Jadran Lenarčič
Philippe Wenger
Editors

Advances in Robot Kinematics

Analysis and Design

Springer

Advances in Robot Kinematics: Analysis and Design

Jadran Lenarčič • Philippe Wenger

Editors

Advances in Robot Kinematics: Analysis and Design

Springer

Jadran Lenarčič
J. Stefan Institute
Ljubljana, Slovenia

Philippe Wenger
IRCCyN Institute
Nantes, France

ISBN-13: 978-1-4020-8599-4 e-ISBN-13: 978-1-4020-8600-7

Library of Congress Control Number: 2008927513

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Table of Contents

Preface xi

SINGULARITY ANALYSIS OF PARALLEL MANIPULATORS

A New Assessment of Singularities of Parallel Kinematic Chains 3
Michele Conconi and Marco Carricato

Singularity Analysis through Static Analysis 13
J. Hubert and J.-P. Merlet

Constraint Singularities and the Structural Parameters of Parallel Robots 21
Grigore Gogu

Forward Kinematics and Singularity Analysis of a 3-RPR Planar Parallel Manipulator 29
Xianwen Kong

Geometric Algebra Approach to Singularity of Parallel Manipulators with Limited Mobility 39
Tanio K. Tanev

SinguLab – A Graphical User Interface for the Singularity Analysis of Parallel Robots Based on Grassmann–Cayley Algebra 49
Patricia Ben-Horin, Moshe Shoham, Stéphane Caro, Damien Chablat and Philippe Wenger

Singularity Analysis of Limited-DOF Parallel Manipulators Using Grassmann–Cayley Algebra 59
Daniel Kanaan, Philippe Wenger and Damien Chablat

DESIGN OF ROBOTS AND MECHANISMS

On the Design of Fully Constrained Parallel Cable-Driven Robots 71
M. Gouttefarde, S. Krut, O. Company, F. Pierot and N. Ramdani

Synthesis of Part Orienting Devices for Spatial Assembly Tasks 79
P. Larochelle

Minimum Energy Manipulator Design 89
A. Rojas Salgado and Y. Ledezman Rubio

Synthesis and Analysis of a Constrained Spherical Parallel Manipulator 101
G.S. Soh and J.M. McCarthy

A Nonholonomic 3-DOF Parallel Robot 111
Patricia Ben-Horin and Federico Thomas

4DOF Parallel Architecture for Laparoscopic Surgery 119
Mohammad Aamir Khan, Matteo Zoppi and Rezio Molfino

Flapping-Wing Mechanism for a Bird-Sized UAVs: Design, Modeling and Control 127
Ch. Grand, P. Martinelli, J.-B. Mouret and S. Doncieux

Extended Jacobian Inverse Kinematics and Approximation of Distributions 137
Mariusz Janiak and Krzysztof Tchoń

A Screw Syzygy with Applications to Robot Singularity Computation 147
J.M. Selig and Peter Donelan

Singularity Robust Jacobian Inverse Kinematics for Mobile Manipulators 155
Krzysztof Tchoń and Łukasz Małek

Robots Based on Assur Group A (3.5) 165
Karl Wohlart

Kinematics of Free-Floating Systems through Optimal Control Theory 177
G. Le Vey

Genericity Conditions for Serial Manipulators 185
Peter Donelan

Alternative Forms for Displacement Screws and Their Pitches 193
I.A. Parkin

MOTION PLANNING AND MOBILITY

Time-Invariant Strategies in Coordination of Human Reaching 205
Satyajit Ambike and James P. Schmiedeler

Mobility and Higher Order Local Analysis of the Configuration Space of Single-Loop Mechanisms 215
A. Müller and J.M. Rico

Optimization of a Test Trajectory for SCARA Systems 225
J.-F. Gauthier, J. Angeles and S. Nokleby

Singularity Free Path Planning for Parallel Robots 235
Samir Lahouar, Saïd Zeghloul and Lotfi Romdhane

A Comparison between Two Motion Planning Strategies for Kinematically Redundant Parallel Manipulators 243
J.A. Carretero, I. Ebrahimi and R. Boudreau

Trajectory Planning of Parallel Manipulators for Global Performance Optimization 253
Ofelia G. Alba-Gómez, J. Alfonso Pamanes and Philippe Wenger

Human Motion Reconstruction by Direct Control of Marker Trajectories 263
Emel Demircan, Luis Sentis, Vincent De Sapio and Oussama Khatib

PERFORMANCE AND PROPERTIES OF MECHANISMS

New Self-Motions of Parallel Manipulators 275
Adolf Karger

Exact Envelope Computation for Moving Surfaces with Quadratic Support Functions 283
Margot Rabl, Bert Jüttler and Laureano Gonzalez-Vega

A Compound-Structure Frame for Improving the Performance of a Dielectric Elastomer Actuator 291
Giovanni Berselli, Rocco Vertechy, Gabriele Vassura and Vincenzo Parenti Castelli

