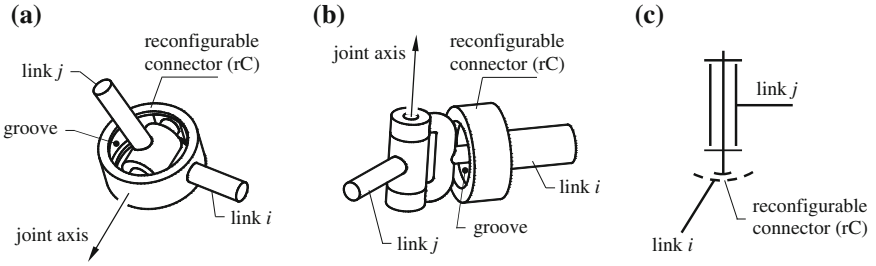


Fig. 1 Variable revolute (vR) joint and its variants

Connecting two links by a variable revolute (vR) joint, the two links can have various relative geometric configurations and thus variable relative structure parameters. Therefore, if a vR joint is used to replace the traditional R joint in a linkage, the linkage readily becomes a reconfigurable linkage that can change itself from one mechanism type to the other having different motion properties and functions.

2.2 A Reconfigurable Generic 4R Linkage with a Three-Phase Variable Revolute Joint

Figure 2a gives a conventional spherical 4R linkage with its four revolute joints arranged at the four corners of a square and their joint axes intersecting at a common point V. As shown in Fig. 2b, by replacing all R joints of the original spherical 4R linkage with vR joints, a reconfigurable spherical 4R mechanism can be obtained.

In the evolved reconfigurable spherical 4R linkage, the variable revolute (vR) joints are connected by reconfigurable connectors (rCs) which are rigidly attached to the links (Herein, for the sake of clarity, the detailed structure of the reconfigurable connectors is not illustrated in the figure, readers can refer to Fig. 1b for it). The reconfigurable connectors have the functions of releasing and locking the vR joints such that directions of the joint axes can be adjusted so as to reconfigure the types of linkages.

In the reconfigurable spherical 4R linkage, since four vR joints are distributed at four corners of a square formed by points A, B, C and D, and axes of the four joints intersect at a common point V, in the configuration shown in Fig. 2b, the angle between the axis of any joint and a line passing through point V and point O (which is the centre point of the square) equals γ. Axes of joints A and C lie in a same vertical plane, and axes of joint B and D forms another same plane π (see Fig. 2b).

We keep the directions of the axes of joints A and C unchanged and as shown in Fig. 2c, release reconfigurable connectors at joints B and D, and adjust the axis directions of vR joints B and D by rotating them respectively about axes $\boldsymbol{e}_{\mathrm{B}}$ and $\boldsymbol{e}_{\mathrm{D}}$ by angle α (Where axes $\boldsymbol{e}_{\mathrm{B}}$ and $\boldsymbol{e}_{\mathrm{D}}$ are aligned with central axes of the grooves embedded in their corresponding rCs passing through points B and D respectively,

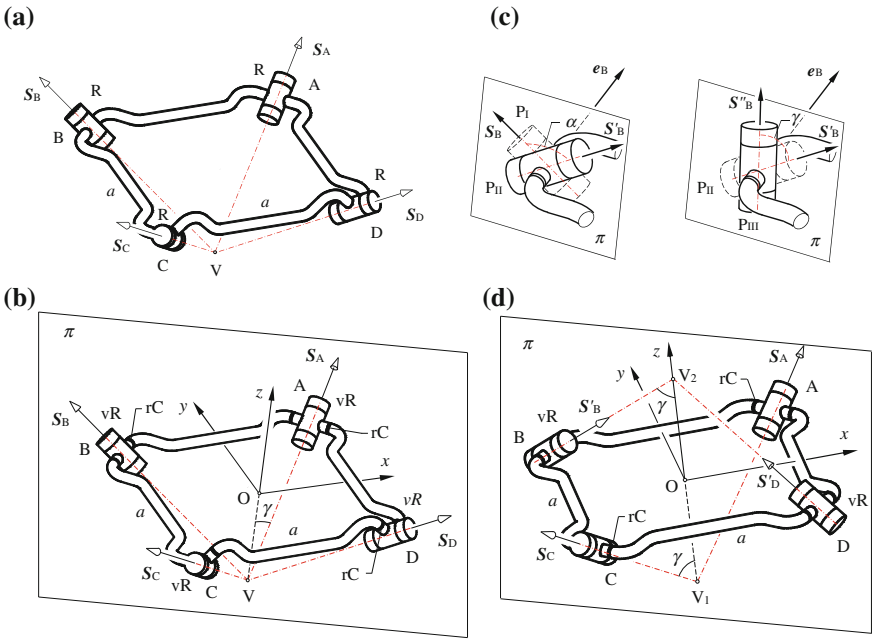


Fig. 2 A reconfigurable general 4R linkage. **a** A spherical 4R linkage. **b** Reconfiguration of vR joint from PI to PII and from PII to PIII. **c** A reconfigurable spherical 4R linkage. **d** A reconfigurable Bennett linkage (*Note* rC stands for reconfigurable connector. PI, PII and PIII stand respectively for phase I, phase II and phase III)

and are perpendicular to plane π.), then fix the vR joints B and D when they reach phase II. In such a configuration as shown in Fig. 2d, the axes of joints A and C intersect at point V_1, and the axes of joints B and D intersect at point V_2, points V_1, V_2 and O are collinear, and the angle between axis of any joint and a line passing through V_1, V_2 and O remains γ. In such a case, the reconfigured linkage turns out to be a Bennett linkage [8].

Further, releasing all the vR joints and adjusting their axis directions to form a configuration that axes of all the four joints are parallel to each other, i.e. each joint reaches phase III as shown in Fig. 2c, then fastening the reconfigurable connectors (rCs) so as to fix the vR joints, a planar parallelogram 4R linkage can be obtained and in this case the angle γ equals 0.

In the stage of Bennett linkage, mobility of the linkage can be calculated and verified by computing the dimension of nullity of the constraint matrix [12] of the linkages as

$$m = \dim\left(N\left(\mathbf{M}_c\right)\right). \tag{1}$$

Where the constraint matrix of the linkage according to its associated graph can be formulated as

$$\mathbf{M}_c = \left[\boldsymbol{S}_A \ \boldsymbol{S}_B \ \boldsymbol{S}_C \ \boldsymbol{S}_D \right], \tag{2}$$

with screws [5] for the joint axes $\boldsymbol{S}_A$, $\boldsymbol{S}_B$, $\boldsymbol{S}_C$ and $\boldsymbol{S}_D$ derived according to Fig. 2d as

$$\begin{cases} \boldsymbol{S}_A = \left[\frac{\sqrt{2}}{2}\cos\gamma \ \ \frac{\sqrt{2}}{2}\cos\gamma \ \ \sin\gamma \ \ -a\sin\gamma \ \ a\sin\gamma \ \ 0 \right]^T \\ \boldsymbol{S}_B = \left[\frac{\sqrt{2}}{2}\cos\gamma \ \ -\frac{\sqrt{2}}{2}\cos\gamma \ \ \sin\gamma \ \ -a\sin\gamma \ \ -a\sin\gamma \ \ 0 \right]^T \\ \boldsymbol{S}_C = \left[-\frac{\sqrt{2}}{2}\cos\gamma \ \ -\frac{\sqrt{2}}{2}\cos\gamma \ \ \sin\gamma \ \ a\sin\gamma \ \ -a\sin\gamma \ \ 0 \right]^T \\ \boldsymbol{S}_D = \left[-\frac{\sqrt{2}}{2}\cos\gamma \ \ \frac{\sqrt{2}}{2}\cos\gamma \ \ \sin\gamma \ \ a\sin\gamma \ \ a\sin\gamma \ \ 0 \right]^T \end{cases}. \tag{3}$$

Where a denotes the length of a link in the linkage.

Substituting Eq. (2) into Eq. (1) with joint screws provided in Eq. (3) gives $m = 1$, which indicates that the linkage at Bennett linkage type has one mobility.

Therefore, this example shows that by integrating variable revolute (vR) joints into a conventional spherical 4R linkage, a reconfigurable generic 4R linkage can be generated which is capable of transforming itself into a planar 4R linkage, a spherical 4R linkage and a Bennett Linkage by adjusting the directions of the joint axes through the reconfigurable connectors (rCs). In order to precisely place axes of the joints in the correct directions, three slots can be fabricated in the reconfigurable connectors which clearly define three phases for each vR joint. In such a way, starting from the spherical 4R linkage configuration shown in Fig. 2b, keeping either pair of joints B and D, or joints A and C unchanged, and adjusting the other pair from phase I to phase II as shown in Fig. 2c, the linkage transforms from a spherical 4R linkage into a Bennett linkage, and vice versa. If all joints are locked at phase III, the linkage turns out to be a planar parallelogram 4R linkage.

As a result, this section demonstrates that the proposed vR joint can be used to replace the traditional R joint leading to reconfigurable linkages/mechanisms which can be converted into linkages/mechanisms of different types that perform diverse motions and functions. In Sect. 3, it shows that the vR joint can lead to the construction of a group of reconfigurable and deployable Platonic mechanisms.

3 Construction of Reconfigurable and Deployable Platonic Mechanisms with a Two-Phase vR Joint

In the previous work, based on a dual-plane-symmetric spatial eight-bar linkage [14] and an overconstrained [2] spatial eight-bar linkage [13], the authors of this chapter synthesized a group of deployable Platonic mechanisms with radially reciprocating motion as well as a group of Fulleroid-like deployable Platonic mechanisms. All the mechanisms are constructed by purely using R joints to connect the links. Figure 3 shows examples of the two types of Platonic mechanisms, i.e. a deployable dodecahedral mechanism with radially reciprocating motion and a Fulleroid-like deployable

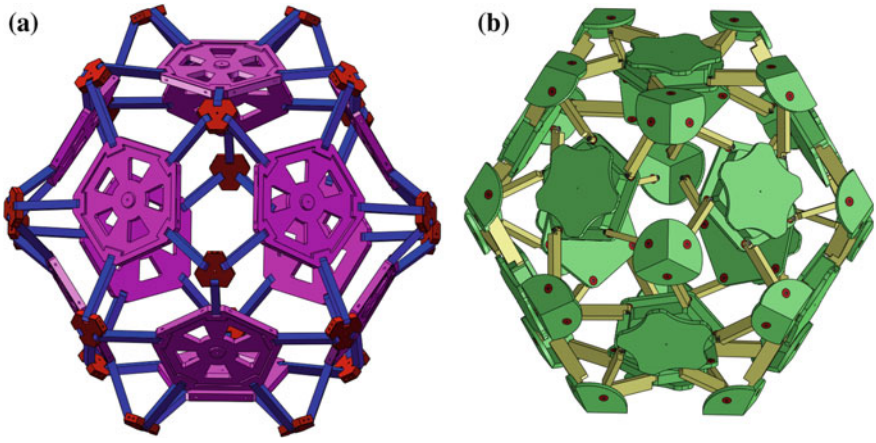


Fig. 3 Two different types of deployable dodecahedral mechanisms. **a** A deployable dodecahedral mechanism with radially reciprocating motion. **b** A Fulleroid-like deployable dodecahedral mechanism

dodecahedral mechanism. As indicated in Fig. 3a, for the deployable dodecahedral mechanism with radially reciprocating motion, in each pentagonal facet all R joints are parallel to the facet, while, for the Fulleroid-like dodecahedral mechanism (see Fig. 3b), in each pentagonal facet all R joints are perpendicular to the facet.

In this chapter, it is found that by replacing all R joints in either type of aforementioned deployable Platonic mechanisms with two-phase vR joints, a group of reconfigurable and deployable Platonic mechanisms can be constructed. The mechanisms are capable of transforming themselves from one type to the other executing motions and functions possessed by both types of aforementioned deployable Platonic mechanisms.

Figure 4 shows a reconfigurable and deployable tetrahedral mechanism which is constructed by using a two-phase vR joint. In the mechanism, both ends of each link are connected by vR joints which are mounted in two reconfigurable connectors (rCs), one rC is rigidly embedded in the facet component and the other rC is rigidly fixed in the vertex component. Figure 4a shows the mechanism in a Fulleroid-like deployable tetrahedral mechanism type, in which the vR joints are placed in their phase I configurations such that all joint axes are perpendicular to their corresponding facet as shown in Fig. 4b. Then, moving the mechanism to the configuration that the centre axes of the two rCs at both ends of each link are collinear (see Fig. 4b) and adjusting all the vR joints from phase I to phase II configurations such that all joint axes are parallel to their corresponding facet as shown in Fig. 4c, subsequently the mechanism turns out to be in another type which is capable of performing radially reciprocating motion.

Hence, by integrating the variable revolute (vR) joints into a deployable tetrahedral mechanism, the mechanism becomes a reconfigurable one which has the capacity of converting itself from one type to another and performing diverse functions. For the

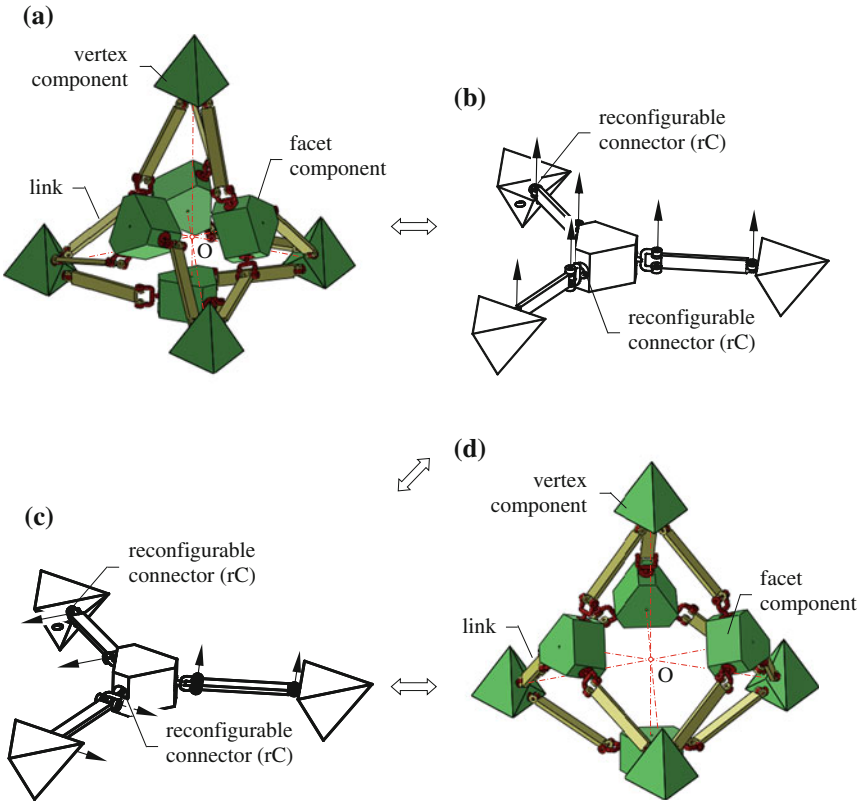


Fig. 4 A reconfigurable and deployable tetrahedral mechanism. **a** In Fulleroid-like linkage type. **b** vR joints in phase I configuration. **c** vR joints in phase II configuration. **d** In star-transformer linkage type

Fulleroid-like tetrahedral mechanism in Fig. 4a, all facet components perform screw motions around the axes that are perpendicular to their corresponding facets and pass centroid point O, and all vertex components execute radially reciprocating motions along the axes passing through their corresponding vertexes and point O. However, for the other type of tetrahedral mechanism shown in Fig. 4d, all components carry out radially reciprocating motions along the aforementioned axes towards/outwards the centroid point O forming star-like tetrahedron which is one of the so called polyhedral star-transformers [15].

Moreover, by applying the two-phase vR joints to the deployable Platonic mechanisms developed by Wohlhart [15] and Wei and Dai [13, 14], a group of reconfigurable and deployable Platonic mechanisms can be constructed as illustrated in Fig. 5. All the reconfigurable and deployable Platonic mechanisms can transform themselves without disassembly from a Fulleroid-like mechanism type (see left column in Fig. 5) to a star-transformer mechanism type (see right column in Fig. 5) or vice versa.

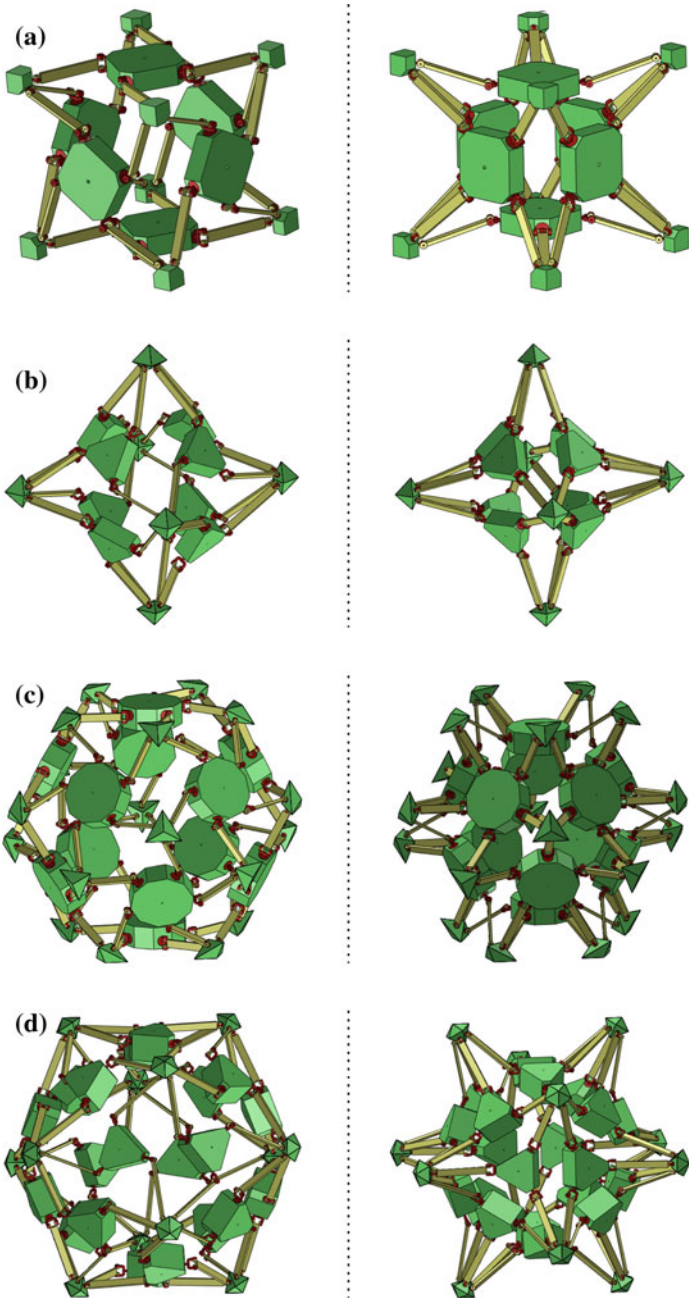


Fig. 5 A group of reconfigurable and deployable Platonic mechanisms. **a** A reconfigurable and deployable hexahedral mechanism and its two types. **b** A reconfigurable and deployable octahedral mechanism and its two types. **c** A reconfigurable and deployable dodecahedral mechanism and its two types. **d** A reconfigurable and deployable icosahedral mechanism and its two types

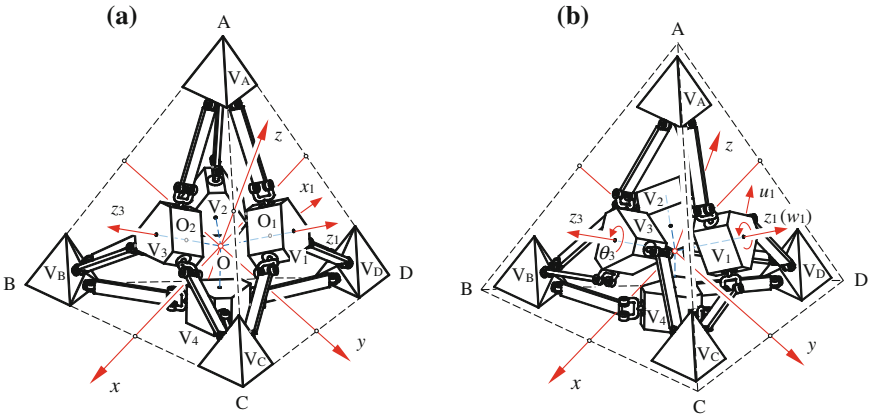


Fig. 6 Geometry and coordinate systems of a Fulleroid-like tetrahedral mechanism. **a** Initial configuration and coordinate systems. **b** A random configuration and coordinate systems

4 Mobility and Kinematics of the Reconfigurable Platonic Mechanisms

For the reconfigurable Platonic mechanisms proposed in this chapter, when the mechanisms are in the star-transformer type, mobility of the mechanisms was verified by the present authors in Ref. [12] and kinematics of the mechanisms was indicated in Ref. [14]. When the reconfigurable mechanisms are in the Fulleroid -like type, mobility and kinematics of the mechanisms can be similarly formulated and analysed.

By following the mobility analysis of star-transformer type mechanisms [12], referring to the coordinate systems provided in Fig. 6 and the joint screws in individual facet presented in Fig. 7a, constraint matrix $\mathbf{M}'_c$ of the Fulleriod-like tetrahedral mechanism can be formulated with the joint screws in the individual facet being modified as

$$\begin{cases} \boldsymbol{S}_{i1} = [0\,0\,1\;b\;0\,0]^T\,,\; \boldsymbol{S}_{i2} = \left[0\,0\,1\;\; -b/2\;\sqrt{3}b/2\,0\right]^T \\ \boldsymbol{S}_{i3} = \left[0\,0\,1\;\; -b/2\;\; -\sqrt{3}b/2\,0\right]^T\,,\; \boldsymbol{S}'_{i1} = [0\,0\,1\;b + l\cos\beta\;\; -l\sin\beta\;0]^T\,, \\ \boldsymbol{S}'_{i2} = \left[0\,0\,1\;\; -b/2 - l(\cos\beta - \sqrt{3}\sin\beta)/2\;\sqrt{3}b/2 + l(\sqrt{3}\cos\beta + \sin\beta)/2\,0\right]^T\,, \\ \boldsymbol{S}'_{i3} = \left[0\,0\,1\;\; -b/2 - l(\cos\beta + \sqrt{3}\sin\beta)/2\;\; -\sqrt{3}b/2 - l(\sqrt{3}\cos\beta - \sin\beta)/2\,0\right]^T\,. \end{cases} \tag{4}$$

Mobility of the mechanism can then be obtained as $m = \dim(N(\mathbf{M}'_c)) = 1$.

Further, using the coordinate systems established in Fig. 6, kinematic analysis of the Fulleroid-like deployable tetrahedral mechanism can be carried out leading to

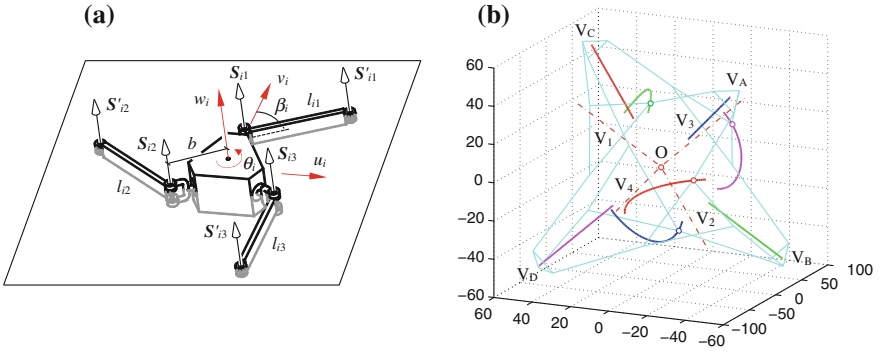


Fig. 7 Joint screws in individual facet and kinematics of the mechanism. **a** Geometry and joint screws in individual facet. **b** Trajectories of the facet and vertex components

the illustration of kinematic characteristics of the mechanism as shown in Fig. 7b. Figure 7b indicates that the Fulleroid-like tetrahedral mechanism has such a property that the facet components execute screw motions around the axes that are perpendicular to their corresponding facets and pass centroid point O, and vertex components execute radially reciprocating motions along the axes passing through their corresponding vertexes and point O. Trajectories of the motions are illustrated in Fig. 7b.

The above mobility and kinematic analysis method can then be extended to the whole group of reconfigurable Platonic mechanisms when they work in the Fulleroid-like mechanism type. Integrating these analysis with those investigations of mobility and kinematics for the star-transformer type mechanisms in [12, 14], kinematic properties of all the reconfigurable and deployable Platonic mechanisms can be fully characterized.

5 Conclusions

In this chapter, a variable revolute (vR) joint was for the first time proposed and its structure was presented. Using a three-phase vR joint, a reconfigurable generic 4R linkage was developed verifying the application of the proposed vR joint. The generic 4R linkage can transform itself into a planar 4R linkage, a spherical 4R linkage and a Bennett linkage. Subsequently, with a two-phase variable revolute joint, a group of reconfigurable and deployable Platonic mechanisms were constructed which have the capacity of converting themselves from the Fulleroid-like linkage type to the star-transformer linkage type performing screw motion integrated with radially reciprocating motion or pure radially reciprocating motion. Mobility of the group of reconfigurable Platonic mechanisms was further investigated and kinematics properties of the mechanisms were characterized with numerical simulation.

Acknowledgments The authors gratefully acknowledge the support from the EU 7th Framework Programme TOMSY under grant No.270436, and the support from the National Natural Science Foundation of China (NSFC) under grant No.51175366.

References

1. Bricard, R.: Mémoire sur la théorie de l'octaèdre articulé. J. Math. Pure Appl. Liouville **3**, 113–148 (1897)
2. Cui, L., Dai, J. S.: Axis constraint analysis and its resultant 6R double-centered overconstrained mechanisms. ASME J. Mech. Robot. **3**(3), 031004 (2011)
3. Dai, J.S., Rees Jones, J.: Mobility in metamorphic mechanisms of foldable/erectable kinds. Trans. ASME: J. Mech. Des. **121**(3), 375–382 (1999)
4. Dai, J.S.: Matrix representation of topological configuration transformation of metamorphic mechanisms. ASME J. Mech. Des. **127**(4), 837–840 (2005)
5. Dai, J.S.: Finite displacement screw operators with embedded Chasles' motion. ASME J. Mech. Robot. **4**(4), 041002 (2012)
6. Denavit, J., Hartenberg, R.S.: A kinematic notation for lower pair mechanisms based on matrices. ASME J. Appl. Mech. **22**(2), 215–221 (1955)
7. Gan, D., Dai, J.S., Liao, Q.: Contraint analysis on mobility change of a novel metamorphic parallel mechanism. Mech. Mach. Theory **45**, 1864–1876 (2010)
8. Hunt, K.H.: Kinematic Geometry of Mechanisms. Clarendon Press, Oxford (1978)
9. Kiper, G., Söylemez, E., Kisisel, A.U.O.: Polyhedral linkages synthesized using cardan motion along radial axes. In: Proceedings of the 12th IFToMM World Congress. Besancon (2007)
10. Röschel, O.: Möbius mechanisms. In: J. Lenarčič, M. Stanišić (eds.) Advances in Robot Kinematics, pp. 375–382. Kluwer Akademie, London (2000)
11. Verheyen, H.F.: The complete set of jitterbug transformers and the analysis of their motion. Comput. Math. Appl. **17**(1–3), 203–250 (1989)
12. Wei, G., Dai, J.S.: Synthesis of a family of regular deployable polyhedral mechanisms. In: Lenarčič, J., Husty, M. (eds.) Latest Advances in Robot Kinematics, pp. 123–130. Springer, New York (2012)
13. Wei, G., Dai, J.S.: An overconstrained eight-bar linkage and its associated fulleroid-like deployable platonic mechanisms. In: 38th Mechanisms and Robotics Conference (2014)
14. Wei, G., Dai, J.S.: A spatial eight-bar linkage and its association with the deployable platonic mechanisms. ASME J. Mech. Robot. **6**(2), 021010 (2014)
15. Wohlhart, K.: Regular polyhedral linkages. In: Proceedings of the 2nd Workshop on Computational Kinematics, pp. 239–248 (2001)
16. Yan, H.S., Kuo, C.H.: Topological representations and characteristics of variable kinematic joints. Trans. ASME: J. Mech. Des. **128**(2), 384–391 (2006)
17. Zhang, K., Dai, J.S., Fang, Y.: Topology and constraint analysis of phase change in the metamorphic chain and its evolved mechanism. ASME J. Mech. Des. **132**(12), 121001 (2010)

Conditions for Sub-6th Order Screw Systems Composed of Three Planar Pencils of Lines

Xianwen Kong and Duanling Li

Abstract Sub-6th order screw systems composed of (the sum of) three planar pencils of lines (PPLs)—3-PPL-systems—are closely related to the static singularity (or forward kinematic singularity) analysis of a number of 3-legged parallel manipulators and inverse kinematic singularity analysis of a class of hybrid manipulators. This paper presents an alternative simple approach to the derivation of the conditions for sub-6th order 3-PPL-systems. The characteristics of fourth order 2-PPL-systems are first revealed by using a reciprocal-screw-system based approach. By decomposing a 3-PPL-system as the sum of a 2-PPL-system and a 1-PPL-system, conditions for sub-6th order 3-PPL-systems can then be derived based on the intersection of screw systems. This paper also contributes to the classification of 3-PPL-systems.

Keywords Parallel manipulator · Hybrid manipulator · Screw theory · Singularity

1 Introduction

Sub-6th order screw systems composed of (the sum of) three planar pencils of lines (PPLs)—3-PPL-systems—are closely related to (a) the static singularity (or forward kinematic singularity) analysis of a number of 3-legged parallel manipulators (PMs) [1–11], in which the unactuated joints in each leg are an S (spherical) joint and an unactuated X joint that can be in the form of any single-DOF (degree-of-freedom) kinematic joint such as an R (revolute) joint and a P (prismatic) joint or the form of

X. Kong (✉)
Heriot-Watt University, Edinburgh, UK
e-mail: X.Kong@hw.ac.uk

D. Li
Beijing University of Posts and Telecommunications, Beijing, China
e-mail: liduanling@163.com

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_51,

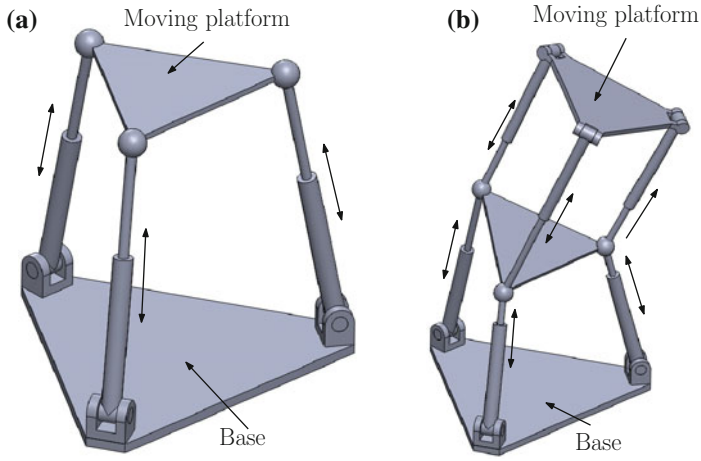


Fig. 1 Parallel and hybrid manipulators associated with 3-PPL-systems: **a** 3-RPS parallel manipulator; **b** 3-RPS-3-SPR hybrid manipulator

any single-DOF closed kinematic chain such as a parallelogram, and (b) the inverse kinematic singularity analysis of a class of hybrid manipulators [12, 13].

Figure 1a shows a 3-RPS PM, which is composed of three RPS legs. Each leg is composed of one actuated prismatic joint (P), one unactuated R joint and one unactuated S joint. In a forward kinematic singular configuration, the moving platform of the PM may undergo infinitesimal motion even if the actuated joints are locked. Figure 1b shows a 3-RPS-3-SPR hybrid manipulator composed of a 3-RPS parallel module and a 3-SPR parallel module connected serially. In an inverse kinematic singular configuration, the intermediate platform of the hybrid manipulator may undergo infinitesimal motion even if the moving platform is fixed and the actuated joints are released. In other words, the moving platform of the hybrid manipulator loses at least one DOF in this singular configuration.

The geometric characteristics of the static singular configuration of this class of PMs, i.e., the conditions for sub-6th order 3-PPL-systems, have been revealed using different approaches [1–11]. Systematic classification of 3-PPL-systems have been presented in [1–3, 11]. The recent method in [11] for revealing the geometric characteristics of the singular configuration of the above PMs was based on the basic manipulations of screw systems and reciprocal systems with fewer derivation.

Inspired by the successful application of the following equation in the type synthesis of PMs (see [15] for example), this paper aims at further simplifying the approach to the derivation of the conditions for sub-6th order 3-PPL-systems.

$$\mathscr{S} = (\mathscr{S}^{\perp})^{\perp} \tag{1}$$

where $\mathscr{S}$ and $\mathscr{S}^{\perp}$ denote a screw system and its reciprocal screw system.

By decomposing a 3-PPL-system as the sum of a 2-PPL-system and a 1-PPL-system, the conditions for sub-6th order 3-PPL-systems can be derived using the basic manipulations on screw systems, including the intersection of screw systems and reciprocal screw systems. In Sect. 2, the essential results from screw theory are recalled. Characteristics of fourth order 2-PPL-systems are identified using a reciprocal-screw-system based approach in Sect. 3. The conditions for sub-6th order 3-PPL-systems are obtained by using the intersection of screw systems in Sect. 4. Finally, conclusions are drawn.

2 Fundamentals of Screw Theory

This section reviews the fundamentals of screw theory (see [2, 11, 14–17] for example) which will be used in deriving the conditions for sub-6th order 3-PPL-system in Sects. 3 and 4.

2.1 A Screw System and Its Reciprocal Screw System

A screw system $\mathscr{S}$ of order n ($0 \leq n \leq 6$) comprises all the screws that are linearly dependent on n given linearly independent screws. In most cases of design and analysis of PMs, a basis of a screw system can be represented by a set of independent $\$_\infty$ and $\$_0$. Here $\$_\infty$ and $\$_0$ denote screws of ∞- and 0- pitch respectively.

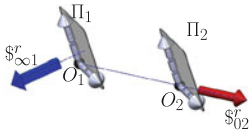
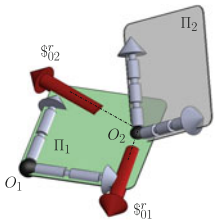
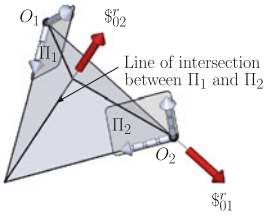
For example, column 3 in Table 1 shows three 4-$\$_0$-systems—fourth order 2-PPL-systems [11], each of which is composed of 2 PPLs. A planar pencil of lines (PPL) is a 2-$\$_0$-system which can be represented by a set of two basis $\$_0$ represented with dashed arrows with their axes meeting at one point. For convenience, the centre of PPL i is denoted by O_i, and the plane of PPL i defined by the lines is denoted by Π_i.

The reciprocal screw system, $\mathscr{S}^\perp$, of a given screw system, $\mathscr{S}$, can be determined using the following conditions.

- The axis of a $\$^r_\infty$ is perpendicular to all the axes of $\$_0$.
- The axis of a $\$^r_0$ is coplanar with the axis of each $\$_0$ and perpendicular to the directions of all the $\$_\infty$.

Table 1 shows the reciprocal screw systems of three 4-$\$_0$-systems—fourth order 2-PPL-systems [11]. In the Case 4a 2-PPL-system, the planes of the two PPLs, Π_1 and Π_2, are parallel, and a basis of its reciprocal system is composed of $\$^r_{\infty 1}$ and $\$^r_{02}$. The direction of the $\$^r_{\infty 1}$ is perpendicular to Π_1 and Π_2, while the axis of $\$^r_{02}$ passes through the centres, O_1 and O_2, of the PPLs. In the Case 4b 2-PPL-system, the centre, O_2, of PPL 2 is on the plane, Π_1, of PPL 1, a basis of its reciprocal system is composed of $\$^r_{01}$ and $\$^r_{02}$ with their axes located on Π_1 and passing through O_2. In the Case 4c 2-PPL-system, a basis of its reciprocal system is composed of $\$^r_{01}$ along line O_1O_2 and $\$^r_{02}$ along the line of intersection between planes Π_1 and Π_2.

Table 1 2-PPL-systems of order 4 and their reciprocal screw systems

Order	Class	Figure	Condition	Reciprocal screw system
4	4a		$\Pi_1 \parallel \Pi_2$	1-$\$^r_\infty$-1-$\$^r_0$-system
	4b		O_2 is on Π_1	2-$\$^r_0$-system
	4c		—	2-$\$^r_0$-system

2.2 Sum of Two Screw Systems

Let $\mathscr{S}_1$ and $\mathscr{S}_2$ denote two screw systems. The sum, $\mathscr{S}$, of $\mathscr{S}_1$ and $\mathscr{S}_2$ is

$$\mathscr{S} = \mathscr{S}_1 + \mathscr{S}_2 \tag{2}$$

The order of $\mathscr{S}$ can be determined using

$$dim(\mathscr{S}) = dim(\mathscr{S}_1) + dim(\mathscr{S}_2) - dim(\mathscr{S}_1 \cap \mathscr{S}_2) \tag{3}$$

Equation (3) will be used to derive the conditions for sub-6th order 3-PPL-systems in Sect. 4.

3 Characteristics of Fourth Order Screw Systems Composed of Two Planar Pencils of Lines

A 3-PPL-system can be regarded as the sum of one 1-PPL-system and one 2-PPL-system. In order to derive the conditions for sub-6th order 3-PPL-systems, the characteristics of the fourth order 2-PPL-systems [11] (Table 1) will be revealed using

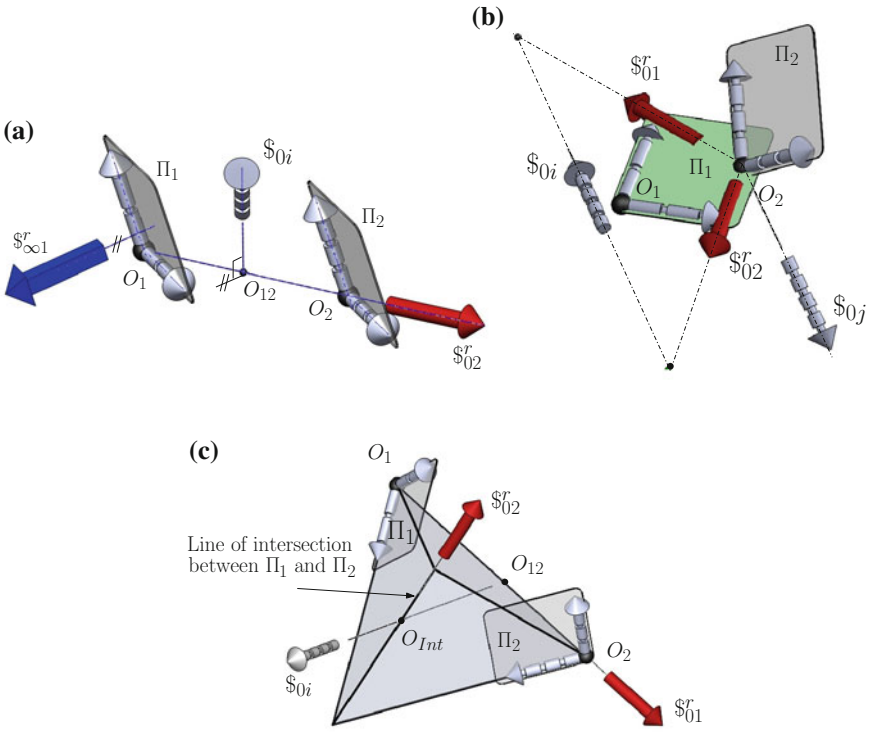


Fig. 2 Characteristics of fourth order 2-PPL-systems. **a** Case 4a. **b** Case 4b. **c** Case 4c

a reciprocal-screw-system based approach rather than the algebraic approach in the literature. Unlike in [11] where all the PPLs within a Case 4c 2-PPL-system was revealed, all the $\$_0$ within any of the three fourth order 2-PPL-systems will be revealed in this section.[1]

The reciprocal-screw-system based approach for revealing the characteristics of a screw system is based on Eq. (1), which has been used in the type synthesis of PMs (see [15] for example), in order to avoid any algebraic derivation.

According to Sect. 2.1, a basis of the reciprocal system of a Case 4c 2-PPL-system (Table 1) is composed of $\$^r_{01}$ along line O_1O_2 and $\$^r_{02}$ along the line of intersection between planes Π_1 and Π_2. Reapplying the reciprocity conditions of screws [Eq. (1)], we obtain that a Case 4c 2-PPL-system includes all the $\$_0$ (lines) intersecting with both the axes of $\$^r_{01}$ and $\$^r_{02}$ simultaneously. In other words, the Case 4c 2-PPL-system includes all the $\$_0$ (lines) defined by any point O_{Int} on the line of intersection between the planes, Π_1 and Π_2, of the two PPLs and any point O_{12} on line O_1O_2 (Fig. 2c). Similarly, a Case 4a 2-PPL-system includes all the lines

[1] Screws of non-zero pitch within a fourth order 2-PPL-system are omitted since they are irrelevant to the conditions for sub-6th order 3-PPL-systems.

($\$_{0i}$) that are parallel to planes, Π_1 and Π_2, and pass through a point, such as O_{12}, on line O_1O_2 (Fig. 2a). In the Case 4a 2-PPL-system, any line $\$_{0i}$ if not on the Π_1 and Π_2 planes, and the two planes, Π_1 and Π_2, meet at infinity. A Case 4b 2-PPL-system (Fig. 2b) includes all the lines that are located on the plane Π_1 of PPL 1 ($\$_{0i}$) and/or pass through the center O_2 of PPL 2 ($\$_{0j}$).

The above characteristics will be used to derive the conditions for sub-6th order 3-PPL-systems in Sect. 4.

4 Conditions for Sub-6th Order 3-PPL-System

Let a 3-PPL-system and its compositional 2-PPL-system, which is the one that has the minimum order among the three 2-PPL-systems within the 3-PPL-system, and 1-PPL-system be denoted by $\mathscr{S}$, $\mathscr{S}_1$ and $\mathscr{S}_2$ respectively. The two PPLs of $\mathscr{S}_1$ are denoted by PPL 1 and PPL 2, and the PPL of $\mathscr{S}_2$ is denoted by PPL 3.

From Eq. (3), we obtain that a 3-PPL-system is a sub-6th order 3-PPL-system if (a) a PPL degenerates into a line (for hybrid manipulators), (b) $dim(\mathscr{S}_1) \leq 3$, and/or (c) $dim(\mathscr{S}_1 \cap \mathscr{S}_2) \geq 1$. Since cases involving a degenerated PPL (Case (a)) or 2-PPL-systems order 2 and 3 (Case (b)) are trivial and well documented (see [11] for example), this section will focus on the conditions for sub-6th order 3-PPL-systems composed of a fourth order 2-PPL-system (Table 1) and 1-PPL-system (Case (c)).

In a sub-6th order 3-PPL-system involving a fourth order 2-PPL-system, we have $dim(\mathscr{S}_1 \cap \mathscr{S}_2) \geq 1$. This requires that the intersection of the 2-PPL-system and 1-PPL-system includes at least one $\$_0$.