Transitions between Multiple Solutions of the Direct Kinematic Problem 301
E. Macho, O. Altuzarra, C. Pinto and A. Hernandez

Kinetostatic Performance of a Planar Parallel Mechanism with Variable Actuation 311
N. Rakotomanga, D. Chablat and S. Caro

Results on Planar Parallel Manipulators with Cylindrical Singularity Surface 321
G. Nawratil

Stiffness Matrix of Compliant Parallel Mechanisms 331
C. Quennouelle and C.M. Gosselin

MEASURE AND CALIBRATION

A Pair of Measures of Rotational Error for Axisymmetric Robot End-Effectors 345
Sébastien Briot and Ilian A. Bonev

Angular-Velocity Estimation from the Centripetal Component of the Rigid-Body Acceleration Field 353
Philippe Cardou and Jorge Angeles

A Novel Point of View to Define the Distance between Two Rigid-Body Poses 361
Raffaele Di Gregorio

Parallel Robot Calibration by Working Mode Change 371
P. Last, C. Budde, D. Schütz, J. Hesselbach and A. Raatz

Augmented Model of the 3-PRS Manipulator for Kinematic Calibration 381
Steven M. O'Brien and Juan A. Carretero

The Calibration of a Parallel Manipulator with Binary Actuation 391
Jean-Sébastien Plante and Steven Dubowsky

Stability Measure of Postural Dynamic Equilibrium Based on Residual Radius 399
Sébastien Barthélemy and Philippe Bidaud

KINEMATIC ANALYSIS AND WORKSPACE

A Geometrical Characterization of Workspace Singularities in 3R Manipulators 411
M. Husty, E. Ottaviano and M. Ceccarelli

Kinematic Analysis of a Planar Tensegrity Mechanism with Pre-Stressed Springs 419
Carl D. Crane III, Jahan Bayat, Vishesh Vikas and Rodney Roberts

Inverse Kinematics of Robot Manipulators with Multiple Moving Control Points 429
Agostino De Santis and Bruno Siciliano

On the Delassus Parallelogram 439
Chung-Ching Lee and Jacques M. Hervé

Forward Kinematics and Workspace Determination of a Wire Robot for Industrial Applications 451
Andreas Pott

Kinematic Modeling and Workspace Generation for a New Paralell Robot Used in Minimally Invasive Surgery 459
Doina Pisla, Nicolae Plitea and Calin Vaida

Author Index 469

Subject Index 471

Preface

Springer published the first book of the series of Advances in Robot Kinematics in an edited format in 1991. Since 1994, Kluwer and Springer published a book every two years without interruptions. These books deal with the theory and practice of robot kinematics and treat the kinematic aspects of robot motion and design of robots. Each book of Advances in Robot Kinematics reports the most recent research projects and presents important new discoveries. The series of Advances in Robot Kinematics is considered a most important source of information in its area.

The present book emphasizes kinematic analysis and design. The issues addressed are fundamentally kinematic in nature, including synthesis, calibration, redundancy, force control, dexterity, inverse and forward kinematics, kinematic singularities, as well as over-constrained systems. Methods used include line geometry, quaternion algebra, screw algebra, and linear algebra. These methods are applied to both parallel and serial multi-degree-of-freedom systems. The results should interest researchers, teachers and students, in fields of engineering and mathematics related to robot theory, design, control and application.

The contributions had been rigorously reviewed by independent reviewers. The authors discussed their results at the eleventh international symposium on Advances in Robot Kinematics which was held in June 2008 in Batz-sur-Mer, France. The symposium was organized by the Institut de Recherche en Communications et Cybernetique de Nantes, France in collaboration with the J. Stefan Institute, Ljubljana, Slovenia, under the patronage of the International Federation for the Promotion of Mechanism and Machine Science.

We are grateful to the authors for their contributions and to the reviewers for their timely reviews and recommendations. We are also indebted to the personnel at Springer and Jolanda Karada (Karada Publishing Services) for their excellent technical and editorial support.

Jadran Lenarčič and Philippe Wenger, editors

SINGULARITY ANALYSIS OF PARALLEL MANIPULATORS

A New Assessment of Singularities of Parallel Kinematic Chains

Michele Conconi and Marco Carricato

DIEM – Department of Mechanical Engineering, University of Bologna, Italy;
e-mail: {michele.conconi,marco.carricato}@mail.ing.unibo.it

Abstract. An exhaustive hierarchical-level-based classification of singularities of parallel kinematic chains is presented. Singular events are identified, interpreted and related to the direct physical phenomena originating them. An in-depth study of the interaction of different types of singularity is performed.

Key words: singularities, instantaneous kinematics, parallel mechanisms.

1 Introduction

Singularities are critical configurations in which the kinetostatic behaviour of a mechanism suddenly changes with respect to a full-cycle condition. They have been studied, under different perspectives, by several authors. Just a few contributions are recalled here, as a comprehensive discussion is beyond the scope of this paper (for more detailed bibliographic records, the reader may refer, for instance, to [8]).