Based on the characteristics (Fig. 2c) of Case 4c 2-PPL-system, we can obtain that the condition for a sub-6th order 3-PPL-system is that line $O_{12}O_{Int}$ ($\$_{0i}$) of the 2-PPL-system is located on the plane, Π_3, of PPL 3 and passes through O_3 (Fig. 3d). Since line $O_{12}O_{Int}$ is the intersection of the plane, Π_3, of PPL 3 and plane $O_1O_2O_3$, the geometric characteristics of the sub-6th order 3-PPL system is that the planes, Π_i ($i = 1$, 2 and 3), of these three PPLs and the plane $O_1O_2O_3$ defined by the centres of these PPLs have a common point O_{Int} (Fig. 3). The above conclusion is also true for 3-PPL-systems involving classes 4a and 4b 2-PPL-systems. In a 3-PPL-system involving Case 4a 2-PPL-system (Fig. 2a), the condition for sub-6th order 3-PPL-system is that line $\$_{0i}$ of the 2-PPL-system is located on the plane of PPL 3 and passes through O_3 (Fig. 3a). Since line $\$_{0i}$ is the intersection of the plane, Π_3, of PPL 3 and plane $O_1O_2O_3$, the geometric characteristics of the sub-6th order 3-PPL system is that the planes, Π_i ($i = 1$, 2 and 3), of these three PPLs and the plane $O_1O_2O_3$ defined by the centres of these PPLs have a common point at infinity (Fig. 3a). In a 3-PPL-system involving a Case 4b 2-PPL-system (Fig. 2b), the condition for sub-6th order 3-PPL-system is that line $\$_{0i}$ or $\$_{0j}$ of the 2-PPL-system is located on the plane, Π_3, of PPL 3 and passes through O_3 (Fig. 3a). In the former case (Fig. 3b), both O_2 and O_3 are located on plane Π_1. This leads to that planes Π_1 and $O_1O_2O_3$ coincide. The geometric characteristics of the sub-6th order 3-PPL system is that the planes Π_i (i =1, 2 and 3) and $O_1O_2O_3$, among which at most three planes are distinct,

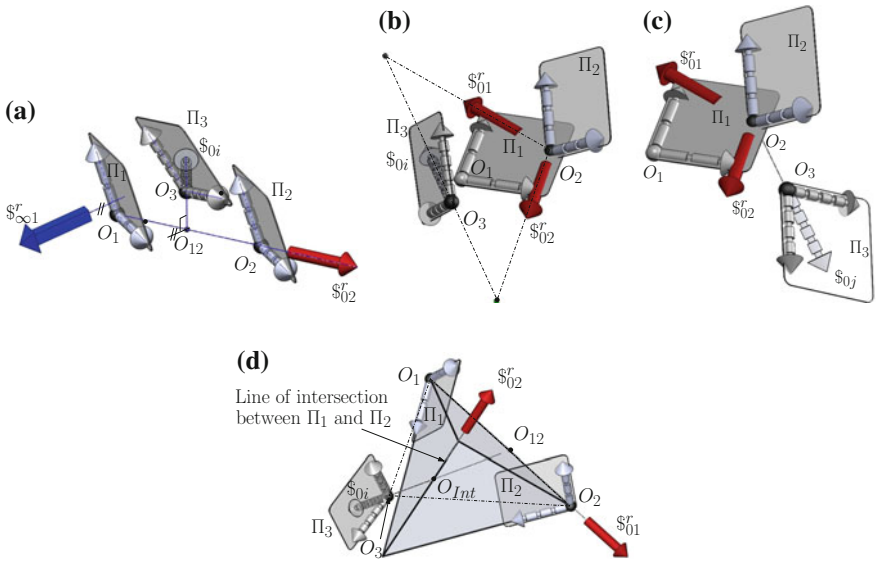


Fig. 3 Condition for sub-6th order 3-PPL-systems. **a** Case 4a. **b** Case 4b(1). **c** Case 4b(2). **d** Case 4c

have a common point. In the latter case (Fig. 3c), point O_2 is located on planes Π_1 and Π_3. The geometric characteristics of the sub-6th order 3-PPL system is that the planes Π_i ($i = 1, 2$ and 3) and $O_1O_2O_3$ meet at point O_2.

Unlike the approaches in the literature (see [1–11] for example), the approach presented in this paper is a straightforward geometric approach which needs fewer derivations and requires no knowledge about properties of linear complex or the Ceva theorem. The 3-PPL-system shown in Fig. 3a is a class [4a, 4c, 4c] fifth-order 3-PPL-system, which was not identified in [11]. Following the notation in [11], the class of a fifth order 3-PPL-system is represented by the orders of its three 2-PPL-systems here.

5 Conclusions

A simplified approach has been proposed for deriving the conditions for sub-6th order 3-PPL-systems which requires only the basic manipulations on screw systems, especially the reciprocal screw systems. The reciprocal-screw-system based approach for revealing the characteristics of the fourth order 2-PPL-systems has been found to be efficient. This paper also contributes to the classification of 3-PPL-systems. The results may help readers better understand the geometric characteristics of the singular configurations for a number of 3-legged parallel manipulators and hybrid manipulators and design parallel/hybrid manipulators with singularity considerations.

The method proposed in this paper is being extended to the singularity analysis of other classes of parallel manipulators and hybrid manipulators.

Acknowledgments The first author would like to acknowledge the financial support of Engineering and Physical Sciences Research Council of the UK (EPSRC) under grant Nos. EP/I016333/1 and EP/K018345/1. The second author would like to acknowledge the support of National Science Foundation China under Grant No. 51375058. The authors also acknowledge the support from the Overseas Famous Scholar Program Sponsored by Chinese Ministry of Education, China. Thanks to Mr Ruiming Li from Beijing Jiaotong University, China for creating the CAD models in Fig. 1.

References

1. Merlet, J.-P.: Parallel Robots, 2nd edn. Springer, Dordrecht (2006)
2. McCarthy, J.M.: Geometric Design of Linkages, 2nd edn. Springer, Heidelberg (2011)
3. Ebert-Uphoff, I., Lee, J.-K., Lipkin, H.: Characteristic tetrahedron of wrench singularities for parallel manipulators with three legs. IMechE, J. Mech. Eng. Sci. **216**(C1), 81–93 (2002)
4. Kong, X., Gosselin, C.M.: Uncertainty singularity analysis of parallel manipulators based on the instability analysis of structures. Int. J. Robot. Res. **20**(11), 847–856 (2001)
5. Yang, G., Chen, I.-M., Lin, W., Angeles, J.: Singularity analysis of three-legged parallel robots based on passive joint velocities. Trans. Robot. Autom. **17**(4), 413–422 (2001)
6. Downing, D.M., Samuel, A.E., Hunt, K.H.: Identification of special configurations of the octahedral manipulator using the pure condition. Int. J. Robot. Res. **21**(2), 147–159 (2002)
7. Huang, Z., Chen, L.H., Li, Y.W.: The singularity principle and property of Stewart parallel manipulator. J. Robot. Syst. **20**(4), 163–176 (2003)
8. Di Gregorio, R.: Forward problem singularities in parallel manipulators which generate SX-YS-ZS structures. Mech. Mach. Theory **40**(5), 600–612 (2005)
9. Ben-Horin, P., Shoham, M.: Singularity condition of six degree-of-freedom three-legged parallel robots based on Grassmann-Cayley algebra. IEEE Trans. Robot. **22**(4), 577–590 (2006)
10. Pendar, H., Mahnama, M., Zohoor, H.: Singularity analysis of parallel manipulators using constraint plane method. Mech. Mach. Theory **48**(1), 33–43 (2011)
11. Kong, X., Johnson, A.: Classification of screw systems composed of three planar pencils of lines for singularity analysis of parallel mechanisms. ASME J. Mech. Robot. **6**(2), 021008 (2014)
12. Kong, X., Yang, T.: Formulation of dynamic equations for hybrid robots using a component approach. In: Proceedings of the 4th Chinese National Youth Conference on Robotics, pp. 1–5, China (in Chinese) (1992)
13. Hu, B., Lu, Y., Yu, J. J., Zhuang, S.: Analyses of inverse kinematics, statics and workspace of a novel 3RPS-3SPR serial-parallel manipulator. Open Mech. Eng. J. **6**(Suppl1-M5), 65–72 (2012)
14. Davidson, J.K., Hunt, K.H.: Robots and Screw Theory: Applications of Kinematics and Statics to Robotics. Oxford University Press, New York (2004)
15. Kong, X., Gosselin, C.: Type Synthesis of Parallel Mechanisms. Springer, New York (2007)
16. Dai, J.S.: Finite displacement screw operators with embedded Chasles' motion. ASME J. Mech. Robot. **4**(4), 041002 (2012)
17. Dai, J.S.: An historical view of the theoretical development of rigid body displacements from Rodrigues parameters to the finite twist. Mech. Mach. Theor. **41**(1), 41–52 (2006)

Automatic Optimal Biped Walking as a Mixed-Integer Quadratic Program

Aurelien Ibanez, Philippe Bidaud and Vincent Padois

Abstract This chapter proposes an original Model Predictive Control approach to the walking control for humanoid robots, which allows to generate stable walking motions without the prior definition of footsteps positions and instants. Both the instant and amplitude of the changes in the supporting surface are part of the walking motion generation problem, and are described by a set of highly-constrained integer and real variables. Combined with the center of mass trajectory of the robot, this description leads to the formulation of a Mixed-Integer Quadratic Program in a Model Predictive Control framework aiming at reaching high-level objectives, such as velocity tracking and tip-over risk minimization. The contribution of this approach is illustrated by the simulation of two scenarii, demonstrating the validity of the steps and trajectories computed in push-recovery and walking velocity tracking cases.

Keywords Biped walking · Balance control · Hybrid systems · Footsteps planning · Push recovery · Mixed-integer quadratic programming

A. Ibanez (✉) · P. Bidaud · V. Padois
Institut des Systèmes Intelligents et de Robotique, Sorbonne Universités, UPMC Univ Paris 06, UMR 7222, 75005 Paris, France
e-mail: ibanez@isir.upmc.fr

A. Ibanez · P. Bidaud · V. Padois
Institut des Systèmes Intelligents et de Robotique, CNRS, UMR 7222, 75005 Paris, France

P. Bidaud
ONERA, 91123 Palaiseau, France
e-mail: bidaud@isir.upmc.fr

V. Padois
e-mail: padois@isir.upmc.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_52,

1 Introduction

The works introduced in this chapter address the control problem of safe biped walking, aiming at the predictive generation of stable motions without the use of prior gait strategy, such as footsteps positions or triggers, in response to high-level objectives and large disturbances.

Maintaining postural balance under a large physical disturbance may go through several types of strategies, ranging from body torquing and motion of the center of mass (CoM) to the adaptation of the base of support. It is clear that changes in the base of support, either by stepping or grasping an object, have the potential to provide a greater degree of stabilization than solely using inertial forces in a fixed-support configuration [4]. These changes might be necessary when the system undergoes strong physical disturbances or when required by the activity. If so, the best feet placement is a question which can find answers in the capture of the system's dynamics and kinematics with reduced models, as found in approaches such as Capture Region [5] or using Model Predictive Control [2]. Although the Capture Region method provides means to determine which strategy to adopt and when, the end of step and ongoing physical activity could potentially affect the system's ability to execute changes of support. A predictive approach may allow to capture such effects and act accordingly.

This chapter proposes to describe the contact state of the system using a highly constrained, mixed-integer set of variables. The resulting MPC formulation of the problem takes the form of a computationally efficient Mixed-Integer Quadratic Program (MIQP), which allows to preview and compute optimal changes of the supporting surface (i.e. feet positions over time) and CoM trajectory. Contribution of this approach is that walking motions are generated as a whole as the result of a quadratic problem, without the use of prior low-level inputs such as footsteps positions and instants, and relatively to an ongoing walking activity. This activity is simply specified as two high-level objectives, being balance and CoM tracking.

The method is introduced as follows. First, an original description of the contact state of a humanoid is introduced in order to bring a linear biped model subject to constraints related to the walking motion. This model is then involved in a MPC framework, leading to the definition of a MIQP. Simple results are finally presented to demonstrate the validity and contribution of this approach, in push-recovery and walking cases.

2 Linear Mixed-Integer Biped Model

The general approach of the method presented in this chapter consists in finding admissible changes in the supporting surface, in order to ensure safe balance while tracking a desired behavior of CoM of the system. The balance state of the robot in walking motion can be captured by relating its CoM dynamics to its contact

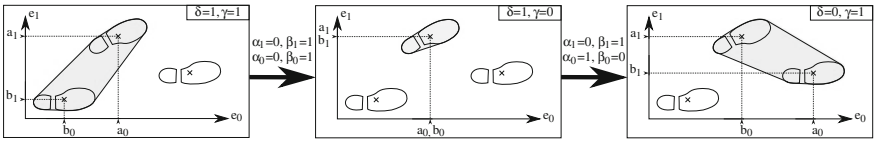


Fig. 1 Overview of the real-valued variables a and b, along with the binary variables δ and γ: a and b bound the position of the feet in contact, γ differentiates single and double support phases, and δ restrains the evolution of bounds a, b during transitions from double to single support

state. This section therefore aims at describing the robot state with a set of variables representing both its center of mass behavior and contact state.

2.1 Preliminaries: Mixed-Integer Contact State Description

In walking motions without sliding, changes in contact state are discrete and thus subject to several constraints, as illustrated in Fig. 1. The set of variables capturing the contact state of the robot should therefore be chosen adequately, in order to be able to express these constraints in a desirable form with a view to future optimization; linear constraints are generally preferred as feasibility is easily maintained in such problems.

This section proposes a choice of linearly-constrained descriptors of the contact state. The supporting surface is characterized by its bounding box: this box is described, in the horizontal plane of the ground, by real-valued variables $(\boldsymbol{a}, \boldsymbol{b}) \in \mathbb{R}^2 \times \mathbb{R}^2$ defined as the upper and lower bounds, respectively, of the position in the two horizontal directions of centers of the feet in contact,[1] as depicted in Fig. 1.

Changes of these variables are subject to several constraints, called *shape constraints*, detailed in Appendix A. Shape constraints mainly enforce that variables $(\boldsymbol{a}, \boldsymbol{b})$ cannot change continuously: binary variables $(\boldsymbol{\alpha}, \boldsymbol{\beta}) \in \{0, 1\}^2 \times \{0, 1\}^2$ are introduced as rising and falling edges of $\boldsymbol{a}$ and $\boldsymbol{b}$ respectively, in order to define constraints restraining the evolution of $(\boldsymbol{a}, \boldsymbol{b})$ to discrete changes solely.

Because variables $(\boldsymbol{a}, \boldsymbol{b})$ are implicitly related to feet positions, additional constraints, called *admissibility constraints*, must be brought. As illustrated in Fig. 1 and detailed in Appendix B, binary variables $(\gamma, \delta) \in \{0, 1\} \times \{0, 1\}$ are introduced to express such constraints, differentiating single support phases from double support ones and relating feet configurations to bounds $(\boldsymbol{a}, \boldsymbol{b})$.

The resulting set $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \delta, \gamma)$ is therefore a linearly-constrained, redundant description of the contact state of the robot. The following section presents a complete biped model relating contact state to CoM dynamics in order to capture the balance state of the system.

[1] This description implies that the feet contact and lift-off the ground parallel to it, and are rigid bodies.

In the rest of this chapter, time is sampled at discrete control instants t_i, and notation v_j for function v of time t denotes the value $v(t_j)$, and $v_{k|j}$ the value $v(t_j)$ estimated from control time t_k.

2.2 Linear Biped Model

The state $\boldsymbol{\xi}$ of the robot in walking motion is described through its contact state $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \delta, \gamma)$ and CoM horizontal dynamics $(\boldsymbol{c}, \dot{\boldsymbol{c}}, \ddot{\boldsymbol{c}})$, where $\boldsymbol{c} \in \mathbb{R}^3$ is the position of the CoM in the world frame.

The major contribution of the description $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \delta, \gamma)$ of the contact state of the robot, introduced in Sect. 2.1, is that it can be shown that shape and admissibility constraints (15)–(23), acting on these state variables, can be put in linear form (equalities and inequalities) with respect to $\boldsymbol{\xi}$ in discrete time.[2] These constraints (15)–(23) can be written in the form

$$\forall k \in \mathbb{N}, \quad \mathbf{A}_{c,l}\boldsymbol{\xi}_k + \mathbf{A}_{c,r}\boldsymbol{\xi}_{k+1} \preceq \boldsymbol{f}_c, \tag{1}$$

where $\mathbf{A}_{c,l}$ and $\mathbf{A}_{c,r}$ are $n_c \times 19$ matrices, and $\boldsymbol{f}_c$ a vector in $\mathbb{R}^{n_c}$.

However, validity of the contact state with respect to the system dynamics requires that the center of pressure lies within the convex hull of the supporting surface. In these works, the position of the CoP on the ground is approximated as in the ZMP Preview Control framework [3], i.e. neglecting rotational effects and thus only considering inertial effects resulting from the dynamics of the center of mass of the system.[3] The position $\boldsymbol{p} \in \mathbb{R}^2$ of the CoP on the ground can hence be derived from the CoM dynamics as, with g the gravity acceleration, $\boldsymbol{p} = \boldsymbol{h} - \frac{\boldsymbol{c}\cdot\boldsymbol{e}_2}{g}\ddot{\boldsymbol{h}}$, where $\boldsymbol{e}_2$ is the ascendant vertical direction and $\boldsymbol{h} = \boldsymbol{c} - (\boldsymbol{c}\cdot\boldsymbol{e}_2)\boldsymbol{e}_2$ the horizontal position of the CoM. We recall that validity of the model imposes that $\boldsymbol{p}$ lies within the convex hull of the supporting surface. Although this inclusion constraint is linear with respect to the horizontal dynamics of the CoM, the fact that changes in the supporting surface are reflected by changes in state variables $(\boldsymbol{a}, \boldsymbol{b})$ brings quadratic terms. Nevertheless, as shown in [2], overestimating the convex hull of this surface by its bounding box in forward and lateral directions brings the definition of a set of linear inequality constraints with respect to the state $\boldsymbol{\xi}$ of the system. Under this overestimation, the CoP constraints write

$$\forall k \in \mathbb{N}, \quad \mathbf{A}_p\boldsymbol{\xi}_k \preceq \boldsymbol{f}_p, \tag{2}$$

[2] Under the hypothesis that there exists upper bounds for the difference $\|a - b\|_1$ and for the variations of a and b between two consecutive instants. This hypothesis is always verified in the case of a biped robot which is subject to geometrical constraints.

[3] Moreover, vertical dynamics of the center of mass are also neglected and this hypothesis needs to be enforced in the whole-body control of the robot.

where $\mathbf{A}_p$ is a $n_p \times 19$ matrix, and $\boldsymbol{f}_p$ a vector in $\mathbb{R}^{n_p}$.
The biped model finally writes at time t_k

$$\forall k \in \mathbb{N}, \quad \boldsymbol{\xi}_k = \begin{bmatrix} \boldsymbol{a}_k & \boldsymbol{b}_k & \boldsymbol{\alpha}_k & \boldsymbol{\beta}_k & \delta_k & \gamma_k & \boldsymbol{c}_k & \dot{\boldsymbol{c}}_k & \ddot{\boldsymbol{c}}_k \end{bmatrix}^T$$

$$\text{s.t.} \begin{cases} \mathbf{A}_{c,r}\boldsymbol{\xi}_k \preceq \boldsymbol{f}_c - \mathbf{A}_{c,l}\boldsymbol{\xi}_{k-1}, & \mathbf{A}_p\boldsymbol{\xi}_k \preceq \boldsymbol{f}_p, \\ (\boldsymbol{\alpha}_k,\ \boldsymbol{\beta}_k) \in \{0,1\}^2 \times \{0,1\}^2, & (\delta_k,\ \gamma_k) \in \{0,1\} \times \{0,1\}, \\ (\boldsymbol{a}_k,\ \boldsymbol{b}_k) \in \mathbb{R}^2 \times \mathbb{R}^2, & (\boldsymbol{c}_k,\ \dot{\boldsymbol{c}}_k,\ \ddot{\boldsymbol{c}}_k) \in \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3. \end{cases} \tag{3}$$

2.3 High-Level Walking Motion Constraints

A selection of additional high-level constraints, related to the walking motion in itself and limitations of the robot, are introduced in the problem.

Maximal leg span and step length can be easily put in the form of linear inequality constraints with respect to the system state $\boldsymbol{\xi}$. Additionally, minimum durations of single and double support phases can be enforced, in the form of a set of linear inequality constraints, in order to avoid the generation of small successive steps. Last, the ratio between step size and single support duration can be bounded by a maximal swinging foot velocity implemented as a set of linear inequality constraints, constraints that will hinder solutions requiring an excessively fast movement of the swinging foot in the transition between two distinct double support phases.

Let $\bar{v}$ denote the maximal swinging foot velocity, $\bar{s}$ the maximal step length and $\bar{t}$ the largest of minimum durations of single and double support phases, these high-level constraints write

$$\forall k \in \mathbb{N} \quad \sum_{t_{k+j} - t_k \leq \max(\bar{s}/\bar{v}, \bar{t})} \mathbf{A}_{w,j}\boldsymbol{\xi}_{k+j} \preceq \boldsymbol{f}_w, \quad j \in \mathbb{N}, \tag{4}$$

where $\mathbf{A}_{w,j}$ is a $n_w \times 19$ matrix, and $\boldsymbol{f}_w$ a vector in $\mathbb{R}^{n_w}$ depending on a history of states $\boldsymbol{\xi}$ prior to t_k. It can be shown that $\boldsymbol{f}_w$ can be written in the form

$$\boldsymbol{f}_w = \boldsymbol{f}_{w_0} + \sum_{t_k - t_{k-j} < \max(\bar{s}/\bar{v}, \bar{t})} \mathbf{B}_{w,j}\boldsymbol{\xi}_{k-j}, \quad j \in \mathbb{N}^*. \tag{5}$$

3 Quadratic Mixed-Integer Predictive Control

The control problem of walking motion generation is formulated as follows: find, over a preview horizon, admissible changes in the support surface (foot steps position and instants) and optimal center of mass trajectory (balance adjustments) that minimize, among other secondary objectives, a tracking error with a desired center of mass path

and velocity (stand still, walk forward, etc.) and maximize the distance of the CoP to the bounds of the supporting surface (minimum tip-over risk).

3.1 Quadratic Model Predictive Control

The aim of the balance MPC problem is to find at each control step t_k, in addition to admissible changes in the supporting surface, a horizon of CoM trajectory that minimizes both a tracking error to a reference trajectory (desired displacements of the system as a whole) and a tip-over risk. Due to the hypotheses of the ZMP-based balance criterion, a horizontal trajectory solely is of interest in such a problem.

A future trajectory $\left(\boldsymbol{h}_{k|k+j},\ \dot{\boldsymbol{h}}_{k|k+j},\ \ddot{\boldsymbol{h}}_{k|k+j}\right)_{j=1..N}$ of the center of mass, previewed from state $\boldsymbol{\xi}_k$ at control step t_k over N time steps, can be entirely defined from a horizon of N piecewise constant CoM jerks $\boldsymbol{u} = \partial^3 \boldsymbol{h}/\partial t^3$, using a straightforward discrete integration scheme. With $\hat{\boldsymbol{h}} = \left[\boldsymbol{h}\ \dot{\boldsymbol{h}}\ \ddot{\boldsymbol{h}}\right]^T$, the CoM preview writes

$$\forall\,(k,\,j) \in \mathbb{N} \times \mathbb{N}, \quad \hat{\boldsymbol{h}}_{k|k+j+1} = \mathbf{A}_h \hat{\boldsymbol{h}}_{k|k+j} + \mathbf{B}_h \boldsymbol{u}_{k|k+j+1}, \tag{6}$$

where $\mathbf{A}_h$ and $\mathbf{B}_h$ are integration matrices. Under the assumption that the altitude of the CoM remains constant over time, a previewed state $\boldsymbol{\xi}_{k|k+j+1}$ can be linearly derived from previous state $\boldsymbol{\xi}_{k|k+j}$ given the value $\boldsymbol{\chi}_{k|k+j+1}$ of the unknown variable defined as

$$\boldsymbol{\chi} = \left[\ \boldsymbol{a}\ \ \boldsymbol{b}\ \ \boldsymbol{\alpha}\ \ \boldsymbol{\beta}\ \ \delta\ \ \gamma\ \ \boldsymbol{u}\ \right]^T. \tag{7}$$

Notation (7) and relation (6) indeed brings, with $\mathbf{Q}$ and $\mathbf{T}$ state description matrices, the linear time-invariant process

$$\forall\,(k,\,j) \in \mathbb{N} \times \mathbb{N}, \quad \boldsymbol{\xi}_{k|k+j+1} = \mathbf{Q}\boldsymbol{\xi}_{k|k+j} + \mathbf{T}\boldsymbol{\chi}_{k|k+j+1}, \tag{8}$$

which allows the preview, from the actual state $\boldsymbol{\xi}_{k|k} = \boldsymbol{\xi}_k$, of a horizon $\boldsymbol{C}_{k,N}$ of N future states $\boldsymbol{\xi}_{k|k+j}$ from a horizon $\boldsymbol{X}_{k,N}$ of N future inputs $\boldsymbol{\chi}_{k|k+j}$. Indeed, Eq. (8) can be spread over N steps and bring

$$\boldsymbol{C}_{k,N} = \begin{bmatrix} \boldsymbol{\xi}_{k|k+1} \\ \vdots \\ \boldsymbol{\xi}_{k|k+N} \end{bmatrix} = \mathbf{P}\boldsymbol{\xi}_k + \mathbf{R} \begin{bmatrix} \boldsymbol{\chi}_{k|k+1} \\ \vdots \\ \boldsymbol{\chi}_{k|k+N} \end{bmatrix} = \mathbf{P}\boldsymbol{\xi}_k + \mathbf{R}\boldsymbol{X}_{k,N}, \tag{9}$$

and thus constraints (1), (2) and (4) can be rewritten

$$\mathbf{A}\boldsymbol{X}_{k,N} \preceq \boldsymbol{f} \tag{10}$$

where $\boldsymbol{f}$ depends on a history of states $\boldsymbol{\xi}$, as introduced in (1), (4) and (5).

The minimization of tip-over risk can be expressed at the level of the center of pressure: safe balance motor control should aim at maximizing the distance between the CoP and the convex hull of the supporting surface.[4] This objective can be expressed as the cost function J_b to be minimized

$$J_{bk} = \sum_{j=1}^{N} \left\| \boldsymbol{p}_{k|k+j} - \boldsymbol{r}_{k|k+j} \right\|^2, \tag{11}$$

which measures a cumulated square distance between the CoP and the center $\boldsymbol{r}$ of the supporting surface, where $\boldsymbol{r} = (\boldsymbol{a} + \boldsymbol{b})/2$.

Walking motions generally aim at reaching a desired horizontal displacement of the system as a whole (null displacement in the case of standstill activities), that can be expressed as a reference CoM horizontal trajectory $\hat{\boldsymbol{h}}^{ref}$. This second objective can be reached through the minimization of the cost function J_w that writes

$$J_{wk} = \sum_{j=1}^{N} \left\| \mathbf{S} \left(\hat{\boldsymbol{h}}_{k|k+j} - \hat{\boldsymbol{h}}^{ref}_{k|k+j} \right) \right\|^2, \tag{12}$$

where $\mathbf{S}$ is a 6×6 weighting selection matrix, diagonal, defining whether position, velocity and/or acceleration are tracked in each of the two horizontal directions.

Secondary objectives can be added to the problem in the form of the quadratic cost q, for regulation purposes or additional input, for example in order to prefer double support phases over single support ones.

Objectives (11) and (12) are quadratic with respect to $\boldsymbol{X}_{k,N}$, and can be synthesized in the form

$$J_k = \omega_b J_{bk} + \omega_w J_{wk} + q = \boldsymbol{X}_{k,N}^T \mathbf{H} \boldsymbol{X}_{k,N} + \boldsymbol{d}^T \boldsymbol{X}_{k,N}, \tag{13}$$

where $\mathbf{H}$ is a positive definite matrix, $\boldsymbol{d}$ a vector and (ω_b, ω_w) scalar weights. The MPC problem finally writes

$$\min_{\boldsymbol{X}_{k,N}} \boldsymbol{X}_{k,N}^T \mathbf{H} \boldsymbol{X}_{k,N} + \boldsymbol{d}^T \boldsymbol{X}_{k,N}$$
$$\text{s.t.} \begin{cases} \mathbf{A} \boldsymbol{X}_{k,N} \preceq \boldsymbol{f} \\ \boldsymbol{\xi}_{k|k} = \boldsymbol{\xi}_k, \quad \boldsymbol{\xi}_{k|k+j} = \mathbf{Q} \boldsymbol{\xi}_{k|k+j-1} + \mathbf{T} \boldsymbol{\chi}_{k|k+j}, & j = 1..N \\ \left(\boldsymbol{a}_{k|k+j}, \boldsymbol{b}_{k|k+j}, \boldsymbol{u}_{k|k+j} \right) \in \mathbb{R}^2 \times \mathbb{R}^2 \times \mathbb{R}^2, & j = 1..N \\ \left(\boldsymbol{\alpha}_{k|k+j}, \boldsymbol{\beta}_{k|k+j} \right) \in \{0,1\}^2 \times \{0,1\}^2, & j = 1..N \\ \left(\delta_{k|k+j}, \gamma_{k|k+j} \right) \in \{0,1\} \times \{0,1\}, & j = 1..N \end{cases} \tag{14}$$

[4] As a tip-over case (instantaneous rotation around a point on the ground) is the result of the CoP being on the edge of the convex hull of the supporting surface.

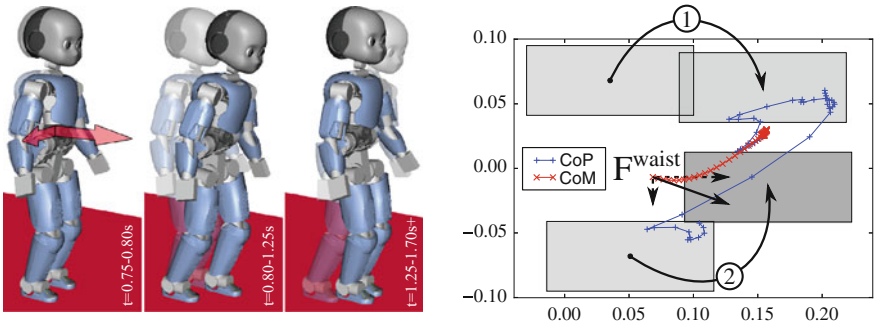


Fig. 2 Overview of the results from the MIQP predictive framework (14)—*left* optimal automatic push-recovery steps—*right* evolution of the CoP and CoM of the robot after impact $\boldsymbol{F}^{waist}$

which is a MIQP. Note that the complexity of problem (14), relatively to a standard QP, is strongly related to the number of integer variables. Binary variables ($\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, δ, γ) can be discretized with a larger time increment than real-valued variables, and the previous formulation still holds given appropriate re-sampling matrices, while reducing the size of problem (14). The only drawbacks of such an approach is that it restrains, in the preview solely, the occurrence of changes in the supporting surface to fewer potential instants.

4 Results

The validity of the mixed-integer predictive formulation (14) of the walking control problem is demonstrated through the simulation of two common biped scenarii: push recovery and forward walking. In both cases, motion of the center of mass and of the supporting contacts are automatically computed at each control step, as an optimal response to high-level CoM velocity tracking and balance objectives J_w and J_b. In both scenarii, weights ω_w and ω_b are equal.

Simulations are performed using the ARBORIS- PYTHON simulator [1] developed at ISIR, and whole-body motion from optimal outputs of the MPC (14) is ensured by an LQP-based controller [6] for an iCub [7] robot model. The system state is previewed over a horizon of 1.0 s.

The push-recovery scenario is specified as follows: the system must track a null CoM velocity $\|\dot{\boldsymbol{h}}^{ref}\| = 0\,\mathrm{m}\cdot\mathrm{s}^{-1}$ while maintaining balance, and an unknown force $\boldsymbol{F}^{waist} = [60N \;\; -20N]$ is applied to the waist of the robot during a period of 0.1 s. Outputs from (14) automatically compensate the sudden changes in the system state resulting from this external force by triggering a sequence of footsteps previewed over a future horizon. As the system reaches its desired state $\|\dot{\boldsymbol{h}}\| = 0\,\mathrm{m}\cdot\mathrm{s}^{-1}$, the optimal solution of (14) requires no further steps. An overview of the results is presented on Fig. 2.

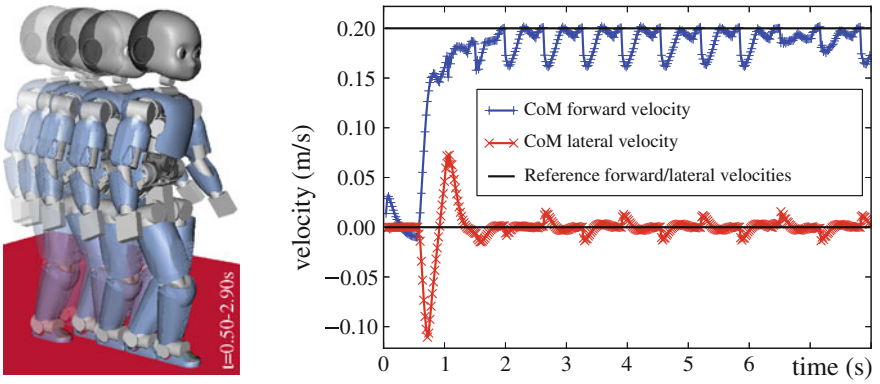


Fig. 3 Overview of the results from the MIQP predictive framework (14)—*left:* optimal automatic gait from high-level velocity tracking—*right* minimization of the CoM velocity tracking objective

In the walking scenario, a velocity $\dot{\boldsymbol{h}}^{ref} = [0.20\,\mathrm{m}\cdot\mathrm{s}^{-1}\ 0\,\mathrm{m}\cdot\mathrm{s}^{-1}]$ is set as a reference. This reference states that the system must move forward while minimizing lateral velocity of its CoM. Results shown on Fig. 3 shows that the generated gait effectively tracks the desired velocity in both lateral and forward directions. An interesting result is that optimal footsteps of such a problem are aligned in order to minimize velocity of the CoM in the lateral direction. Note that collision avoidance constraints and objectives might challenge this result.

5 Conclusion

The Model Predictive Control framework introduced in this chapter allows the generation of optimal walking motions for biped robots, able to reach high-level objectives such as velocity tracking and tip-over risk minimization, without the use of prior heuristics such as predefined feet positions or footsteps triggers.

The contact state of the robot is described through a mixed set of variables involving integers, conjointly optimized with the CoM horizontal trajectory over a preview horizon in order to maintain balance while tracking high-level objectives.

The validity of the resulting MIQP formulation is demonstrated in push-recovery and walking speed tracking scenarii, exhibiting automatic footsteps computation and trigger.

Future works will investigate different behaviors in more complex cases, while providing more insights on the computational cost of such problems.

Appendix A

Variables $\boldsymbol{a}$ and $\boldsymbol{b}$ are piecewise constant functions of time[5]; binary variables $(\boldsymbol{\alpha}, \boldsymbol{\beta}) \in \{0, 1\}^2 \times \{0, 1\}^2$ are therefore introduced as rising and falling edges of $\boldsymbol{a}$ and $\boldsymbol{b}$ respectively, defined as follows

$$\forall i \in \{0, 1\}, \forall t \in \mathbb{R}, \quad \begin{cases} \alpha_i(t) = 1 & \Rightarrow \quad \alpha_i(t^-) = \alpha_i(t^+) = 0 \\ \alpha_i(t) = 0 & \Rightarrow \quad a_i(t^-) = a_i(t^+) = a_i(t) \end{cases} \tag{15}$$

and similarly for the pair $(\boldsymbol{\beta}, \boldsymbol{b})$. Variables $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ can have their value set to 1 at distinct instants solely (15), and $\boldsymbol{\alpha} = \boldsymbol{0}$ imposes constancy of $\boldsymbol{a}$ as written in (15).

Note that changes in $\boldsymbol{a}$ and $\boldsymbol{b}$ are subject to the following additional constraints

$$\forall i \in \{0, 1\}, \forall t \in \mathbb{R}, \quad b_i(t) \leq a_i(t), \tag{16}$$

$$\forall i \in \{0, 1\}, \forall t \in \mathbb{R}, \quad \alpha_i(t) + \beta_i(t) \leq 1, \tag{17}$$

$$\forall t \in \mathbb{R}, \quad \alpha_0(t) + \beta_0(t) = \alpha_1(t) + \beta_1(t). \tag{18}$$

The definition of $\boldsymbol{a}$ and $\boldsymbol{b}$ as respectively upper and lower bounds has to be enforced (16); also, as only one foot can move at a time, it imposes that, in a given direction, only one of the bounds $\boldsymbol{a}$ or $\boldsymbol{b}$ can change (17). Last, a change in a first direction for either bound $\boldsymbol{a}$ or $\boldsymbol{b}$ must allow a change in the other direction (18).

Constraints (15)–(17) define the bounding box of the support surface as restricted between two bi-dimensional, piecewise constant variables $\boldsymbol{a}$ and $\boldsymbol{b}$. However, the nature of the changes occurring in $\boldsymbol{a}$ and $\boldsymbol{b}$ imposes the consideration of additional variables and constraints to define and guarantee the admissibility of such changes.

Appendix B

As shown in Fig. 1, a phase of single support requires that $\boldsymbol{a}$ and $\boldsymbol{b}$ are equal, and phases of double support (DS) are necessarily separated by, at least, a single support phase (SS). Binary variable $\gamma \in \{0, 1\}$ is therefore introduced to capture the alternation between single and double support phase: $\gamma = 1$ describes a double support phase and $\gamma = 0$ a single support one. The single support constraint writes

$$\forall t \in \mathbb{R}, \ \gamma(t) = 0 \quad \Rightarrow \quad \boldsymbol{a}(t) = \boldsymbol{b}(t), \tag{19}$$

and the alternation ones, accounting for (18)

[5] The discrete nature of contact states in non-sliding walking motions imposes that changes in $\boldsymbol{a}$ and $\boldsymbol{b}$ can only occur at discrete events.

$$\alpha_0(t) + \beta_0(t) = 1 \quad \Rightarrow \quad \gamma(t) = \begin{cases} 0 \text{ if } & \gamma(t^-) = 1 \\ 1 \text{ if } & \gamma(t^-) = 0 \end{cases} \tag{20}$$

and

$$\alpha_0(t) + \beta_0(t) = 0 \quad \Rightarrow \quad \gamma(t) = \gamma(t^-), \tag{21}$$

which state that γ must change if and only if a change in variables $\boldsymbol{a}$ or $\boldsymbol{b}$ occurs.[6] However, these variables and constraints are not sufficient to guarantee the admissibility of changes in the bounding box described by $(\boldsymbol{a}, \boldsymbol{b})$. Indeed, as shown in Fig. 1, potential changes in $\boldsymbol{a}$ and $\boldsymbol{b}$ from a DS phase to a SS one depend on the configuration of the feet, relatively to bounds $\boldsymbol{a}$ and $\boldsymbol{b}$, in the DS phase. If one of the two feet in contact defines the upper bound $\boldsymbol{a}$ in both directions (*cf.* Fig. 1, far left), the following single support phase can only result from a change of either $\boldsymbol{a}$ or $\boldsymbol{b}$ in both directions ; symmetrically, if one of the two feet in contact defines the upper bound $\boldsymbol{a}$ in one direction and lower bound $\boldsymbol{b}$ in the other (*cf.* Fig. 1, far right), the following SS phase will result from a change in $\boldsymbol{a}$ in one direction, and $\boldsymbol{b}$ in the other. This time history constraint can be captured by the introduction of the binary variable $\delta \in \{0, 1\}$, having 1 for value in the first case and 0 in the second. The indicator of configuration δ must remain constant through each DS phase in order to keep the information for the next transition towards a SS phase; that is

$$\forall t \in \mathbb{R},\ \gamma(t) = 1 \quad \Rightarrow \quad \delta(t^+) = \delta(t), \tag{22}$$

and, as described earlier, changes in a and b are related to the values of δ. This relation writes

$$\forall t \in \mathbb{R},\ \ \delta(t) = 1 \Rightarrow \begin{cases} \alpha_0(t) = \alpha_1(t) \\ \beta_0(t) = \beta_1(t) \end{cases}, \text{ and } \delta(t) = 0 \Rightarrow \begin{cases} \alpha_0(t) = \beta_1(t) \\ \beta_0(t) = \alpha_1(t). \end{cases} \tag{23}$$

Note that this relation is bilateral : if δ is unconstrained by (22), changes in $\boldsymbol{a}$ and $\boldsymbol{b}$ define the value of δ and else, the value of δ restrains potential changes in $\boldsymbol{a}$ and $\boldsymbol{b}$.

[6] Note that variable γ is thus fully defined through variables $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, given some initial conditions.

References

1. Barthelemy, S., Salini, J., Micaelli, A.: Arboris-python. https://github.com/salini/arboris-python
2. Herdt, A., Diedam, H., Wieber, P.B., Dimitrov, D., Mombaur, K., Diehl, M.: Online walking motion generation with automatic footstep placement. Adv. Rob. **24**, 719–737 (2010)
3. Kajita, S., Kanehiro, F., Kaneko, K., Kajiwara, K., Harada, K., Yokoi, K., Hirukawa, H.: Biped walking pattern generation by using preview control of zero-moment point. In: Proceedings of the IEEE ICRA (2003)
4. Maki, B.E., Mcilroy, W.E., Fernie, G.R.: Change-in-support reactions for balance recovery. IEEE Eng. Med. Biol. Mag. **22**(2), 20–26 (2003)
5. Pratt, J., Carff, J., Drakunov, S., Goswami, A.: Capture point: A step toward humanoid push recovery. In: Proceedings of the IEEE-RAS International Conference on Humanoid Robots, pp. 200–207. IEEE (2006)
6. Salini, J., Padois, V., Bidaud, P.: Synthesis of complex humanoid whole-body behavior: a focus on sequencing and tasks transitions. In: Proceedings of the IEEE ICRA, pp. 1283–1290. IEEE (2011)
7. Sandini, G., Metta, G., Vernon, D.: The icub cognitive humanoid robot: an open-system research platform for enactive cognition. In: 50 Years of Artificial Intelligence, Lecture Notes in Computer Science, chap. 32, pp. 358–369. Springer (2007)

Mechanisms with Decoupled Freedoms Assembled from Spatial Deployable Units

Shengnan Lu, Dimiter Zlatanov, Xilun Ding, Rezia Molfino and Matteo Zoppi

Abstract The chapter presents a family of new deployable mechanisms obtained by conjoining an indefinite number of copies and variants of the same basic spatial linkage. The unit mechanism is an assembly of scissor and Sarrus linkages able to move in two or three independent directions. Unlike most deployable structures, which have one degree of freedom, the proposed devices can be deployed and compacted independently in two or three directions. Moreover, the physical boundary of the mechanism can be used to approximate and control the shape of some geometric curves. Kinematic simulations are performed to validate the proposed designs.

Keywords Deployable mechanism · Deployable unit · Decoupled dof

1 Introduction

Deployable mechanisms (DMs) based on linkages have been studied by a number of researchers over the past decades [5, 8, 9, 11]. The simple scissor mechanism has been used as a basic element in many creative designs of deployable structures,

S. Lu (✉) · D. Zlatanov · R. Molfino · M. Zoppi
University of Genoa, Genova, Italy
e-mail: lvshengnan5@gmail.com

D. Zlatanov
e-mail: zlatanov@dimec.unige.it

R. Molfino
e-mail: molfino@dimec.unige.it

M. Zoppi
e-mail: zoppi@dimec.unige.it

S. Lu · X. Ding
Beihang University, Beijing, China
e-mail: xlding@buaa.edu.cn

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_53,

[6, 12, 13]. Recently, more attention has been paid to spatial mechanisms as possible building units for the design of DMs. Chen developed a class of DMs using the Bennett, the Myard, and the Bricard linkages [1]. Ding et al. proposed a novel mast mechanism using spatial polyhedral linkages arranged along radial axes [2]. Applications of deployable mechanisms can be found in the field of aerospace structures [4, 10], and civil engineering, like the covering of swimming pools [3]. Possible future use is the design of reconfigurable fixtures for large pieces, such as the ones used in auto manufacturing.

Most proposed DMs have been with one degree of freedom (dof), this limits the range of the mechanisms' potential application. In this chapter, we present a family of unit-based deployable mechanisms with two or three dof. The basic unit is composed of both planar and spatial linkages. (The two-dof variant was introduced in [7].) Importantly, the multiple freedoms are decoupled, allowing very good control of the movement of the mechanism.

The two deployable units, with two and three dof, are introduced in Sect. 2. These are used, respectively, to construct deployable mechanisms with two decoupled degrees of freedom and different boundary outlines in Sect. 3, and a spatial deployable mobile assembly in Sect. 4.

2 The Deployable Mechanism Units

Typically, a DM is made of identical basic units. Both the 2D and 3D deployable units (DUs) we propose are derived as combinations of scissor and Sarrus linkages.

2.1 The Two-Dof Deployable Unit

The proposed deployable unit is composed of four scissor linkages and two Sarrus linkages. The two Sarrus linkages are connected only by a common link. In a Sarrus, there are two links constrained to perform relative translation. The shared link is one of these two. The only geometric constraint between the two merged linkages is that their translation directions must be different.

Figure 1 shows the schematic diagram of the DU, while its CAD model is in Fig. 2. The three translating links of the two Sarrus linkages are the ones housing joints 3 and 4, 9 and 10, and 1 and 6 (or 7 and 12), respectively. In the figures, they are indicated with vertical (parallel to Oz) bars and segments. In each Sarrus, apart from the translating links, the other four links form two hinged pairs, and each can be thought of as a scissor linkage.

During the movement of the DU, the two Sarrus branches can move independently, realizing an arbitrary planar translation of link 3–4 with respect to link 9–10. The movement of each Sarrus can be transmitted to a neighboring unit via a shared

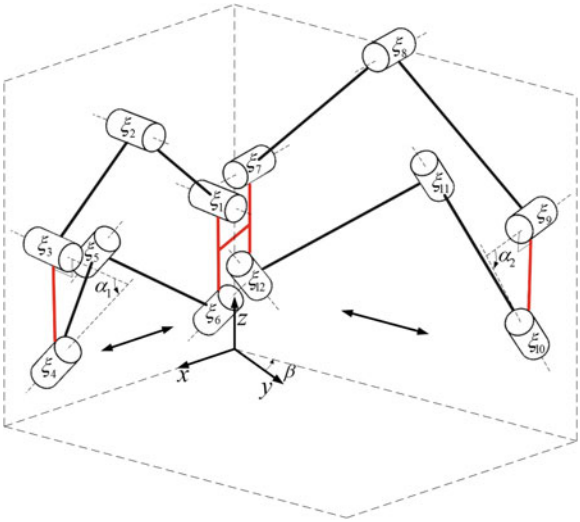


Fig. 1 A two-dof deployable unit combining scissor and Sarrus linkages

Fig. 2 CAD model of the two-dof deployable unit

scissor linkage. Thus, each branch of the DU can fold and deploy in one direction independently from the other branch, and hence with two dof, but synchronous with other attached DUs, maintaining the same two dof within the larger assembly.

2.2 The Three-Dof Deployable Unit

A natural next step is to generalize the ability to expand in two directions in a new unit linkage that can provide three decoupled translations. This can be achieved by adding another Sarrus linkage translating in a third independent direction. A deployable unit with three dof is obtained, Fig. 3. One can describe it as three extendable branches originating from a common base link.