Singular configurations are commonly identified by analyzing the *input-output* kinematic equations of a mechanism [3]. However, it has been shown that such an approach may fail to detect some critical configurations of closed-loop chains. Zlatanov et al. [11], in particular, recognized the role of passive joints, emphasizing the necessity to develop the study of singularities in the entire configuration space of the mechanism. They consider the mechanism as an input-output device and base their study on a definition of singularities that requires the selection of a set of actuated variables. This choice is necessary for the sake of completeness, but it inevitably renders the classification complex and, in some cases, of non-immediate or ambiguous application.

Hunt [4] pioneered the idea of independently considering the phenomena solely depending on the mechanism geometry, introducing the concepts of uncertainty and stationary configurations for single-loop chains. Park and Kim [9] and Liu et al. [7] developed a more complete level-based analysis. They showed that critical configurations are coordinate-invariant and proposed, for closed-loop chains, a hierarch-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 3–12.

ical classification that distinguishes between the effects caused by the geometric arrangement of joints, the choice of the actuation and the output-frame location. Their approach, based on the tools of differential geometry, does not, however, provide an easy insight into the physical causes of singular phenomena.

Joshi and Tsai [5], in turn, developed an easily-interpretable screw-theory-based Jacobian analysis of parallel mechanisms. Such an analysis is able to express the distinctive contributions of both the passive and active constraints acting between the fixed base and the end-effector, but it does not take into account the entirety of singular events caused by passive joints.

The aim of this study is to merge the advantages of the above approaches in order to obtain a singularity classification that (i) proves detailed and exhaustive; (ii) recognizes hierarchical levels in which physical causes of different phenomena may be distinguished and more easily interpreted; (iii) employs analytical tools that make the physical comprehension of singularities immediate. Particular attention is reserved for the interaction of singularities of different types, an aspect that has seldom received attention in the literature. Indeed, different critical events may simultaneously take place, either compensating or amplifying each other or even originating new phenomena.

For the sake of simplicity, the paper is focused on fully-parallel kinematic chains only. They are composed of a base, a platform and a number of serial chains connecting them. A configuration is addressed as *full-cycle* if a finite neighborhood thereof exists in which instantaneous motions may occur that do alter neither the connectivity of the platform with respect to the base nor the dimension of the screw system associated with any subset of pairs pertaining to a same leg. A full-cycle condition represents a reference state with respect to which singular configurations are identified. The full-cycle condition of a given kinematic chain is not necessarily unique [2] and the problem of its identification is not addressed here. Full-cycle quantities are overlined by a tilde accent throughout the text. As joints with multiple degrees of freedoms (dof) may always be represented by convenient (instantaneous) ensembles of individual freedoms, only single-dof joints are considered, without loss of generality.

The paper is organized as follows. Section 2 presents the singularities of an *unactuated kinematic chain* (UKC). At this level, no link is a priori elected as a frame or end-effector and no actuation scheme is chosen. In Section 3 the *actuated kinematic chain* (AKC) is considered, its peculiar singularities are introduced and their interactions with the ones defined at the first level are analyzed. Section 4 briefly discusses the possible consequences of singularities on a task-dedicated mechanism, namely a device in which frame and output links have been chosen. For the sake of the reader's convenience, two tables are provided at the end of the paper, summarizing the obtained results. Finally, conclusions are drawn.

2 Level 1: Unactuated-Kinematic-Chain Singularities

At this level, no link is selected as a frame or end-effector and no joint is actuated. As a consequence, kinetostatic properties only depend on the kind, number and mutual disposition of the joints composing the UKC.

Two members, called base (member b) and platform (member p), are connected by a number l of serial chains, named legs. A superscript i denotes quantities referring to the ith leg. $\dot{q}^i_j \$^i_j$, $j = 1, \ldots, f^i$, is the twist associated with the jth pair of such a leg, f^i being the total number of joints in it. $\mathbf{J}^i = [\$^i_1, \ldots, \$^i_{fi}]$ and $\dot{\mathbf{q}}^i = [\dot{q}^i_1, \ldots, \dot{q}^i_{fi}]^T$ are the Jacobian matrix and the joint-velocity array of the leg, respectively. $S^i = \text{span}(\$^i_1, \ldots, \$^i_{fi})$ is the subspace spanned in $\mathbb{R}^3 \times \mathbb{R}^3$ by the leg screws and r^i is its dimension. A leg is said to be *redundant* if, in a full-cycle configuration, it is $f^i > \tilde{r}^i$. The difference $m^i = f^i - r^i$ represents the internal mobility of the leg, to wit the number of velocity variables that needs to be assigned within the leg in order to completely determine its velocity state while p and b are kept fixed. C^i is the subspace of the wrenches reciprocal to the leg screws, namely the *passive constraints or reactions* that may be transmitted between b and p. $c^i = 6 - r^i$ is the dimension of C^i. The total number of joints in the KC is $F = \sum_{i=1}^{l} f^i$.

Quantities referring to the relative movement between b and p (in brief, the pb motion) are denoted by a superscript pb. $S^{pb} = \bigcap_{i=1}^{l} S^i$ is the subspace of the admissible pb twists $\mathbf{t}^{pb}$ and r^{pb} is its dimension. $C^{pb} = \sum_{i=1}^{l} C^i$ is the subspace, of dimension $c^{pb} = 6 - r^{pb}$, of the reactions reciprocal to the screws of S^{pb}.

2.1 Instantaneous Mobility

As $\mathbf{t}^{pb}$ may be expressed as a linear combination of the joint twists in each leg, namely $\mathbf{t}^{pb} = \sum_{i=1}^{f^i} \dot{q}^i_j \$^i_j = \mathbf{J}^i \dot{\mathbf{q}}^i$, $i = 1, \ldots, l$, the first-order-kinematics loop-closure equations of the KC may be expressed as

$$\mathbf{L}\dot{\mathbf{q}} = \underbrace{\begin{bmatrix} -\mathbf{J}^1 \\ -\mathbf{J}^1 \\ \vdots \\ -\mathbf{J}^1 \end{bmatrix}}_{\mathbf{L}_1} \underbrace{\begin{bmatrix} \mathbf{J}^2 & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{J}^3 & \ldots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{J}^l \end{bmatrix}}_{\mathbf{L}_{2l}} \begin{bmatrix} \dot{\mathbf{q}}^1 \\ \dot{\mathbf{q}}^2 \\ \vdots \\ \dot{\mathbf{q}}^l \end{bmatrix} = \mathbf{0}, \tag{1}$$

where $\mathbf{L}$ is a $6(l-1) \times F$ matrix and $\dot{\mathbf{q}}$ is an F-dimensional vector.

As $\mathbf{L}_{2l}$ is a block diagonal matrix, a column of $\mathbf{L}_1$ linearly depends on other columns of $\mathbf{L}$ only if the corresponding column of $\mathbf{J}^1$ may be obtained as a linear combination of the columns of each one of the matrices $\mathbf{J}^i$, for $i = 2, \ldots, l$. The number of linearly dependent columns comprised in $\mathbf{L}_1$ amounts, thus, to the di-

mension of the intersection of the column spaces of all $\mathbf{J}^i$, i.e. $\dim\left(\bigcap_{i=1}^{l} S^i\right) = \dim(S^{pb}) = r^{pb}$. It immediately follows that

$$\operatorname{rank}(\mathbf{L}) = \sum_{i=1}^{l} r^i - r^{pb} = R - r^{pb}, \tag{2}$$

where $R = \sum_{i=1}^{l} r^i$.

The instantaneous mobility of the KC, to wit the number of first-order kinematic variables that must be specified in order to completely determine the KC velocity state, may thus be calculated as

$$m = F - \operatorname{rank}(\mathbf{L}) = \sum_{i=1}^{l} f^i - \sum_{i=1}^{l} r^i + r^{pb} = \sum_{i=1}^{l} m^i + r^{pb}. \tag{3}$$

Thus, m is the sum of the internal mobilities of all legs plus the connectivity of p with respect to b [6].

In the following, the singularities characteristic of the UKC are introduced and their effects on the KC mobility are analyzed. Numerical information is derived by computing the rank of $\mathbf{L}$ and its submatrices (for an exhaustive singularity analysis, such computing is a necessary step, even though, in many instances, it may be performed by geometric inspection rather than numeric calculation [4, 10]).

2.2 Serial Singularities

A *serial singularity* (SS) occurs in a leg any time n screws of a previously linearly independent subset s_n^i become linearly dependent. As a consequence, the rank r_n^i of s_n^i becomes lower than n and $\infty^{n-r_n^i}$ non-trivial combinations of relative motions not affecting the kinematic state of the other joints in the chain become possible among the joints of the subset. We call this phenomenon *instantaneous localized mobility* (ILM). As an effect of the localization of mobility, a number $n - r_n^i$ of velocity parameters needs to be specified among the joints sharing the SS, in order to determine the complete kinematic state of the KC.

An SS which causes the leg screw system to lose its full-cycle dimension, to wit $r^i < \tilde{r}^i$, is named a *leg singularity* (LS). A necessary and sufficient condition for LSs to occur is $R < \tilde{R}$, their number being $N_{LS} = \Delta R = \tilde{R} - R$. It is worth remarking that not every SS causes an LS. Indeed, in the case of redundant leg, an SS may occur in a subset of screws that already exhibits a full-cycle linear dependency (a so-called *redundant set*), so that the dimensions of the screw systems associated with both the redundant set and the overall leg do not vary. For instance, in a leg comprising a redundant set consisting of four revolute pairs with parallel axes, an SS may occur that causes three of such joints to become coplanar. Nonetheless, as the dimension of the screw system associated with the overall redundant set is still equal to three, r^i remains unaltered.