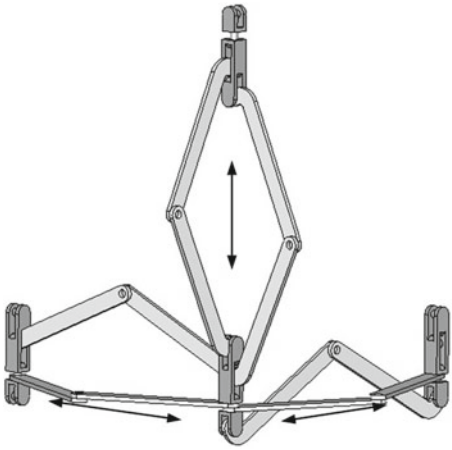


Fig. 3 CAD model of the three-dof deployable unit

3 Two-Dof Deployable Mechanisms

The described two-dof deployable units can be assembled into deployable mechanisms, which can be folded (and deployed) independently in two directions. An unlimited number of DUs can be combined while the degree of freedom of the assembly does not increase. The mechanism can have a parallelogram outline, but it can also be made to approximate certain curves, e.g. an ellipse or a hyperbola.

3.1 Deployable Parallelograms

For simplicity, in the following we assume that each Sarrus-linkage leg has two equal-length links (e.g., $l_{12} = l_{23}$ and $l_{56} = l_{78}$, with l_{ii+1} the distance between adjacent axes).

Figure 4, shows the DU used to assemble a DM in Figs. 5 and 6. The translating directions relative to D_1 of bars D_2 and D_3, are at an angle $\pi/2 + \beta$. Two DUs can be assembled by merging the corresponding links with the same translation direction. In Fig. 5, D_1^1, D_2^1, D_3^1 are the corner bars in the first unit, and D_1^2, D_2^2, D_3^2 are the corresponding parts in the second unit. By sharing the links $D_2^1 = D_3^2$ and $D_3^1 = D_2^2$, a new mechanism assembled from two deployable units is created.

By assembling more identical deployable units, novel two-dof deployable mechanisms are obtained and the degree of freedom stays constant. To have a parallelogram footprint, an even number of units is needed. In a DM with two units, no scissor linkage is needed. However, with four or more units scissor linkages are essential to transfer the motion between adjacent units. As the assembled mechanism is a highly overconstraint system, some of the links are removed during modeling when they contribute only redundant constraints. The CAD model of an 8-unit structure is shown in Fig. 6.

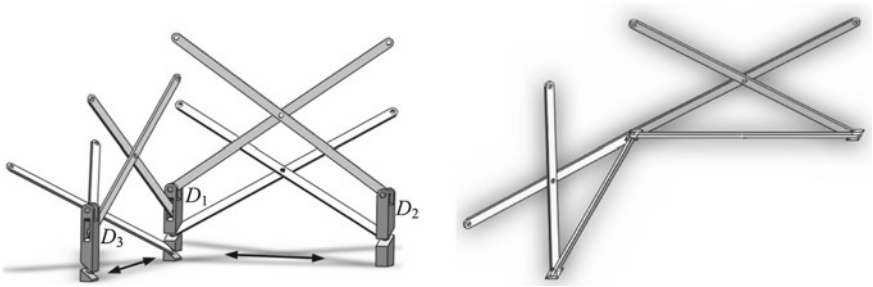


Fig. 4 The deployable unit used in the assembly

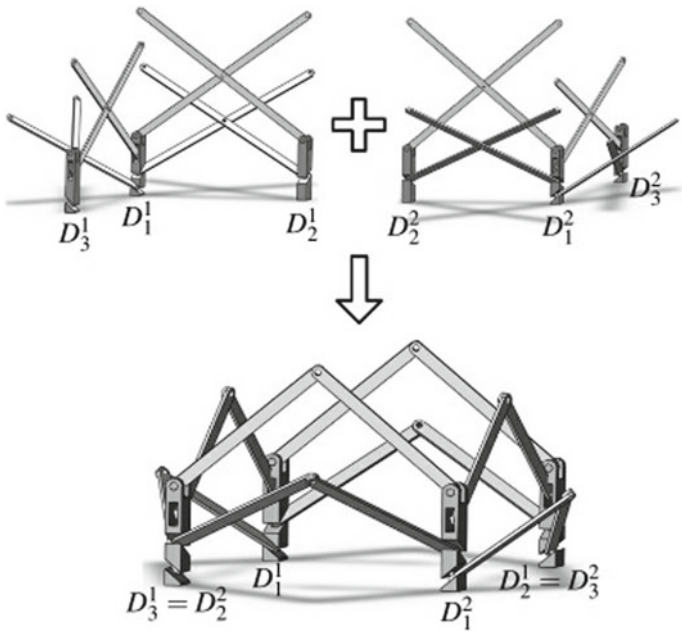


Fig. 5 Assembly process of the two-dof deployable mechanism

We can see that the geometric shape generated by the footprint of the translating bars is decided by the angle β in Fig. 1. By varying β, different parallelograms can be obtained. When $\beta = 0$, the footprint is a rectangle, Fig. 7.

3.2 A Deployable Ellipse

The shape and size of an ellipse centered at the origin, $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$, is controlled by the values of the semi-major and semi-minor axes, a and b. The two-dof DUs

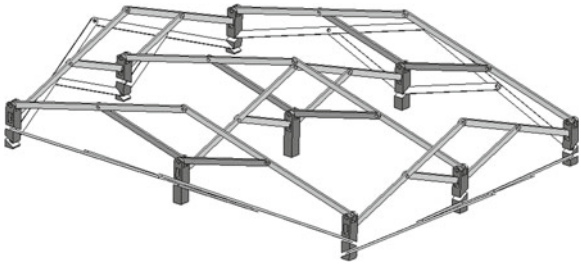


Fig. 6 Assembly with multiple units

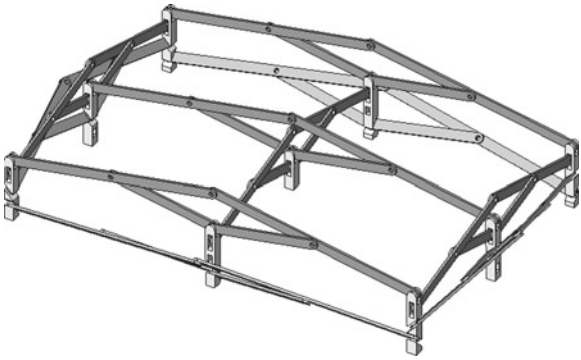


Fig. 7 A deployable mechanism with a rectangular outline

can be used to assemble a DM with an approximate elliptical or circular boundary. Moreover, the values of a and b are controlled independently by the two dof, each varying between $r_{\min}$ and $r_{\max}$. Unlike the parallelogram DM, all used DUs are not exactly identical: the ones at the periphery are smaller. A model of such a mechanism with 14 DUs has been designed and simulated. During the deployment process, the angle between each two edges of the approximating polygon varies, so some additional revolute joints are introduced into the system to solve this problem, Fig. 8. Figure 9 shows the shape change of the DM viewed from the top.

3.3 A Deployable Hyperbola

Similar designs allow to approximate and control the shape of other two-parameter curves. An example is the mechanism in Fig. 10, in which the boundary points always lie on a hyperbola, $\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1$, whose parameters a and b vary when the configuration changes. As in the elliptical case, some revolute joints are added in the simulated model. The CAD model of this mechanism is built, and the simulations are performed as shown in Fig. 10.

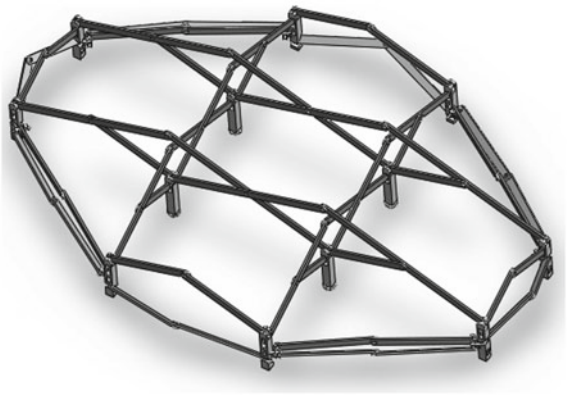


Fig. 8 A deployable ellipse

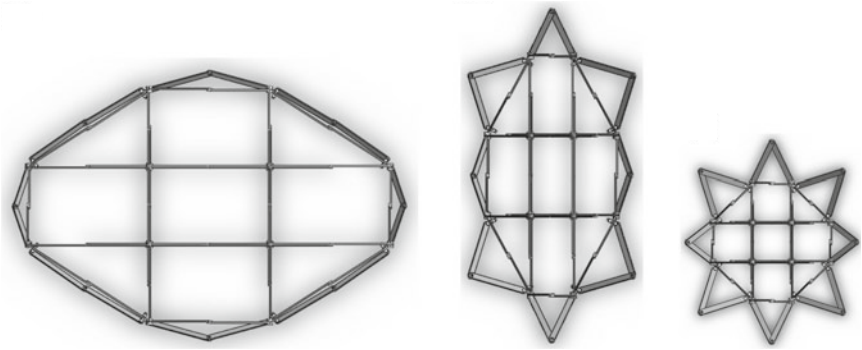


Fig. 9 Shape change of the elliptical outline

4 Three-Dof Deployable Mechanisms

Using the same assembling method, by linking the three-dof deployable units, three-dof DMs can be obtained. The mechanism can be deployed in three directions independently. An unlimited number of unit mechanisms can be assembled together while the degree of freedom of the assembly stays three. As in the planar case, the scissor linkage is used to transmit motion between units. Two simulated models of two DMs, with two and eight units, respectively, are shown in Fig. 11. Each becomes a cube when fully deployed. The deployment process of the 8-unit DM is illustrated in Fig. 12.

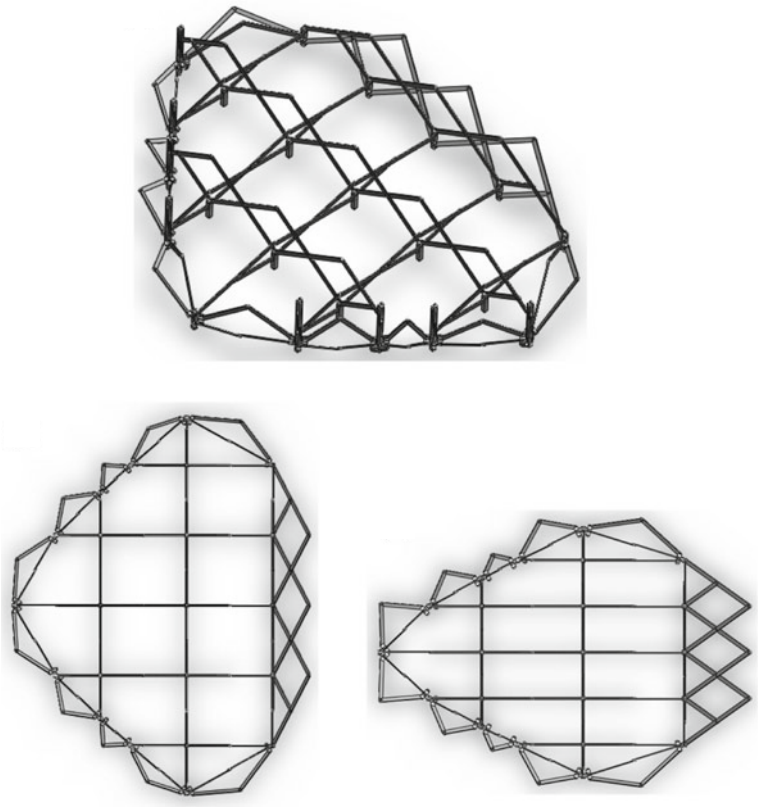


Fig. 10 A CAD model of a mechanism with a hyperbolic outline

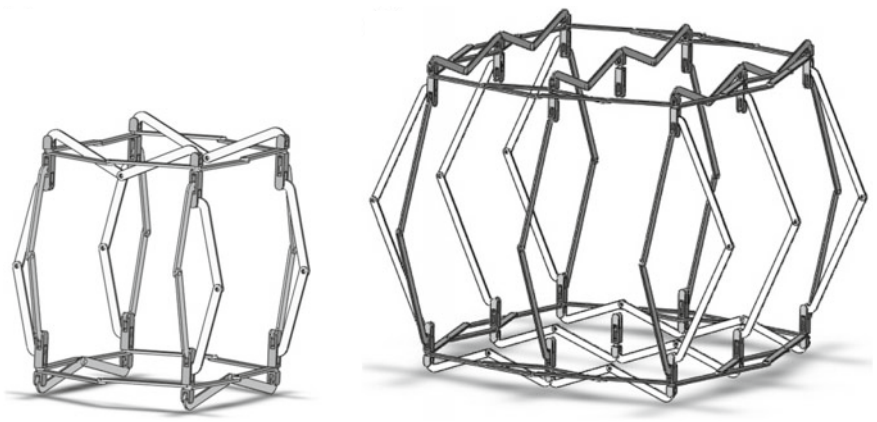


Fig. 11 Three-dof deployable mechanisms

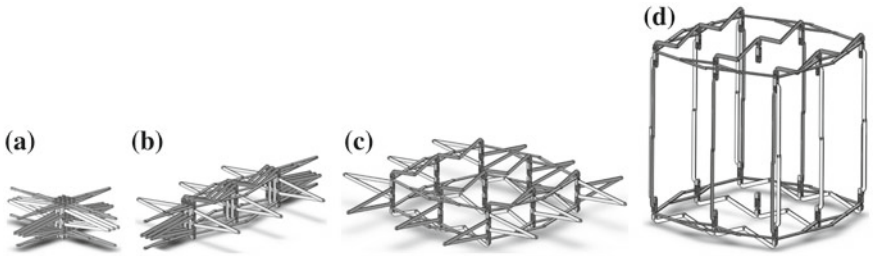


Fig. 12 Simulation of the three-dof DM: compacted configuration (a); partial expansion in one (b) and two (c) directions; deployed configuraion (d)

5 Conclusion

A family of unit-based mechanisms able to deploy and compact in two or three directions independently has been reviewed. In the deployable unit, Sarrus linkages provide the motion directions, and the scissor linkages connect the units, while maintaining the dof of the whole system constant. The DUs are connected by merging links and adding revolute joints. Thus, DMs with various boundary outlines and different dof are obtained. Simulations verify the feasibility of the proposed designs.

References

1. Chen, Y.: Design of structural mechanism. PhD Thesis, University of Oxford, Oxofrd (2003)
2. Ding, X.L., Yang, Y., Dai, J.S.: Topology and kinematic analysis of color-changing ball. Mech. Mach. Theory **46**(1), 67–81 (2011)
3. Escrig, F., Valcarcel, J.P., Sanchez, J.: Deployable cover on a swimming pool in Seville. J. Int. Assoc. Shell Spatial Struct. **37**(1), 39–70 (1996)
4. Fazli, N., Abedian, A.: Design of tensegrity structures for supporting deployable mesh antennas. Scientia Iranica **18**(5), 1078–1087 (2011)
5. Gantes, C.J., Konitopoulou, E.: Geometric design of arbitrarily curved bi-stable deployable arches with discrete joint size. Int. J. Solids Struct. **41**(20), 5517–5540 (2004)
6. Hoberman, C.: Radial expansion/retraction truss structures. US Patent 5024031 (1991)
7. Lu, S.N., Zlatanov, D., Ding, X.L., Molifino, R., Zoppi, M.: A novel deployable mechanism with two decoupled degrees of freedom. In: Proceedings of the ASME IDETC/CIE 2013, p. 13187 (2013)
8. Nagaraj, B.P., Pandiyan, R., Ghosal, A.: Kinematics of pantograph masts. Mech. Mach. Theory **44**(4), 822–834 (2009)
9. Pellegrino, S., Guest, S.D.: IUTAM-IASS Symposium on Deployable Structures: Theory and Applications. Kluwer Academic Publishers, Dordrecht (2000)
10. Puig, L., Barton, A., Rando, N.: A review on large deployable structures for astrophysics missions. Acta Astronautica **67**(1), 12–26 (2010)
11. Wei, G.W., Ding, X.L., Dai, J.S.: Mobility and geometric analysis of the Hoberman switch-picth ball and its variant. ASME Trans. J. Mech. Robot. 2(3), 031,010-1-031,010–9 (2010)
12. You, Z., Pellegrino, S.: Foldable bar structures. Int. J. Solids Struct. **15**(34), 1825–1847 (1997)
13. Zhao, J.S., Chu, F.L., Feng, Z.J.: The mechanism theory and application of deployable structures based on SLE. Mech. Mach. Theory **44**(2), 324–335 (2009)

Motion Capability of the 3-RPS Cube Parallel Manipulator

Latifah Nurahmi, Josef Schadlbauer, Manfred Husty, Philippe Wenger and Stéphane Caro

Abstract This chapter deals with the analysis of motion capability of the 3-RPS Cube parallel manipulator. The constraint equations of the manipulator are first expressed and it is shown that its moving-platform is capable of any orientation determined by a three parameter set u, v, w. The translation part of the motion is coupled to these three parameters. It is shown that this type of three dof motion has been studied by Darboux in 1897. Moreover, the moving-platform can perform the Vertical Darboux Motion, namely, it can rotate and translate about and along the same axis simultaneously. The surface generated by the moving-platform path turns out to be a right-conoid. The axodes generated by the motion are two coinciding lines passing through the origin of the fixed frame.

Keywords 3-RPS cube · Parallel manipulator · Motion type · Axodes · Vertical Darboux motion

L. Nurahmi · P. Wenger · S. Caro (✉)
Institut de Recherche en Communications et Cybernétique de Nantes, Nantes, France
e-mail: latifah.nurahmi@irccyn.ec-nantes.fr

P. Wenger
e-mail: philippe.wenger@irccyn.ec-nantes.fr

S. Caro
e-mail: stephane.caro@irccyn.ec-nantes.fr

J. Schadlbauer · M. Husty
Institute for Basic Sciences in Engineering, Unit for Geometry and CAD,
University of Innsbruck, Innsbruck, Austria
e-mail: josef.schadlbauer@uibk.ac.at

M. Husty
e-mail: manfred.husty@uibk.ac.at

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_54,

1 Introduction

The 3-RPS parallel manipulator with cube-shaped base was analyzed in [1]. The kinematic characteristics of this mechanism were studied in [2] by identifying the principal screws and the authors showed that the manipulator can rotate in any direction and the corresponding axes do not intersect. It is shown in [3] that the mechanism is capable to perform a motion along its diagonal, which is known as the Vertical Darboux Motion (VDM). Several mechanical generators of the VDM were later revealed by Lee and Hervé [4], in which one point in the platform is compelled to move in a plane. By analyzing the locus of series of the Instantaneous Screw Axis (ISA), Chen et al. showed in [5] that this mechanism may perform parasitic motions, in which the rotations are not about a fixed point or axis.

Using Study-parameters $x_0, x_1, x_2, x_3, y_0, y_1, y_2, y_3$ to describe the spatial Euclidean displacements [6], the motion capability of the 3-RPS Cube manipulator is analyzed in more details in this chapter. To characterize the general motion, the set of constraint equations are derived. They reveal the existence of only one operation mode. The chapter shows that the moving platform can produce any orientation and gives the complete parametrization of these orientations. Under the condition that the prismatic lengths are equal, the moving-platform of the manipulator is able to perform the VDM. It follows from Bottema and Roth [7] that this motion is the result of a rotation about an axis and a harmonic translation along the same axis. Eventually, the ISA are illustrated during this motion and turn out to be congruent to each other.

2 Manipulator Architecture and Constraint Equations

The 3-RPS Cube parallel manipulator shown in Fig. 1, is composed of a cube-base, an equilateral triangle platform, and three identical legs. Each leg is composed of a revolute joint, an actuated prismatic joint and a spherical joint mounted in series. The fixed frame Σ_0 is shifted along $\sigma_0 = [h_0, h_0, h_0]$ from the center of the base in order to fulfil the identity condition, as shown by the large and red dashed-box in Fig. 1. Likewise, the origin of the moving frame is shifted along $\sigma_1 = [h_1, h_1, h_1]$ as described by the small and blue dashed-box in Fig. 1.

The revolute joint in the ith ($i = 1 \ldots 3$) leg is located at point A_i, its axis being along vector $\mathbf{s}_i$, while the spherical joint is located at point B_i, the ith corner of the moving-platform. The distance between the origin O of the base frame Σ_0 and point A_i is equal to $h_0\sqrt{2}$. The axes $\mathbf{s}_1$, $\mathbf{s}_2$ and $\mathbf{s}_3$ are orthogonal to each other. The moving-platform has an equilateral triangle shape and its circumradius is equal to $d = h_1\sqrt{6}/3$.

As the direction of the ith prismatic joint is perpendicular to vector $\mathbf{s}_i$, line A_iB_i lies in a plane normal to $\mathbf{s}_i$. The length of the ith prismatic joint is denoted as r_i.