Due to an LS, c^i increases as much as r^i lessens. The leg gains an internal mobility and exerts a further constraint between b and p.

When the new reactions are linearly independent from the others, r^{pb} decreases as much as r^i: as a consequence, the overall mobility does not vary (cf. Eq. (3)). Following Hunt [4], we call a configuration for which $\Delta r = r^{pb} - \tilde{r}^{pb} < 0$ a *stationary configuration* (SC) and we address the quantity $N_{SC} = \max(0, -\Delta r)$ as its order. This is also the number of LSs leading to an SC ($N_{LS\text{-}SC} = N_{SC}$).

On the contrary, when the new reactions are linearly dependent on the others, r^{pb} does not change and the mobility of the mechanism necessarily increases. Following Zlatanov et al. [11], we say that the KC exhibits an *instantaneous increased mobility* (IIM). The number of LSs leading to an IIM is $N_{LS\text{-}IIM} = N_{LS} - N_{LS\text{-}SC} = \Delta R - N_{SC}$.

2.3 Constraint Singularities

A *passive-constraint singularity* or simply *constraint singularity* (CS) occurs when the dimension of the pb motion space increases with respect to its full-cycle value, i.e. $r^{pb} > \tilde{r}^{pb}$ (and hence $c^{pb} < \tilde{c}^{pb}$). The mobility criterion (3) shows that a CS necessarily leads to an IIM configuration. While in an IIM due to an LS the new freedom is localized in a leg, at a CS it regards the pb motion and, potentially, it affects the overall movement of the KC. The quantity $N_{CS} = \max(0, \Delta r)$ is addressed as the order of the CS. The overall increase of the instantaneous mobility, namely $\Delta m = m - \tilde{m} = \Delta R + \Delta r$, is of course equal to $N_{IIM} = N_{LS\text{-}IIM} + N_{CS}$.

The CS definition adopted here is strictly related to the dimension of the constraint subspace C^{pb} and not to the *type* of the exerted reactions. Indeed, the original definition by Zlatanov et al. [10] is more ambiguous under this perspective, as it seems to relate the occurrence of a CS to the acquisition of a new relative freedom between b and p, a different one with respect to those previously available. An example may clarify this. In a 3-UPU translational parallel manipulator, for instance, every leg exerts a pure reaction torque between b and p and the pb motion is purely translational [5]. If, in a certain configuration, the three reaction torques become linearly dependent and, at the same time, a leg reaches an LS applying a new force between b and p, the four reaction wrenches still span a three-dimensional subspace and $r^{pb} = \tilde{r}^{pb} = 3$. Even if the nature of the pb motion changes (it is no longer translational [1]) and an IIM occurs, the KC is not at a CS, according to our interpretation (it is, indeed, $N_{SC} = N_{CS} = 0$ and $N_{LS} = N_{LS\text{-}IIM} = 1$).

According to our definitions, SCs and CSs are mutually exclusive.

3 Level 2: Actuated-Kinematic-Chain Singularities

At this level, motorized joints are specified. Actuation is effectual only if it allows the complete control of the KC, i.e. if the KC is transformed into a structure once

the actuators are locked. The velocity equations on which the kinematic study of this level is based are obtained from the system (1) by eliminating the screws associated with the actuated joints. By choosing and locking the entire set of motors, a KC with *locked actuation* (KC-LA) is generated. Instead, by choosing and locking only the actuators comprised in the redundant sets (if any), an UKC with *locked redundancy* (UKC-LR) is obtained. All quantities referring to a KC-LA are underlined, whereas quantities referring to the UKC-LR are denoted by the subscript LR.

By virtue of the above reasoning, a good choice of the actuation scheme must then guarantee $\underline{\tilde{m}} = 0$, i.e. $\underline{\tilde{m}}^i = 0$, for $i = 1, \ldots, l$, and $\underline{\tilde{r}}^{pb} = 0$. The former condition requires the choice of $\tilde{m}^i$ actuators among the joints of the redundant set of the ith leg (if present), the latter the selection of further $\tilde{r}^{pb}$ motors to control the pb motion in order to assure $\underline{\tilde{c}}^{pb} = 6$. The total number of actuators must hence be equal to $\tilde{m}$ (full-cycle actuation redundancy is excluded here).