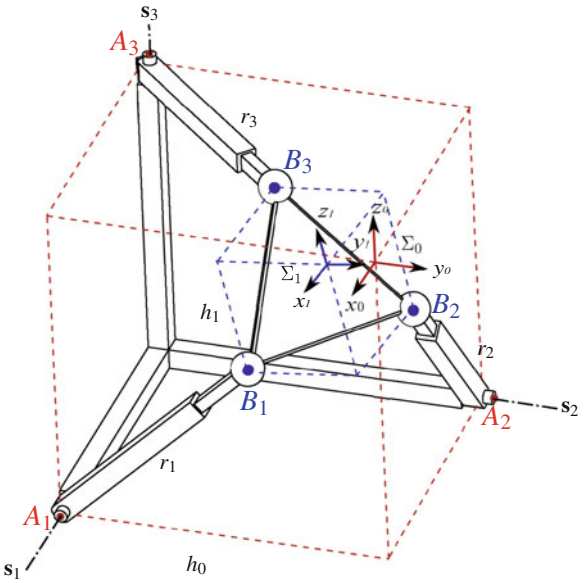


Fig. 1 The 3-RPS Cube parallel manipulator

By using the Study kinematic mapping, the constraint equations, which comprise three plane conditions ($g_1 - g_3$), three distance conditions ($g_4 - g_6$), the Study condition (g_7) and a normalizing condition (g_8), are enumerated in Eq. (1). Their solutions give the possible poses of the platform expressed in frame Σ_0. A primary decomposition has been computed and shows that the manipulator has only one operation mode contrary to the 3-RPS manipulator with planar equilateral base and platform, which has two operation modes [8].

$$
\begin{aligned}
g_1 &: -h_1x_0x_2 + h_1x_0x_3 - h_1x_1x_2 - h_1x_1x_3 - x_0y_1 + x_1y_0 - x_2y_3 + x_3y_2 = 0\\
g_2 &: h_1x_0x_1 - h_1x_0x_3 - h_1x_1x_2 - h_1x_2x_3 - x_0y_2 + x_1y_3 + x_2y_0 - x_3y_1 = 0\\
g_3 &: -h_1x_0x_1 + h_1x_0x_2 - h_1x_1x_3 - h_1x_2x_3 - x_0y_3 - x_1y_2 + x_2y_1 + x_3y_0 = 0\\
g_4 &: 2h_0^2x_0^2 + 2h_0^2x_1^2 + 2h_0^2x_2^2 + 2h_0^2x_3^2 - 4h_0h_1x_0^2 + 4h_0h_1x_1^2 - 8h_0h_1x_2x_3 + 2h_1^2x_0^2 + 2h_1^2x_1^2\\
&\quad + 2h_1^2x_2^2 + 2h_1^2x_3^2 - r_1^2x_0^2 - r_1^2x_1^2 - r_1^2x_2^2 - r_1^2x_3^2 - 4h_0x_0y_2 - 4h_0x_0y_3 - 4h_0x_1y_2\\
&\quad + 4h_0x_1y_3 + 4h_0x_2y_0 + 4h_0x_2y_1 + 4h_0x_3y_0 - 4h_0x_3y_1 + 4h_1x_0y_2 + 4h_1x_0y_3 - 4h_1x_1y_2\\
&\quad + 4h_1x_1y_3 - 4h_1x_2y_0 + 4h_1x_2y_1 - 4h_1x_3y_0 - 4h_1x_3y_1 + 4y_0^2 + 4y_1^2 + 4y_2^2 + 4y_3^2 = 0\\
g_5 &: 2h_0^2x_0^2 + 2h_0^2x_1^2 + 2h_0^2x_2^2 + 2h_0^2x_3^2 - 4h_0h_1x_0^2 - 8h_0h_1x_1x_3 + 4h_0h_1x_2^2 + 2h_1^2x_0^2 + 2h_1^2x_1^2\\
&\quad + 2h_1^2x_2^2 + 2h_1^2x_3^2 - r_2^2x_0^2 - r_2^2x_1^2 - r_2^2x_2^2 - r_2^2x_3^2 - 4h_0x_0y_1 - 4h_0x_0y_3 + 4h_0x_1y_0 \qquad (1)\\
&\quad - 4h_0x_1y_2 + 4h_0x_2y_1 - 4h_0x_2y_3 + 4h_0x_3y_0 + 4h_0x_3y_2 + 4h_1x_0y_1 + 4h_1x_0y_3 - 4h_1x_1y_0\\
&\quad - 4h_1x_1y_2 + 4h_1x_2y_1 - 4h_1x_2y_3 - 4h_1x_3y_0 + 4h_1x_3y_2 + 4y_0^2 + 4y_1^2 + 4y_2^2 + 4y_3^2 = 0\\
g_6 &: 2h_0^2x_0^2 + 2h_0^2x_1^2 + 2h_0^2x_2^2 + 2h_0^2x_3^2 - 4h_0h_1x_0^2 - 8h_0h_1x_1x_2 + 4h_0h_1x_3^2 + 2h_1^2x_0^2 + 2h_1^2x_1^2
\end{aligned}
$$

$$+ 2h_1^2x_2^2 + 2h_1^2x_3^2 - r_3^2x_0^2 - r_3^2x_1^2 - r_3^2x_2^2 - r_3^2x_3^2 - 4h_0x_0y_1 - 4h_0x_0y_2 + 4h_0x_1y_0$$
$$+ 4h_0x_1y_3 + 4h_0x_2y_0 - 4h_0x_2y_3 - 4h_0x_3y_1 + 4h_0x_3y_2 + 4h_1x_0y_1 + 4h_1x_0y_2 - 4h_1x_1y_0$$
$$+ 4h_1x_1y_3 - 4h_1x_2y_0 - 4h_1x_2y_3 - 4h_1x_3y_1 + 4h_1x_3y_2 + 4y_0^2 + 4y_1^2 + 4y_2^2 + 4y_3^2 = 0$$

$$g_7 : x_0y_0 + x_1y_1 + x_2y_2 + x_3y_3 = 0$$

$$g_8 : x_0^2 + x_1^2 + x_2^2 + x_3^2 - 1 = 0$$

3 The General Motion

The set of eight constraint equations is written as a polynomial ideal $\mathscr{I}$ with variables $x_0, x_1, x_2, x_3, y_0, y_1, y_2, y_3$ over the coefficient ring $\mathbb{C}[h_0, h_1, r_1, r_2, r_3]$.

$$\mathscr{I} = < g_1, g_2, g_3, g_4, g_5, g_6, g_7, g_8 > \tag{2}$$

The general motion performed by the 3-RPS Cube parallel manipulator is characterized by solving the ideal $\mathscr{I}$. The equations $g_1, g_2, g_3, g_4, g_5, g_6, g_7$ from ideal $\mathscr{I}$ can be solved linearly for variables $y_0, y_1, y_2, y_3, R_1, R_2, R_3$ [8], R_i being the square of the prismatic lengths, i.e., $R_i = r_i^2$, and $\delta = x_0^2 + x_1^2 + x_2^2 + x_3^2$. The Study-parameters become:

$$\begin{aligned}
y_0 &= \frac{x_1^2x_2 + x_1^2x_3 + x_1x_2^2 + x_1x_3^2 + x_2^2x_3 + x_2x_3^2}{\delta} \\
y_1 &= -\frac{x_0^2x_2 - x_0^2x_3 + x_0x_2^2 + x_0x_3^2 - x_2^2x_3 + x_2x_3^2}{\delta} \\
y_2 &= \frac{x_0^2x_1 - x_0^2x_3 - x_0x_1^2 - x_0x_3^2 - x_1^2x_3 + x_1x_3^2}{\delta} \\
y_3 &= -\frac{x_0^2x_1 - x_0^2x_2 + x_0x_1^2 + x_0x_2^2 - x_1^2x_2 + x_1x_2^2}{\delta}
\end{aligned} \tag{3}$$

The terms R_i are also expressed in terms of x_0, x_1, x_2, x_3. However, the expressions are very lengthy and the readers may refer to [9]. The remaining Study-parameters are still linked in equation $g_8 : x_0^2 + x_1^2 + x_2^2 + x_3^2 - 1 = 0$, which amounts to a hypersphere equation in space (x_0, x_1, x_2, x_3). The hypersphere equation can be parametrized by setting $x_0 = \sin(u)\sin(v)\cos(w)$, $x_1 = \sin(u)\sin(v)\sin(w)$, $x_2 = \sin(u)\cos(v)$, and $x_3 = \cos(u)$. As a result, the manipulator workspace can be parametrized by three parameters u, v, w.

This parametrization provides us with an interpretation of the general motion performed by the manipulator. The moving-platform of the manipulator is capable of all orientations determined by the parameter set u, v, w. The translational motion is coupled to the orientations via Eq. (3). Then, all Study-parameters and the transformation matrix can be obtained in terms of u, v, w. The readers may refer to [9] as the expressions are very long and cannot be displayed here.

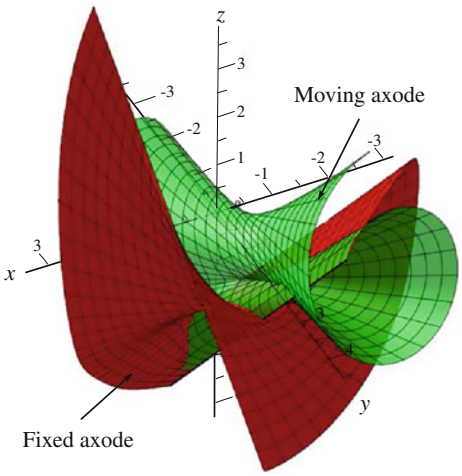


Fig. 2 ISA axodes of general motion

Note that the motion of the manipulator platform is essentially determined by the fact that three points (B_1, B_2, B_3) move in three planes. According to Bottema and Roth [7] this motion has been studied already by Darboux in 1897. The special case at hand, namely that the three planes are mutually orthogonal, is treated in great detail in [7]. We mention especially that the workspace boundary of each point of the moving platform (with exception of the vertices) is bounded by a Steiner surface.

The ISA of any motion of the moving-platform are obtained from the entries of the velocity operator:

$$\mathbf{A} = \dot{\mathbf{T}}\ \mathbf{T}^{-1} \tag{4}$$

The components of matrix **A** are the twists of the ISA. To illustrate a simple example of a one parameter motion subset of the three parameter motion, the parameters are set to $u = t, v = t, w = t$. Figure 2 shows the axodes obtained from the twists of the ISA and for t varying between $-\pi/2$ and $\pi/2$. The red surface shows the fixed axode, for which all twists are computed with respect to the fixed coordinate frame. The green surface depicts the moving axode with respect to the moving coordinate frame.

4 The Vertical Darboux Motion

The condition for the manipulator to generate the VDM is that all prismatic lengths are equal, i.e., $r_1 = r_2 = r_3$. By solving the direct kinematics of the manipulator with the same prismatic lengths, the Study-parameters to perform the VDM yield

$x_1 = x_2 = x_3$ and $y_1 = y_2 = y_3$. By substituting those values into the ideal $\mathscr{I}$, the set of eight constraint equations becomes:

$$\begin{aligned}\mathscr{I} : \{&-x_0y_1 - 2x_1^2 + x_1y_0 = 0, -x_0y_1 - 2x_1^2 + x_1y_0 = 0, -x_0y_1 - 2x_1^2 + x_1y_0 = 0,\\ &-R_1x_0^2 - 3R_1x_1^2 + 2x_0^2 - 8x_0y_1 + 22x_1^2 + 8x_1y_0 + 4y_0^2 + 12y_1^2 = 0,\\ &-R_2x_0^2 - 3R_2x_1^2 + 2x_0^2 - 8x_0y_1 + 22x_1^2 + 8x_1y_0 + 4y_0^2 + 12y_1^2 = 0,\\ &-R_3x_0^2 - 3R_3x_1^2 + 2x_0^2 - 8x_0y_1 + 22x_1^2 + 8x_1y_0 + 4y_0^2 + 12y_1^2 = 0,\\ &y_0x_0 + 3y_1x_1 = 0, x_0^2 + 3x_1^2 - 1 = 0\}. \quad (5)\end{aligned}$$

It follows from Eq. (5) that the first three constraint equations are the same. Likewise, the next three equations are identical. Mathematically, one has to find the case of 1-*dof* motion, as known as cylindrical motion, with one parameter that describes the VDM. Equation 5 can be solved linearly for the variables R_i, y_0, y_1 in terms of x_0, x_1, as follows:

$$\begin{aligned}R_1 = R_2 = R_3 &= -\frac{(-2x_0^4 - 44x_0^2x_1^2 - 162x_1^4)}{(x_0^4 + 6x_0^2x_1^2 + 9x_1^4)}\\ y_0 = \frac{6x_1^3}{(x_0^2 + 3x_1^2)}, \quad y_1 &= -\frac{2x_0x_1^2}{(x_0^2 + 3x_1^2)}.\end{aligned} \quad (6)$$

It is confirmed from Eq. (6) that the manipulator can perform the VDM if all prismatic lengths are the same. The remaining Study-parameters x_0 and x_1 are still linked to the eighth equation in $x_0^2 + 3x_1^2 - 1 = 0$, which is simply an ellipse equation in the space x_0 and x_1. This ellipse equation can be parametrized by $x_0 = \cos(u)$ and $x_1 = \frac{1}{3}\sin(u)\sqrt{3}$.

As a result, the workspace of the manipulator performing the VDM is parametrized by the parameter u. Hence, the Study-parameters are expressed as:

$$\begin{aligned}&x_0 = \mathrm{c}(u) \quad x_1 = \frac{1}{3}\mathrm{s}(u)\sqrt{3} \quad y_0 = \frac{2}{3}\mathrm{s}(u)^3\sqrt{3} \quad y_1 = -\frac{2}{3}\mathrm{c}(u)\mathrm{s}(u)^2\\ &x_2 = \frac{1}{3}\mathrm{s}(u)\sqrt{3} \quad x_3 = \frac{1}{3}\mathrm{s}(u)\sqrt{3} \quad y_2 = -\frac{2}{3}\mathrm{c}(u)\mathrm{s}(u)^2 \quad y_3 = -\frac{2}{3}\mathrm{c}(u)\mathrm{s}(u)^2\end{aligned} \quad (7)$$

where $\mathrm{s}(u) = \sin(u)$, $\mathrm{c}(u) = \cos(u)$.

Therefore, the possible poses of the moving-platform can be expressed by the following transformation matrix:

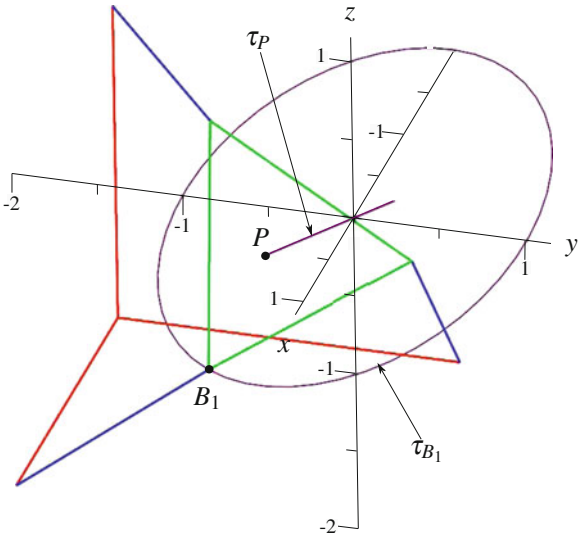


Fig. 3 Trajectories of points B_1 and P

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ a & \frac{4}{3}\mathrm{c}(u)^2 - \frac{1}{3} & -\frac{2}{3}\mathrm{s}(u)(\mathrm{c}(u)\sqrt{3} - \mathrm{s}(u)) & -\frac{2}{3}\mathrm{s}(u)(\mathrm{c}(u)\sqrt{3} - \mathrm{s}(u)) \\ a & -\frac{2}{3}\mathrm{s}(u)(\mathrm{c}(u)\sqrt{3} - \mathrm{s}(u)) & \frac{4}{3}\mathrm{c}(u)^2 - \frac{1}{3} & -\frac{2}{3}\mathrm{s}(u)(\mathrm{c}(u)\sqrt{3} - \mathrm{s}(u)) \\ a & -\frac{2}{3}\mathrm{s}(u)(\mathrm{c}(u)\sqrt{3} - \mathrm{s}(u)) & -\frac{2}{3}\mathrm{s}(u)(\mathrm{c}(u)\sqrt{3} - \mathrm{s}(u)) & \frac{4}{3}\mathrm{c}(u)^2 - \frac{1}{3} \end{bmatrix} \quad (8)$$

where $a = \frac{4}{3}\sin(u)^2$.

5 Trajectory of the Moving-Platform Performing the VDM

Let us consider point B_1 moving in the **yz**-plane and the geometric center P of the moving-platform. The paths followed by those two points are obtained by setting $u = -\frac{\pi}{2} \ldots \frac{\pi}{2}$ by using the transformation matrix **T** defined in Eq. (8).

It appears that those two paths are different as shown in Fig. 3. Point P moves along a straight line denoted as τ_P, whereas point B_1 moves along a planar ellipse that is parallel to the **yz**-plane (τ_{B_1}).

Let us take all segments joining point B_1 to any point of segment B_2B_3 and plot the paths of all points on those segments. All those paths are planar ellipses, except the path followed by point P. Accordingly, the set of all paths forms a ruled surface called *Right-conoid*, which is illustrated in yellow in Fig. 4.

This type of ruled surfaces is generated by moving a straight line such that it intersects perpendicularly a fixed straight line, called the axis of the Right-conoid. The fixed straight line followed by point P is the axis of the Right-conoid.

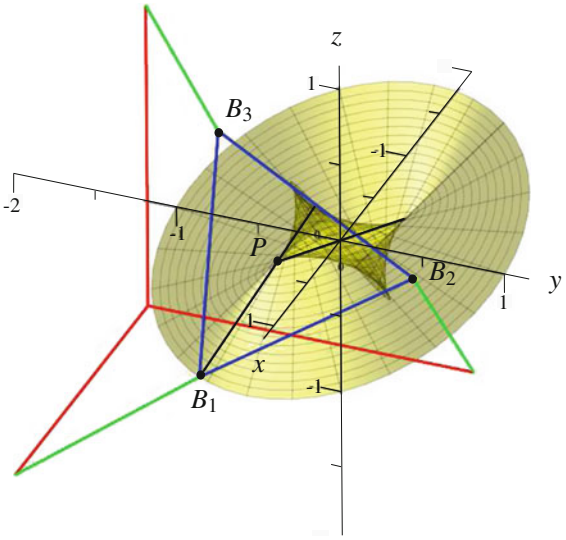


Fig. 4 Right-conoid of the VDM

6 Axodes of the Manipulator Performing the VDM

Having the parametrization of the VDM performed by the 3-RPS Cube parallel manipulator in terms of Study-parameters, it is relatively easy to compute the ISA. The possible poses of the manipulator for each instant in this special motion will only allow the orientations that are given by one parameter u. The ISA are obtained from the entries of the velocity operator:

$$\mathbf{A} = \dot{\mathbf{T}}\,\mathbf{T}^{-1} \tag{9}$$

By setting $u = t$, matrix $\mathbf{A}$ becomes:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{8}{3}\,\mathrm{c}(t)\,\mathrm{s}(t) & 0 & -\frac{2}{3}\sqrt{3} & \frac{2}{3}\sqrt{3} \\ \frac{8}{3}\,\mathrm{c}(t)\,\mathrm{s}(t) & \frac{2}{3}\sqrt{3} & 0 & -\frac{2}{3}\sqrt{3} \\ \frac{8}{3}\,\mathrm{c}(t)\,\mathrm{s}(t) & -\frac{2}{3}\sqrt{3} & \frac{2}{3}\sqrt{3} & 0 \end{bmatrix} \tag{10}$$

The instantaneous screw axis of the moving-platform is obtained from the components of matrix $\mathbf{A}$ after normalization:

$$ISA = \left[\frac{1}{\sqrt{3}},\ \frac{1}{\sqrt{3}},\ \frac{1}{\sqrt{3}},\ \frac{4}{3}\,\mathrm{c}(t)\mathrm{s}(t),\ \frac{4}{3}\,\mathrm{c}(t)\mathrm{s}(t),\quad \frac{4}{3}\,\mathrm{c}(t)\mathrm{s}(t)\right]^T \tag{11}$$

All twists of the manipulator are collinear. As a consequence, the fixed axode generated by the ISA is a straight line of unit vector $[1/\sqrt{3}, 1/\sqrt{3}, 1/\sqrt{3}]^T$. In the moving coordinate frame, the moving axode corresponding to this motion is congruent with the fixed axode as depicted in Fig. 4. However, the moving axode does not appear clearly as it is congruent with the fixed axode. Indeed, the moving axode internally slides and rolls onto the fixed axode.

7 Conclusions

The analysis of the motion capability of the 3-RPS Cube parallel manipulator was carried out by parametrizing the general motion. It was shown that the moving-platform of the manipulator is capable of all orientations parametrized by the set u, v, w. The 3-dof motion was shown to be already discussed by Darboux in 1897 and in very detail by Bottema and Roth. A special one parameter submotion is the Vertical Darboux Motion (VDM), which is obtained by keeping all leg parameters the same. In this VDM the moving-platform rotates about an axis and translates along the same direction. The paths followed by all points belonging to the moving-platform, except its geometric center, are ellipses that form a ruled surface called Right-conoid. The geometric center of the platform moves along a straight line. Finally, since the 3-RPS Cube parallel manipulator has only one operation mode, the general motion and the VDM occur inside the same operation mode. The effect of changes in the geometric parameters will be the subject of future work.

References

1. Huang, Z., Fang, Y.: Kinematic characteristics analysis of 3 DOF in-Parallel actuated pyramid mechanisms. Mech. Mach. Theory **31**(8), 1009–1018 (1996)
2. Huang, Z., Wang, J.: Identification of principal screws of 3-DOF parallel manipulators by quadric degeneration. Mech. Mach. Theory **36**(8), 893–911 (2001)
3. Huang, Z., Dejun, M., Zeng, D.: The screw motion simulation of 3-RPS parallel pyramid mechanism. In: Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation, pp. 2860–2864 (2007)
4. Lee, C.-C., Hervé, J.: On the Vertical Darboux Motion. Latest Advance in Robot Kinematics, pp. 99–106. Springer, Dordrecht (2012)
5. Chen, Z., Ding, H., Cao, W., Huang, Z.: Axodes Analysis of the Multi DOF Parallel Mechanisms and Parasitic Motion. In: Proceedings of the ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference (2013)
6. Husty, M.L., Pfurner, M., Schröcker, H.-P., Brunnthaler, K.: Algebraic methods in mechanism analysis and synthesis. Robotica **25**(6), 661–675 (2007)
7. Bottema, O., Roth, B.: Theoretical Kinematics, pp. 304–322. Dover Publishing, New York (1990)
8. Schadlbauer, J., Nurahmi, L., Husty, M., Wenger, P., Caro, S.: Operation Modes in Lower-Mobility Parallel Manipulators. In: Second Conference on Interdisciplinary Applications of Kinematics, Lima, Peru, Sept, pp. 9–11 (2013)
9. http://www.irccyn.ec-nantes.fr/~caro/ARK2014/AppendixARK2014paper49.pdf

Coupling of Trajectories for Human–Robot Cooperative Tasks

Andrej Gams, Bojan Nemec, Tadej Petrič and Aleš Ude

Abstract Since human motion is not completely repeatable, the synthesis of robot trajectories for human–robot cooperation must allow for easy modulation. This should take place by taking into account the external sensory feedback that enables interpretation of the person's intentions. In this chapter we present a method for coupling of robot trajectories to the measured force feedback arising from the interaction with the environment, where the environment can be an object, a robot, or a person. The algorithm is based on Dynamic Movement Primitives, a kinematic representation of robot trajectories. In the chapter we show how to consider the measured external forces and torques within the kinematic DMP framework. We further develop the approach by introducing iterative learning control in order to anticipate the behavior and achieve minimal errors of motion for stationary conditions. The usefulness of the proposed approach was demonstrated on two KUKA LWR robots performing a bimanual human–robot collaborative task.

Keywords Bimanual tasks · Force feedback · Dynamic movement primitives · Iterative learning control

A. Gams (✉) · B. Nemec · T. Petrič · A. Ude
Humanoid and Cognitive Robotics Lab, Department of Automatics, Biocybernetics and Robotics, Jožef Stean Institute, Jamova Cesta 39, 1000 Ljubljana, Slovenia
e-mail: andrej.gams@ijs.si

B. Nemec
e-mail: bojan.nemec@ijs.si

T. Petrič
e-mail: tadej.petric@ijs.si

A. Ude
e-mail: ales.ude@ijs.si

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_55,

1 Introduction

A robotic household assistant acting in an uncontrolled and cluttered environment, such as the kitchen [8], will use its sensors and previously programmed or even attained knowledge to synthesize trajectories to execute given tasks. Known algorithms, for example generalization from existing knowledge, and modifying trajectories online [3], might not result in sufficiently accurate executions of trajectories for desired manipulation tasks. The trajectories will have to be adapted through learning and autonomous exploration, and modulated accordingly.