By locking all the actuators in a leg, new wrenches become available that constrain the pb motion. If $\underline{C}^i$ is the space of all screws reciprocal to the passive joints of the ith leg ($\underline{c}^i = \dim(\underline{C}^i)$), C^i is necessarily a subspace of it. As a consequence, there exists a subspace A^i whose generators are reciprocal to the screws of C^i and so that $\underline{C}^i = C^i \oplus A^i$. A^i is said to be the subspace of the *actions* exerted by the ith leg, its dimension a^i being equal to the number of motors contributing to control the pb motion in the ith leg. $A^{pb} = \sum_{i=1}^{l} A^i$ is hence the subspace of the overall *actions* between b and p, a^{pb} being its dimension. Evidently, it is $\underline{c}^{pb} = c^{pb} + a^{pb} - c^{\cap}$, $c^{\cap}$ being the dimension of $C^{pb} \cap A^{pb}$.

A good choice of actuation must guarantee $\underline{\tilde{c}}^{pb} = \tilde{c}^{pb} + \tilde{a}^{pb} - \tilde{c}^{\cap} = 6$ and hence $\tilde{a}^{pb} = \tilde{r}^{pb} + \tilde{c}^{\cap}$. Since, in any case, it must also be $\tilde{a}^{pb} \leq \sum_{i=1}^{l} \dim(A^i) = \tilde{r}^{pb}$, it follows that the actuation is well chosen if and only if $\tilde{c}^{\cap} = 0$ (and hence $\tilde{a}^{pb} = \tilde{r}^{pb}$).

It is clear that any increase of mobility undergone by the UKC cannot be compensated by actuation. On the contrary, this may cause further labilities due to its instantaneous ineffectiveness. Globally, actuation may be considered singular if $\underline{m} > \Delta m = N_{IIM}$. However, as any freedom present in the KC-LA is an uncontrolled one, it is particularly useful to provide a further distinction between labilities localized in the legs, which simply affect the subset of joints sharing an SS, and those regarding the pb motion, which potentially affect the overall movement of the KC (including its legs). The former freedoms, associated with *localized losses of control*, are caused by *actuated-leg singularities* (ALSs) and occur if $\Delta\underline{R} = \underline{\tilde{R}} - \underline{R} > 0$. The latter, associated with a *global loss of control*, are due to *pb-motion singularities* (PBSs) and occur when $\underline{r}^{pb} > 0$. The overall number of *uncontrolled freedoms* (UF) in the KC-LA is, accordingly, $N_{UF} = N_{ALS} + N_{PBS} = \Delta\underline{R} + \underline{r}^{pb}$.

ALSs and PBSs depend on singular phenomena distinctive of both the UKC and the KC-LA. In the following, the effects of the UKC singularities on the KC-LA are assessed and singularities characteristic of the KC-LA (namely, *redundant-set singularities* and *action singularities*) are introduced. The *instantaneous redundancy of actuators* (IAR) is also evaluated. This occurs when the number of actions exerted by the motors among b and p exceeds the dimension of the motion space that they instantaneously control, namely when an action linearly depends on other

constraints, either passive or active. In such occurrences, actuators may work either against each other or against joint constraints, causing overheating or breakdown.

3.1 Redundant-Set Singularities

When an SS is not an LS for the UKC, it necessarily regards a group of joints comprised in a set that is redundant for full-cycle motions of the leg. Such a set thus exhibits an internal ILM. If a proper number of joints sharing the SS are actuated (a number equal to the order of the singularity), the redundant set remains controllable and nothing special happens. Conversely, if an insufficient number of actuators among those reserved for the control of the redundancy are included in the joints sharing the SS, the KC-LA (as well as the UKC-LR) exhibits an LS not present in the UKC. It is an ALS and we call it a *redundant-set singularity* (RSS). In such a condition, an actuator originally intended to control the redundant set of joints exerts a further action between b and p. Depending on whether such an action belongs to C^{pb} or not, it does not or it does contribute to reducing the dimension of the pb motion space. In the former case it acts similarly to an LS causing an IIM, in the latter as an LS leading to an SC. By analogy, we address the RSS as RSS-IIM or RSS-SC respectively. While the action generated by an RSS-SC contributes to A^{pb} and cooperates in controlling the pb motion (bringing about the possibility for an IAR to occur), the action produced by an RSS-IIM belongs to C^{pb} and yields no effects to this aim (however, since it lets $c^{\cap} \neq 0$, it necessarily causes an IAR).

It is worth emphasizing that RSS-IIMs and RSS-SCs are not physically related to the SCs and IIMs defined in Section 2.2. The latter phenomena, in fact, refer to the UKC, which is not affected by RSSs. RSS-IIMs and RSS-SCs are conceptually similar phenomena, but appearing in the UKC-LR. Indeed, the number of RSSs, RSS-SCs and RSS-IIMs is given by, respectively, $N_{RSS} = R - R_{LR}$, $N_{RSS\text{-}SC} = r^{pb} - r^{pb}_{LR}$ and $N_{RSS\text{-}IIM} = N_{RSS} - N_{RSS\text{-}SC}$. N_{RSS} may also be evaluated by simply looking at the rank of the screw subspaces associated with the actuated redundant sets.

3.2 Constraint Singularities and Action Singularities

A PBS occurs when $\underline{r}^{pb} > 0$ and it is caused by a deficiency of the global constraints restraining the relative motion between b and p. While a CS represents the shortcoming of passive constraints (reactions), an *action singularity* or *active-constraint singularity* (AS) expresses the instantaneous inadequacy of the active ones (actions). A PBS is always associated with a global loss of control.

If $\Delta r < 0$, i.e. the UKC lies in an SC, or $\Delta r = 0$, no increase of the pb-motion-space dimension is owed to reactions and a value of $\underline{r}^{pb}$ greater than zero may only be due to insufficient independent actions. In this case, an AS occurs any time $\underline{r}^{pb} > 0$.

If $\Delta r > 0$, namely the UKC exhibits a CS, the rise of $\underline{r}^{pb}$ may depend on both a CS and an AS. If there are no RSSs, any lability caused by a CS is still present at level two, while any further increment of $\underline{r}^{pb}$ is due to actuation. As a consequence, it is necessarily $\underline{r}^{pb} \geq \Delta r$ and an AS occurs only when the strict inequality holds. Conversely, if RSSs are present, these may compensate a CS and it may happen that $\underline{r}^{pb} < \Delta r$. In such a case, actuation produces no detrimental effect on the control of the pb motion (on the contrary, it mitigates the effects of the reaction linear dependency) and no AS is present. The order N_{AS} of the AS is then given by $\underline{r}^{pb}$ if $\Delta r \leq 0$, by $\underline{r}^{pb} - \Delta r$ if $0 < \Delta r \leq \underline{r}^{pb}$ and by 0 if $\Delta r > \underline{r}^{pb}$.

3.3 Leg Singularities

Depending on actuator location, an LS of the UKC may or may not be present in the KC-LA. We refer to the former as an *uncontrolled* LS (ULS) and to the latter as a *controlled* LS (CLS).

If there is an LS in the UKC not present in the KC-LA (CLS, $\underline{r}^i = \tilde{\underline{r}}^i$), some actuator that normally controls the pb motion instantaneously instead controls the localized freedom produced by the LS, being no longer available for the original task. If the LS determines an SC for the UKC ($\Delta r < 0$), however, the KC remains completely controllable, as the ILM is accompanied by a corresponding reduction of the dimension of the pb motion space. If, on the contrary, the LS produces an IIM for the UKC, the pb motion space preserves its dimension, but it loses one of the actions intended for its control. Unless such an action is compensated by that resulting from an RSS, the CLS leads to an AS with global loss of control ($\underline{r}^{pb} > 0$).

If the LS appears both in the actuated and the unactuated KC, then $\underline{r}^i < \tilde{\underline{r}}^i$ and the KC-LA exhibits an ULS, hence an ALS. The quantity $N_{ALS} = \Delta\underline{R} = \tilde{\underline{R}} - \underline{R}$ represents the number of uncontrolled localized freedoms. As any RSS also produces an ALS, the number of ULSs is given by $N_{ULS} = N_{ALS} - N_{RSS}$ and consequently $N_{CLS} = N_{LS} - N_{ULS}$.

In an ULS, the leg exerts more reactions between p and b than it does in a full-cycle configuration ($\underline{c}^i > \tilde{\underline{c}}^i$). If the LS determines an SC for the UKC, the new reactions are linearly independent from the others ($c^{pb} > \tilde{c}^{pb}$) and they contribute to the pb motion. Hence, they may interfere with actions, potentially leading to an IAR. If the new reactions linearly depend on the others, namely the LS is a IIM for the UKC, they belong to C^{pb} and no IAR may occur.

3.4 Instantaneous Actuator Redundancy

An *instantaneous actuator redundancy* occurs any time an action linearly depends on other constraints, either passive or active. The number of actions generated by the motors between b and p is equal to $\tilde{r}^{pb} + N_{RSS} - N_{CLS}$ and the dimension of the space that they control is equal to $r^{pb} - \underline{r}^{pb}$. The number of linear dependen-

Table 1 Singularities and critical configurations of the unactuated kinematic chain.

Name	Acronym	Number or order
Leg singularity	LS	$N_{LS} = \Delta R$
	$LS\text{-}SC$	$N_{LS\text{-}SC} = \max(0, -\Delta r)$
	$LS\text{-}IIM$	$N_{LS\text{-}IIM} = N_{LS} - N_{LS\text{-}SC} = \Delta R - \max(0, -\Delta r)$
Constraint singularity	CS	$N_{CS} = \max(0, \Delta r)$
Stationary configuration	SC	$N_{SC} = N_{LS\text{-}SC} = \max(0, -\Delta r)$
Increased instantaneous mobility	IIM	$N_{IIM} = N_{LS\text{-}IIM} + N_{CS} = \Delta R + \Delta r = \Delta m$
Definitions: $\Delta R = \tilde{R} - R, \quad \Delta r = r^{pb} - \tilde{r}^{pb}, \quad \Delta m = m - \tilde{m}$		

Table 2 Singularities and critical configurations of the actuated kinematic chain.