Modulating predefined trajectories—in the sense of being pre-calculated by an algorithm—greatly depends on the means of encoding the trajectory. Complete time-position series might be less suitable, as the means of modulation in order to achieve a smooth trajectory might be complicated [5]. On the other hand, splines define the complete trajectory with a small set of points, but complete re-calculation of the parameters is required for modulation. In the chapter we rely on the framework of dynamic movement primitives (DMPs) [6, 7], a kinematic representation which combines differential equations and Gaussian-like kernel functions. They allow easy and smooth modulation by adapting only a few parameters to change their attractor landscape. The latter can be beneficially used in reinforcement learning [9, 14, 15], generalization [16] and for combining of separate trajectories.

DMPs allow modulation both in the canonical as well as in the transformation systems. By combining the former with force or vision feedback, we can execute complex periodic tasks [13]. The latter can be used, for example, for obstacle avoidance [6]. In this chapter we propose a novel approach to modify the transformation system with feedback. We propose recording the sensory information as the robot executes a given trajectory and then using said feedback to improve performance the next time it moves down the exact same trajectory. To do so we propose learning of a so-called coupling term, which we plug into the differential equation of the DMP at both the velocity and acceleration levels. The coupling term is composed of an on-line force/torque feedback and a learned feed-forward signal. The latter is learned in several executions of the same trajectory using iterative learning control (ILC) [1].

ILC incorporates information rich signals from previous attempts at a trajectory in the latest iteration. As ILC only adapts the control input, the amount of training required is low. ILC is also known for converging fast [1]. The contribution of this chapter is in combining the DMP framework with the ILC in a single, robust, and adaptive system capable of coupling trajectories for human–robot cooperative tasks. The introduction of force feedback also expands the kinematic domain of the DMPs to dynamic behavior.

In Sect. 2 of the chapter we give a short introduction on the DMP framework, followed by the proposed coupling approach for cooperative tasks in Sect. 3. Experimental results are presented in Sect. 4 and concluding remarks and discussion in Sect. 5.

2 Dynamic Movement Primitives

The DMP framework was first introduced by Ijspeert et al. [7] and has since been thoroughly discussed and applied throughout the robotics community [6, 16]. In the following we provide a basic introduction for a single degree of freedom (DOF) based on [16]. A DOF is denoted by y, and can either a joint- or a task-space coordinate. In our case it is one of the task-space coordinates. A system of nonlinear differential equations

$$\tau \dot{z} = \alpha_z(\beta_z(g - y) - z) + f(x), \tag{1}$$

$$\tau \dot{y} = z. \tag{2}$$

defines a DMP. The term $f(x)$ is linearly combined from nonlinear radial basis functions as

$$f(x) = \frac{\sum_{i=1}^{N} w_i \Psi_i(x)}{\sum_{i=1}^{N} \Psi_i(x)} x, \tag{3}$$

$$\Psi_i(x) = \exp\left(-h_i\,(x - c_i)^2\right). \tag{4}$$

Here c_i are the centers of radial basis functions distributed along the trajectory and $h_i > 0$ their widths. If parameters $\alpha_z,\ \beta_z,\ \tau > 0$ and $\alpha_z = 4\beta_z$, the linear part of the system (1)–(2) is critically damped, resulting in a unique attractor point at $y = g,\ z = 0$. In (1), (3) and (4) a phase variable x is used, which removes direct dependency of f on time. Its dynamics is defined by a canonical system given by

$$\tau \dot{x} = -\alpha_x x, \tag{5}$$

with initial value $x(0) = 1$. α_x is a positive constant.

The weights w_i define the actual shape of the encoded trajectory. See [7] and [16] for the learning of the weights. To realize multiple DOFs, separate sets of (1)–(4) are used and a single canonical system synchronizes them.

3 Coupling of Trajectories

In this section we first show how to couple DMPs, followed by the combination with iterative learning control.

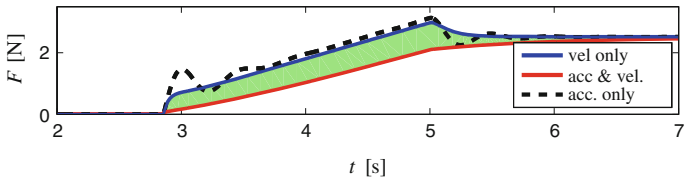


Fig. 1 The difference of coupling with $C = C_{fb}$ at the velocity level (*blue*), velocity and acceleration level (*red*), with the *green area* highlighting the difference. Acceleration level modulation results of a DMP are in *black dotted*

3.1 Coupling for Cooperative Tasks

The framework of DMPs allows online modulation, for example including a virtual repulsive force to prevent from moving beyond defined limits [5]. Such a force is defined by changing (2) into

$$\tau \dot{y} = z + h(y), \tag{6}$$

with $h(y)$ a predefined function of the position, while leaving (1) in the original form.

We propose modifying this type of modulation at the velocity level to include a coupling term C composed of (1) a feedback term C_{fb} on the real, measured force, which occurs upon contact with the environment; and (2) a feed-forward term C_{ff}, which includes information from a previous trial. Note that the environment is everything that is not the robot itself, i.e. a static obstacle, another robot, a person. Furthermore, to reduce the overshoot of forces upon contact, we propose adding a derivative of C to the acceleration level. Therefore, a DMP is modified from (1, 2) into

$$\tau \dot{z} = \alpha_z(\beta_z(g - y) - z) + f(x) + c_2 \dot{C}, \tag{7}$$

$$\tau \dot{y} = z + C, \tag{8}$$

$$C = C_{fb} + C_{ff}, \tag{9}$$

where c_2 is a positive constant. Figure 1 shows the simulated results of using the proposed approach with $C = C_{fb}$ only. The plot shows the resulting forces of contact when a predefined trajectory is coupled to a static object. The red line shows the force using velocity and acceleration modulation with $c_2 = 30$, determined empirically. The blue line shows the result of only velocity modulation, with the green area highlighting the difference. The black dotted line shows only acceleration level modulation. The latter results in larger oscillations, reducing the effectiveness of learning the feed-forward term.

3.2 Learning the Feed-Forward Term

Minimizing the error of force upon contact with the environment can be achieved with augmenting the coupling term with a feed-forward term C_{ff}. It can be learned through a few executions of the exact same trajectory using the iterative learning control. Thus, there is no need to accurately model the dynamics of the robot and the environment. A thorough review by Bristow et al. [1] gives details on ILC.

To learn C_{ff}, the task needs to allow several repetitions with exactly the same initial conditions. Each time, the sensors register the resulting force and update the term. The learning update is then defined for both the feedback and feed-forward terms as suggested by the ILC theory [1]

$$C_i = C_{fb} + C_{ff} = c\, e_i + F_{c,i}, \tag{10}$$
$$F_{c,i} = Q(F_{c,i-1} + L\, c\, \dot{e}_{i-1}), \tag{11}$$
$$e_i = F_d - F_i. \tag{12}$$

Here index i denotes the i-th epoch and c is the force gain. The coupling force error e_i is calculated from the difference of the desired contact force F_d and the measured force F_i. $F_{c,i}$ is the learned feed-forward term and Q and L are positive scalars. The coupling term given by (10) is known as current iteration learning control, since it incorporates instantaneous feedback in the first term and learning update in the second term. The tunable parameters are Q, L and c and we used $Q = 0.99$, $L = 1$ and $c = 0.5$ in our experiments.

3.3 Bimanual Tasks

While well defined approaches for controlling bimanual tasks with centralized controller exist [2, 11], for coupling of two robots with decentralized controllers, their trajectories of motion in the form of DMPs can be coupled into *cooperative DMPs* with

$$\tau \dot{z_1} = \alpha_z(\beta_z(g_1 - y_1) - z_1) + f_1(x) + c_2\dot{C}_{1,2}, \tag{13}$$
$$\tau \dot{y_1} = z_1 + C_{1,2}, \tag{14}$$
$$\tau \dot{z_2} = \alpha_z(\beta_z(g_2 - y_2) - z_2) + f_2(x) + c_2\dot{C}_{2,1}, \tag{15}$$
$$\tau \dot{y_2} = z_2 + C_{2,1}, \tag{16}$$

following constrains apply to (13–16):

$$F_{1,2} = F_d - (F_1 - F_2), \tag{17}$$
$$F_{2,1} = -F_{1,2} = -k(d_d - d_a), \tag{18}$$

where F_i is the force measured at each end effector and F_d the desired force. In simulation the force is a function of the positions with k the stiffness of a virtual spring between the robots and d_d, d_a the desired and actual distances, respectively. Due to constraint (18), only one feed-forward term needs to be learned.

The numerical stability of both the coupling of DMPs and the learning algorithm can be analytically proven, but are beyond the scope of this chapter.

4 Experimental Evaluation

We conducted a real-world experiment where we coupled two independently controlled KUKA LWR robots holding a bar at the end of a box-lid. The other side was held by a person, thus combining the task of mutual coupling of the robots to each-other with adaptation to external, human interference. The latter is not fully repeatable and is subject to considerable noise due to the human in the loop. The task demanded that the robots—together with the human—place a lid with a very tight fit on a wooden box.

First, the initial DMP trajectories for cooperative closing of the lid were learned by demonstration. Then, the box was moved backward 12 cm ($-p_x$ direction) and right 7 cm ($p_y$direction) from the demonstrated position. The proposed coupling approach was used to correct for the new position of the box in eight epochs. p_x and p_y directions were corrected, while p_z (up–down) was not. The approach was applied to minimize the exerted force between the robots and towards the human, i.e., $F_{d,x,y} = 0$ for both robots. The resulting forces and positions of both robots are shown in Fig. 2.

The force plots in Fig. 2 show that the person had to push during the first three epochs (see F_y plots). Additionally, the left F_z plot shows that the lid did not fit in the first four epochs. Once it did, the forces were reduced to desired values at 0 N. Note that a F_z value represents the force the human is exerting on the lid, while positive values indicate simply the weight of the lid. Figure 3 shows the box and the lid after the demonstration and also after the first five epochs. Other consecutive epochs show minimal positional difference.

5 Discussion and Conclusion

Dynamic movement primitives with force feedbacks were considered in different scenarios. For example, [4] has shown how it can be applied for learning of periodic tasks.

For discrete tasks, as is the case presented in this chapter, it was applied by Pastor et al. [12]. In their approach, the authors first execute a trajectory and then apply a force-based feedback coupling to control the behavior of the robot under changed

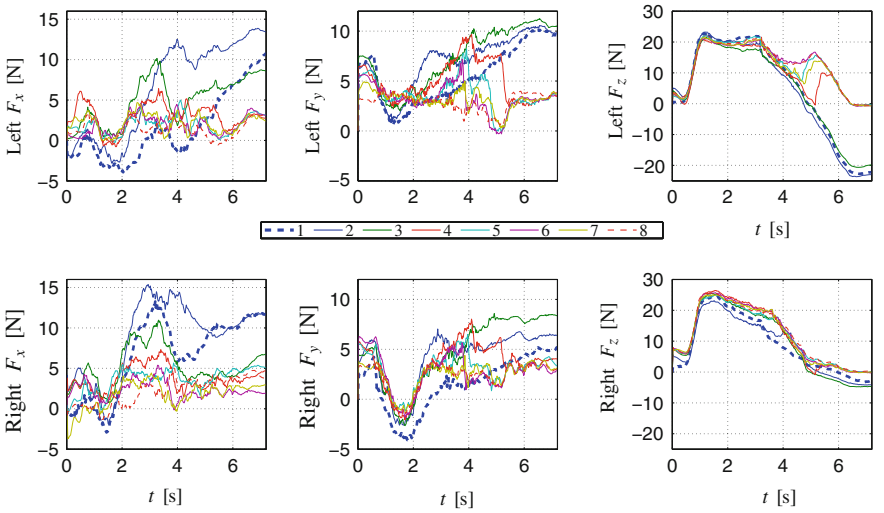


Fig. 2 Results of adaptation of forces for both robots. The legend denotes the *colors* of separate epochs for all six plots. The first is marked with *blue* the *dotted line* and the last with the *red dotted*

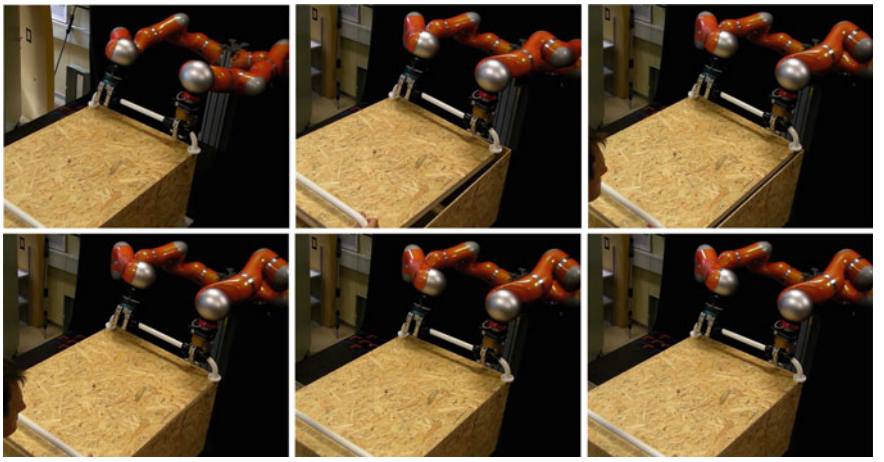


Fig. 3 The position of the box, lid, and the robots in the demonstration in the *leftmost picture*. Positions after each epoch, with epoch number increasing towards the *left*. Only five epochs are shown because the pictures of the final position after epoch five are practically identical

external conditions. As the approach relies only on feedback, some of the error is always present. Their approach utilizes purely acceleration-level coupling.

The approach by Kulvicius et al. [10], based on time-dependent DMP formulation proposes a similar feedback based coupling. The coupling, i.e. feedback gains are learned in several executions using hebbian-type learning. Similarly to [12], a feedback only approach does not ensure complete minimization of the error.

In this chapter we have shown how to couple the DMPs to the environment on both the velocity and acceleration levels. We have also shown that the combination with iterative learning control can effectively be utilized to achieve the desired force contact behavior. The robot learns to anticipate when a contact will occur, and prepares appropriately. The sensory feedback is always present in the system and assures that the robot gradually adapts to a different configuration, if needed. The system is robust enough to cope with noisy and un-repeatable human intervention during repetitions as we have shown in our experiment, where we coupled two robots and the operator to successfully execute the task of closing a wooden box.

References

1. Bristow, D., Tharayil, M., Alleyne, A.: A survey of iterative learning control. IEEE Cont. Syst. Mag. **26**(3), 96–114 (2006)
2. Chiacchio, P., Chiaverini, S., Siciliano, B.: Direct and inverse kinematics for coordinated motion tasks of a two-manipulator system. J. Dyn. Sys. Meas. Cont. **118**(4) (1996)
3. Forte, D., Gams, A., Morimoto, J., Ude, A.: On-line motion synthesis and adaptation using a trajectory database. Robot. Auton. Syst. **60**(10), 1327–1339 (2012)
4. Gams, A., Do, M., Ude, A., Asfour, T., Dillmann, R.: On-line periodic movement and force-profile learning for adaptation to new surfaces. In: 2010 10th IEEE-RAS International Conference on Humanoid Robots (Humanoids), pp. 560–565. Nashville (2010)
5. Gams, A., Ijspeert, A.J., Schaal, S., Lenarčič, J.: On-line learning and modulation of periodic movements with nonlinear dynamical systems. Auton. Robot. **27**(1), 3–23 (2009)
6. Ijspeert, A., Nakanishi, J., Pastor, P., Hoffmann, H., Schaal, S.: Dynamical movement primitives: learning attractor models for motor behaviors. Neural Comput. **25**(2), 328–373 (2013)
7. Ijspeert, A., Nakanishi, J., Schaal, S.: Movement imitation with nonlinear dynamical systems in humanoid robots. In: IEEE International Conference on Robotics and Automation (ICRA), vol. 2, pp. 1398–1403. Washington, DC (2002)
8. Khatib, O., Yokoi, K., Brock, O., Chang, K., Casal, A.: Robots in human environments. In: Proceedings of the First Workshop on Robot Motion and Control (RoMoCo), pp. 213–221 (1999)
9. Kober, J., Wilhelm, A., Oztop, E., Peters, J.: Reinforcement learning to adjust parametrized motor primitives to new situations. Auton. Robot. **33**, 361–379 (2012)
10. Kulvicius, T., Biehl, M., Aein, M.J., Tamosiunaite, M., Wrgtter, F.: Interaction learning for dynamic movement primitives used in cooperative robotic tasks. Robot. Auton. Syst. **61**, 1450–1459 (2013). doi: http://dx.doi.org/10.1016/j.robot.2013.07.009
11. Likar, N., Nemec, B., Žlajpah, L.: Virtual mechanism approach for dual-arm manipulation. Robotica **61**, 948–959 (2013). doi:10.1017/S0263574713000763
12. Pastor, P., Righetti, L., Kalakrishnan, M., Schaal, S.: Online movement adaptation based on previous sensor experiences. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 365–371. San Francisco (2011)
13. Petrič, T., Gams, A., Ijspeert, A.J., Žlajpah, L.: On-line frequency adaptation and movement imitation for rhythmic robotic tasks. Int. J. Robot. Res. **30**(14), 1775–1788 (2011)
14. Stulp, F., Theodorou, E.A., Schaal, S.: Reinforcement learning with sequences of motion primitives for robust manipulation. IEEE Trans. Robot. **28**(6), 1360–1370 (2012)
15. Tamosiunaite, M., Nemec, B., Ude, A., Wörgötter, F.: Learning to pour with a robot arm combining goal and shape learning for dynamic movement primitives. Robot. Auton. Syst. **59**(11), 910–922 (2011)
16. Ude, A., Gams, A., Asfour, T., Morimoto, J.: Task-specific generalization of discrete and periodic dynamic movement primitives. IEEE Trans. Robot. **26**(5), 800–815 (2010)

Dynamic Analysis of 4 Degrees of Freedom Redundant Parallel Manipulator

Samah Shayya, Sébastien Krut, Olivier Company, Cédric Baradat and François Pierrot

Abstract The chapter introduces the dynamic analysis of an actuatedly redundant parallel manipulator, called ARROW V1, with 4 Degrees of Freedom (DoF) providing three translations and one rotation (3T-1R). The dynamic measure used is of solid physical significance and applicable to all types of robots: the isotropic linear acceleration.

Keywords Parallel mechanism · Actuation redundancy · 4 DoFs robot · 3T-1R · Dynamic measure

1 Introduction

In recent decades, a noticeable trend towards the synthesis of lower-mobility parallel manipulators has been noticed, as to reply to the real industrial needs. In fact, regarding some tasks, 4 DoFs (3T-1R) parallel manipulators are sufficient. In others, where extra rotational dof is needed, it can be supplied by a turntable or an additional actuator on the platform. As a matter of fact, literature is very rich with such (3T-1R)

S. Shayya (✉) · C. Baradat
Tecnalia France, MIBI Building, 672 Rue du Mas de Verchant, 34000 Montpellier, France
e-mail: samah.shayya@tecnalia.com

C. Baradat
e-mail: cedric.baradat@tecnalia.com

S. Krut · O. Company · F. Pierrot
LIRMM, UMR 5506 CNRS/University of Montpellier, 161 rue Ada, 34095 Montpellier, France
e-mail: krut@lirmm.fr

O. Company
e-mail: company@lirmm.fr

F. Pierrot
e-mail: pierrot@lirmm.fr

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1_56,

parallel manipulators; among them we mention for instance: Delta robot [3] (with the additional R-U-P-U[1] chain), the Kanuk [11], the SMG in [1], the H4 in [9], the I4 in [7] and the Par4 in [8] with its industrialized version Adept Quattro [10] (fastest industrial pick-and-place robot). Another interesting family of robots is that of [5] presenting fully-isotropic parallel 4 dofs (3T-1R) manipulators, in addition to the decoupled ones.

This was just a sample, but still many others exist; an interested reader may refer to [5] and citations therein for elaborate details regarding the state-of-art matter.

Unfortunately, these and other existing manipulators are not without their own drawbacks that can be briefly summarized as: limited workspace, limited tilting capability, presence of singularities, design complexity, the use of transmission systems and articulated platforms impacting accuracy, poor rigidity (e.g. in [5] despite their interesting isotropic property due to not having all the kinematic chains counteracting the load), etc… So, to overcome the aforementioned limitations, the ARROW V1 (Accurate and Rapid Robot with large Operational Workspace Version 1), has been recently introduced in [12] with its inverse and direct geometric models as well as its kinematic and singularity analysis.

In this chapter, we complement [12] by providing the performance evaluation of ARROW V1 relative to dynamics. The performance measure used here is physically significant, comprehensible, and most importantly can be validated experimentally. The chapter recalls the architecture of ARROW V1 in Sect. 2 with some needed models. Then, Sect. 3 presents the performance analysis. The chapter ends in Sect. 4 with conclusions and perspectives for future works.

2 The ARROW V1 Robot

The graph diagram and CAD drawing of the ARROW V1 robot are presented in Fig. 1.[2] As for the main geometric parameters, they are clarified in Fig. 2.

The robot is capable of performing translational motion (along x, y, and z) and rotational motion about z-axis, with the x-motion being independent of the other motions thanks to having all the sliders along the same x-direction. This enlarges the workspace and allows carrying any performance analysis considering only y, z, and θ_z (rotation about z-axis). The robot is actuatedly redundant having 2 extra actuators; this redundancy aids in the elimination of singularities, homogenizing performances, improving accuracy and precision, increasing rigidity and allowing for elevated dynamic capabilities [4]… For more detailed information regarding robot's models or singularity analysis, refer to [12]. However, it is worth recalling that there are no singularities of any type within the geometrically accessible workspace

[1] R: Revolute, U: Universal, P: Prismatic.

[2] In Fig. 1, P and S stand for prismatic and spherical joints resp. Gray boxes stands for actuated joints and underlined for measured position.

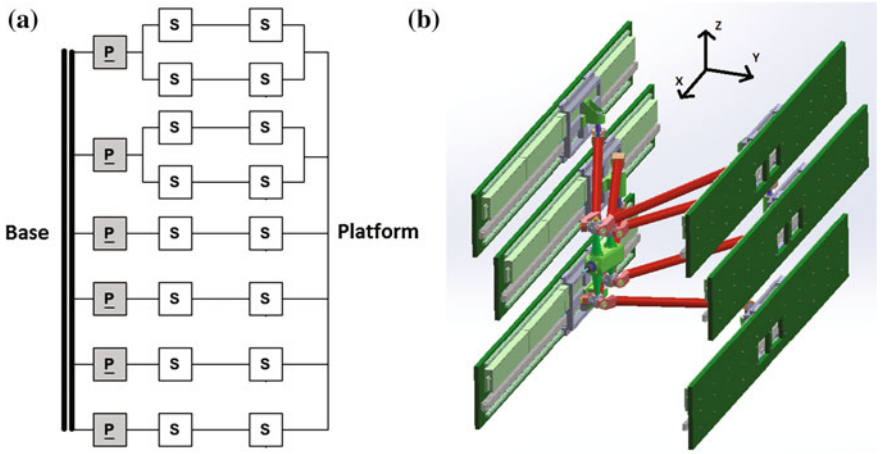


Fig. 1 ARROW V1 robot. **a** Graph diagram. **b** CAD drawing

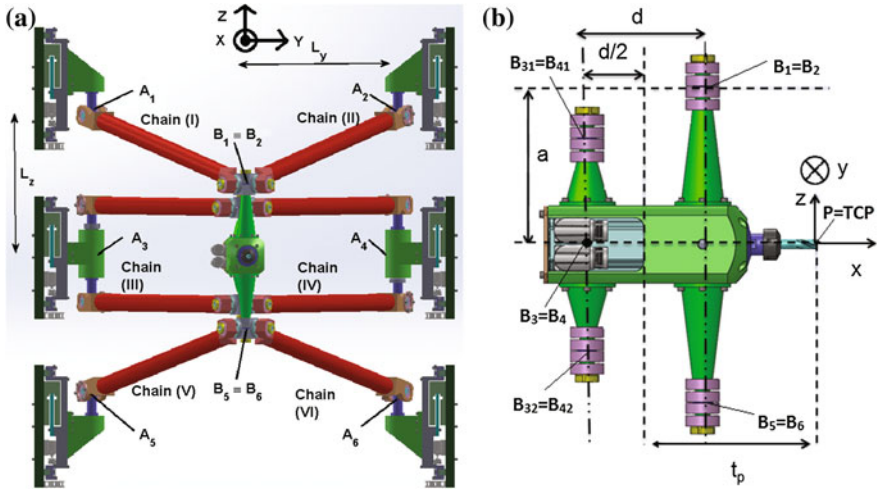


Fig. 2 The main geometric parameters of the robot and the points labelling. **a** Robot front view. **b** Moving platform side view

excluding its boundary (where we might have series-type singularities or parallel-type singularities confounded with series-type ones).

The inverse geometric model (IGM) is given by:

$$q_i = x_{b_i} - \sqrt{L_i^2 - (y_{b_i} - y_i)^2 - (z_{b_i} - z_i)^2} \tag{1}$$

where subscript $i \in \{1 \dots 6\}$ (used along all this chapter). $\boldsymbol{A_i} = (q_i\ y_i\ z_i)^T$ and $\boldsymbol{B_i} = \left(x_{b_i}\ y_{b_i}\ z_{b_i}\right)^T$ are the coordinates of A_i and B_i in the base frame. L_i is the ith arm length (parallelogram arms are of length L_P and simple arms are of length L_S).

Finally, note that knowing the pose $\boldsymbol{x} = \left(\boldsymbol{P}^T \theta_z\right)^T = (x\ y\ z\ \theta_z)^T$ and $\boldsymbol{B}_i^m$ the coordinates of B_i in platform frame (of origin $P \equiv TCP$), we can get $\boldsymbol{B}_i$ via:

$$\boldsymbol{B}_i = \boldsymbol{P}_i + \boldsymbol{R}\boldsymbol{B}_i^m \tag{2}$$

with $\boldsymbol{R} = \boldsymbol{Rot}(\boldsymbol{e_z}, \theta_z)$. The inverse Jacobian matrix $\boldsymbol{J_m}$ (where $\dot{q} = \boldsymbol{J_m}\dot{\boldsymbol{x}}$) is given by:

$$\begin{cases} \boldsymbol{J_m} = \boldsymbol{J_q}^{-1}\boldsymbol{J_x} \\ \boldsymbol{J_q} = \boldsymbol{diag}(\boldsymbol{A}_1\boldsymbol{B}_1^T\boldsymbol{e_x} \cdots \boldsymbol{A}_6\boldsymbol{B}_6^T\boldsymbol{e_x}) \\ \boldsymbol{J_x} = \begin{pmatrix} \boldsymbol{A}_1\boldsymbol{B}_1^T & -\boldsymbol{A}_1\boldsymbol{B}_1^T\ (\boldsymbol{P}\boldsymbol{B}_1 \times \boldsymbol{e}_z) \\ \vdots & \vdots \\ \boldsymbol{A}_6\boldsymbol{B}_6^T & -\boldsymbol{A}_6\boldsymbol{B}_6^T\ (\boldsymbol{P}\boldsymbol{B}_6 \times \boldsymbol{e}_z) \end{pmatrix} \end{cases} \tag{3}$$

Another model needed for our study is the Dynamic Model (DM). Here, we are going to do some simplifications to facilitate the analysis: we neglect both friction and damping and also we estimate the arms motion by considering each arm to be formed of two point masses (each of half-mass of the arm) at its extremities connected by massless rod. In other words, the effect of rod is estimated by dividing its mass in half and putting one half on the corresponding slider, while the other half mass is added to the platform at the corresponding articulation point.

Then, we arrive at the following DM (details are omitted due to space limitation, but can be simply obtained by deriving the equations of motion for the sliders and for the platform. We update the inertia parameters for sliders and platform including the half masses of the arms as previously explained, similar to [4]):

$$\begin{cases} \ddot{\boldsymbol{x}} &= \boldsymbol{H}\boldsymbol{\tau} - \boldsymbol{\Lambda}\dot{\boldsymbol{x}} + \boldsymbol{a}_{\boldsymbol{g}}^{\boldsymbol{offset}} \\ \boldsymbol{H} &= (\boldsymbol{M_P} + \boldsymbol{J}_{\boldsymbol{m}}^T\boldsymbol{M_a}\boldsymbol{J_m})^{-1}\boldsymbol{J}_{\boldsymbol{m}}^T \\ \boldsymbol{\Lambda} &= \boldsymbol{H}\boldsymbol{M_a}\dot{\boldsymbol{J}}_{\boldsymbol{m}} + (\boldsymbol{M_p} + \boldsymbol{J}_{\boldsymbol{m}}^T\boldsymbol{M_a}\boldsymbol{J_m})^{-1}\boldsymbol{\Lambda_c} \\ \boldsymbol{a}_{\boldsymbol{g}}^{\boldsymbol{offset}} &= (\boldsymbol{M_P} + \boldsymbol{J}_{\boldsymbol{m}}^T\boldsymbol{M_a}\boldsymbol{J_m})^{-1}\boldsymbol{M_P}\boldsymbol{g} \end{cases} \tag{4}$$

where $\boldsymbol{M_a} = \boldsymbol{diag}\left(m_{as}\ m_{as}\ m_{ap}\ m_{ap}\ m_{as}\ m_{as}\right)$ is the actuator's inertia matrix, $\boldsymbol{M_P}$ is the platform's inertia matrix and $\boldsymbol{\Lambda}_c$ is the matrix of Coriolis and centrifugal forces. The expressions of the latter two matrices are as follows:

$$\boldsymbol{M_p} = \begin{bmatrix} m_p & 0 & 0 & -bm_p\sin(\theta_z) \\ 0 & m_p & 0 & bm_p\cos(\theta_z) \\ 0 & 0 & m_p & 0 \\ -bm_p\sin(\theta_z) & bm_p\cos(\theta_z) & 0 & I_{pzz} \end{bmatrix}$$

and

$$\boldsymbol{\Lambda_c} = \begin{bmatrix} 0\,0\,0 & -bm_p\ddot{\theta}_z\cos(\theta_z) \\ 0\,0\,0 & bm_p\ddot{\theta}_z\sin(\theta_z) \\ 0\,0\,0 & 0 \\ 0\,0\,0 & 0 \end{bmatrix}.$$

Note that b is the x-component of the center of gravity of the platform's frame (the other y and z components are null) and $\boldsymbol{g} = (0\,0\,-G\,0)^T$ is the gravitational acceleration with $G = 10ms^{-2}$. We do not consider any external loads (except gravity), as we are interested in dynamics in absence of load (transition phase or in case of contactless application such as laser cutting). This is practically valid as in contactless application there would be no external load. As for contact machining phase, the situation is rather quasi-static.

3 Performance Analysis

For dynamic analysis, there are several performance measures for example: dynamic manipulability [13], acceleration radius [6], motion isotropy hypersurface [2] (interested reader may refer to the cited references and therein for more information)… However, each performance measure has its own drawbacks and limitations (especially when it comes to redundant robots), which is not our concern here. Actually, in our case, we have decided to choose a performance measure that has clear significance, and that can be directly validated by experimentation.

Our dynamic study will consider only the dynamic performance starting from rest (i.e. for $\dot{\boldsymbol{x}} = \boldsymbol{0}$) and the isotropic linear acceleration (*ILA*) will be derived at each pose (considering angular acceleration $\ddot{\theta}_z = 0$); meaning that we will calculate the largest value of linear acceleration that can be attained by the robot in all directions assuming $\ddot{\theta}_z = 0$. So, to calculate *ILA*, let us first derive the actuators forces vector in terms of the end-effector acceleration which is given by:

$$\boldsymbol{\tau} = \boldsymbol{C}(\ddot{\boldsymbol{x}} + \boldsymbol{\Lambda}\dot{\boldsymbol{x}} - \boldsymbol{a}_{\boldsymbol{g}}^{offset}) + [\boldsymbol{null}(\boldsymbol{H})]\,\boldsymbol{\sigma} \tag{5}$$

where $\boldsymbol{C} = \boldsymbol{H}^*$ is the pseudo-inverse of $\boldsymbol{H}$, $\boldsymbol{\sigma} \in R^2$ is arbitrary and $|\tau_i| \leq \tau_{max_i}$.

As it is clear from (5), due to actuation redundancy, there is no unique vector $\boldsymbol{\tau}$ for a given end-effector acceleration. However, in our situation, we choose the minimum norm solution, meaning we choose $\boldsymbol{\sigma} = \boldsymbol{0}$ (i.e. $\boldsymbol{\tau}$ is perpendicular to the null space of $\boldsymbol{H}$). This can be obtained practically by control means. Additionally, it is worth emphasizing the fact that $\boldsymbol{null}(\boldsymbol{H}) = \boldsymbol{null}(\boldsymbol{J_m^T})$ since $\boldsymbol{H} = \boldsymbol{S^{-1}J_m^T}$ with $\boldsymbol{S^{-1}} = (\boldsymbol{M_P} + \boldsymbol{J_m^T M_a J_m})^{-1}$ a square non-singular matrix. Finally, based on the above argument and considering the case of zero velocity, Eq. (5) can be rewritten as:

$$\boldsymbol{\tau} + \boldsymbol{C}\boldsymbol{a}_{\boldsymbol{g}}^{offset} = \boldsymbol{C}\ddot{\boldsymbol{x}} \tag{6}$$

Table 1 Geometric parameters (in m)

Parameter	L_s	L_p	L_y	L_z	a	d	t_p	b
Value	0.93	0.93	0.5	0.41	0.164	0.12	0.18	-0.16

Table 2 Inertial parameters (mass in kg and inertias in kgm^2)

Parameter	m_{as}	m_{ap}	m_p	I_{pzz}
Value	8.77	9.45	12.57	0.43

ILA is defined as the largest linear acceleration the robot can achieve regardless of direction and for $\ddot{\theta}_z = 0$, thus it is geometrically the largest sphere included in the linear acceleration zonotope. Then, mathematically we can write:

$$ILA = \min_i \left(\frac{\tau_{max_i}^{new}}{\| \boldsymbol{c}_{pr_i} \|} \right) \tag{7}$$

where $\boldsymbol{c}_{pr_i}$ is the ith row vector of matrix $\boldsymbol{C_p}$ (translational part of $\boldsymbol{C}$), $\tau_{max_i}^{new} = \min\left(\left| -\tau_{max_i} + t_i^{off} \right|, \left| \tau_{max_i} + t_i^{off} \right| \right)$ (with $|\tau_i| \leq \tau_{max_i}$), t_i^{off} is the ith component of $\boldsymbol{t}^{off} = \boldsymbol{C a}_{\boldsymbol{g}}^{off}$ (total offset).

The proof of (7) is very similar to that in [4], except that here we have included gravitation and also we have mixed dofs. Now, that we have clarified *ILA*, we can proceed by presenting the performance analysis for the set of geometric parameters of Table 1.

As for the actuators, they are all identical and with maximal force capacity being $\tau_{max} = 2500$ N (motor type: ETEL Ironless Linear Motor ILM12-060). The inertia parameters are given in Table 2.

The geometric and inertia parameters in Tables 1 and 2 have been chosen in such a way as to allow accessibility to the pre-specified desired workspace centered at the origin with 0.3 m length along y and 0.3 m length along z axes with $\pm 45°$ rotational capability, and based on a dimensional synthesis procedure involving dynamics as one of its criteria in addition to precision related criterion; this dimensional synthesis is not our concern here. The workspace dynamic analysis for $\theta_z = 0°$ and for $-45° \leq \theta_z \leq 45°$ are presented in Figs. 3 and 4 respectively. In these figures, black dotted box is the yz projection of robot's frame, the red-solid box is boundary that should not be exceeded by robot's TCP to avoid colliding with robot's frame, and the magenta coloured box represents the pre-specified desired workspace. It is important to mention that the robot can do $\pm 90°$ rotation, but for our intended application $\pm 45°$ is sufficient. Also, in the case of $-45° \leq \theta_z \leq 45°$ analysis, we have calculated the value of *ILA* for several angles—namely $\pm 45°$, $\pm 30°$, and $0°$ (as they are frequently used angles and to reduce computation time)—and we associated the worst value of *ILA* (smallest value) to the corresponding (y, z) point. The figures show large workspaces and homogenized performance with large isotropic linear acceleration capabilities, in which the robot can achieve $ILA \geq 8.5G$ over the whole desired workspace, which is quite interesting considering the inertia values we are dealing with—having a PKM (parallel kinematic machine tool) as compared

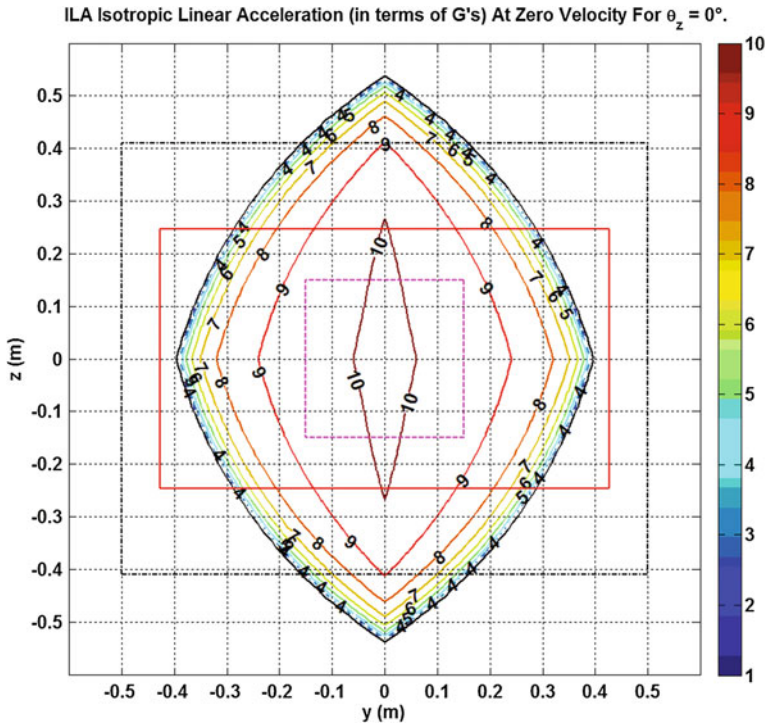


Fig. 3 *ILA* in terms of G's ($G = 10\,\mathrm{ms}^{-2}$) in case of zero rotation

to light-weight robots for pick-and-place applications. Moreover, we have calculated peak linear acceleration *PLA* (the largest linear acceleration attainable regardless of angular acceleration $\ddot{\theta}_z$) starting from rest (i.e. $\dot{\boldsymbol{x}} = \mathbf{0}$) defined mathematically by:

$$PLA = \max_{\tau} \left(\sqrt{\ddot{x}^2 + \ddot{y}^2 + \ddot{z}^2} \right) \tag{8}$$

with $[\boldsymbol{null}(\boldsymbol{H})]^T \boldsymbol{\tau} = \mathbf{0}$ (i.e. based on minimum norm solution for $\boldsymbol{\tau}$), and $|\tau_i| \leq \tau_{max_i}$. We have noticed that *PLA* ranges between 22.5 and 24.5Gs over the desired workspace, which is quite intriguing (note that the detailed plots for peak linear acceleration as function of y, z, and θ_z are not shown due to space limitation)… Hence, based on the results presented above, the PKM at hand is very promising regarding the expected dynamic performance.

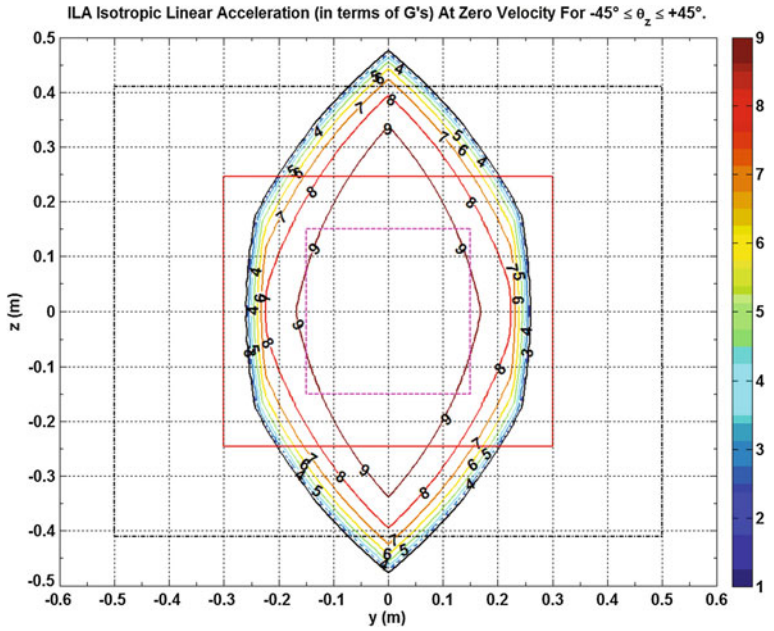


Fig. 4 *ILA* in terms of G's ($G = 10\,\mathrm{ms}^{-2}$) in case of rotation between $\pm 45°$

4 Conclusion and Perspectives

In brief, in this chapter we have recalled a recently introduced robot in [12], and presented its dynamic performance; the results of which are quite interesting. In fact, the robot in addition to having the capability to achieve 8.5Gs in all directions over the desired workspace, its performance is rather homogeneous.

Moreover, peak linear acceleration *PLA* on the desired workspace is undoubtedly intriguing as it is expected to be between 22.5 and 24.5Gs over the desired workspace; all of this while having large inertia dealing with a machine tool as opposed to light-weight pick-and-place parallel manipulators. Regarding future work, the dimensional synthesis, precision related performance and other detailed kinetostatic performances have been done for this PKM and are very promising; these might be the subject for future publication.

Acknowledgments This work has been supported partially by the French National Research Agency within the ARROW project (ANR 2011 BS3 006 01) and by Tecnalia France.

References

1. Angeles, J., Caro, S., Khan, W., Morozov, A.: Kinetostatic design of an innovative schn++nflies-motion generator. Proc. Inst. Mech. Eng., Part C: J. Mech. Eng. Sci. **220**(7), 935–943 (2006). doi:10.1243/09544062JMES258
2. Bowling, A., Khatib, O.: The motion isotropy hypersurface: a characterization of acceleration. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, pp. 965–971. Victoria, BC (1998)
3. Clavel, R.: Une nouvelle structure de manipulateur parallèle pour la robotique légère. Automatique-productique informatique industrielle **23**(6), 501–519 (1989)
4. Corbel, D., Gouttefarde, M., Company, O., Pierrot, F.: Towards 100G with PKM. Is actuation redundancy a good solution for pick-and-place? In: ICRA'2010: International Conference on Robotics and Automation, pp. 4675–4682. IEEE, Anchorage, Alaska, États-Unis (2010). doi:10.1109/ROBOT.2010.5509921
5. Gogu, G.: Structural synthesis of fully-isotropic parallel robots with schonflies motions via theory of linear transformations and evolutionary morphology. Eur. J. Mech. A. Solids **26**(2), 242–269 (2007)
6. Graettinger, T.J., Krogh, B.H.: The acceleration radius: a global performance measure for robotic manipulators. IEEE J Robot. Autom. **4**(1), 60–69 (1988)
7. Krut, S., Company, O., Benoit, M., Ota, H., Pierrot, F.: I4: A new parallel mechanism for SCARA motions. In: Proceedings of IEEE ICRA: International Conference on Robotics and Automation, pp. 1875–1880. Taipei, Taiwan (2003)
8. Nabat, V., Company, O., Krut, S., Rodriguez, M., Pierrot, F.: Par4: very high speed parallel robot for pick and place. In: IROS'05: International Conference on Intelligent Robots & Systems. Edmonton, Alberta (2005)
9. Pierrot, F., Company, O.: H4: a new family of 4-dof parallel robots. In: AIM'99: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 508–513. Atlanta, Georgia (1999)
10. Pierrot, F., Nabat, V., Company, O., Krut, S.: From Par4 to adept quattro. In: Robotic Systems for Handling and Assembly—3rd International Colloquium of the Collaborative Research Center SFB 562, pp. 207–220. Shaker Verlag, Braunschweig (2008)
11. Rolland, L.: The Manta and the Kanuk: novel 4-DOF parallel mechanisms for industrial handling. In: D.S.C.D. Nejat Olgac (ed.) International Mechanical Engineering Congress & Exposition, vol. 67, pp. 831–844. American Society of Mechanical Engineers, ASME, Nashville (1999)
12. Shayya, S., Krut, S., Company, O., Baradat, C., Pierrot, F.: A novel (3T–1R) redundant parallel mechanism with large operational workspace and rotational capability. In: IROS'2013: International Conference on Intelligent Robots and Systems, pp. 436–443. IEEE/RSJ, Tokyo (2013)
13. Yoshikawa, T.: Dynamic manipulability of robot manipulators. J. Rob. Syst. **2**(1), 113–124 (1985)

Author Index

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1,

Subject Index

J. Lenarčič and O. Khatib (eds.), *Advances in Robot Kinematics*,
DOI: 10.1007/978-3-319-06698-1,