Name	Acronym	Number or order
Redundant-set singularity	RRS	$N_{RSS} = R - R_{LR}$
	$RSS\text{-}SC$	$N_{RSS\text{-}SC} = r^{pb} - r^{pb}_{LR}$
	$RSS\text{-}IIM$	$N_{RSS\text{-}IIM} = N_{RSS} - N_{RSS\text{-}SC} = (R - R_{LR}) - (r^{pb} - r^{pb}_{LR})$
Actuated-leg singularity	ALS	$N_{ALS} = \Delta \underline{R}$
Uncontrolled leg-singularity	ULS	$N_{ULS} = N_{ALS} - N_{RSS} = \Delta \underline{R} - R + R_{LR}$
Controlled leg-singularity	CLS	$N_{CLS} = N_{LS} - N_{ULS} = \tilde{R} - R_{LR} - \Delta \underline{R}$
Platform/Base-motion singularity	PBS	$N_{PBS} = \underline{r}^{pb}$
Actuation singularity	AS	$N_{AS} = \begin{cases} \underline{r}^{pb}, & \Delta r \leq 0 \\ \underline{r}^{pb} - \Delta r, & 0 < \Delta r \leq \underline{r}^{pb} \\ 0, & \Delta r > \underline{r}^{pb} \end{cases}$
Uncontrolled freedoms	UF	$N_{UF} = N_{ALS} + N_{PBS} = \Delta \underline{R} + \underline{r}^{pb}$
Instant. redundancy of actuators	IAR	$N_{IAR} = N_{UF} - N_{IIM} = \Delta \underline{R} + \underline{r}^{pb} - \Delta R - \Delta r$
Definition: $\Delta \underline{R} = \tilde{\underline{R}} - \underline{R}$		

cies involving actions, given by the difference between these two quantities, is thus $N_{IAR} = \underline{r}^{pb} - \Delta r + N_{RSS} - N_{CLS}$ and it may be interpreted as the order of the IAR. By considering the expressions given in the previous sections for N_{RSS} and N_{CLS}, it is not difficult to prove that $N_{IAR} = \Delta \underline{R} + \underline{r}^{pb} - \Delta R - \Delta r = N_{UF} - N_{IIM}$.

4 Level 3: Input/Output-Mechanism Singularities

At this level, both the frame and the output member (or members) are specified. This is equivalent to choosing some relative freedoms as particularly significant, but no new phenomena or singularities are introduced.

From this point of view, all phenomena involving a link whose freedoms are chosen as an output acquire more relevance. In practice, any gain or loss of mobility for an output member may alter the mechanism functionality (as well as, indeed, a

change in the nature of the output motion space, an aspect that has not been considered here). In all these conditions, some outputs become, in fact, impossible to achieve (cf. [11]). Particular importance is also assumed by configurations leading to an IAR. In these situations, in fact, some inputs become impossible to produce and actuators (or links) risk being damaged.

5 Conclusions

This study has presented a complete and organic classification of singularities of parallel kinematic chains. The approach employed has the merit of recognizing hierarchical levels in which the objective causes of different physical phenomena may be distinguished and more easily interpreted. A better comprehension of the behaviour of the kinematic chain under the interaction of different types of singularity is thus made possible. The study has emphasized how different critical events may simultaneously take place, either compensating for or amplifying each other, or even originating new phenomena.

The adopted analytical tools, based on the theory of screws, make the comprehension of singularities and their identification particularly simple and physically intelligible.

References

1. Di Gregorio, R. and Parenti-Castelli, V.: Mobility analysis of the 3-UPU parallel mechanism assembled for a pure translation motion. *ASME J. of Mechanical Design* **124**(2), 259–264 (2002).
2. Galletti, C. and Fanghella, P.: Single-loop kinematotropic mechanisms. *Mechanism and Machine Theory* **36**(6), 743-761 (2001).
3. Gosselin, C.M. and Angeles, J.: Singularity analysis of closed-loop kinematic chains. *IEEE Trans. on Robotics and Automation* **6**(3), 281–290 (1990).
4. Hunt, K.H.: *Kinematic Geometry of Mechanisms*. Clarendon Press, Oxford (1978).
5. Joshi, S.A. and Tsai, L.W.: Jacobian analysis of limited-DOF parallel manipulators. *ASME J. of Mechanical Design* **124**(2), 254–258 (2002).
6. Kong, X. and Gosselin, C.: *Type Synthesis of Parallel Mechanisms*. Springer, Dordrecht (2006).
7. Liu, G., Luo, Y. and Li, Z.: Singularities of parallel manipulators: A geometric treatment. *IEEE Trans. on Robotics and Automation* **19**(4), 579–594 (2003).
8. Merlet, J.-P.: *Parallel Robots*. Springer, Dordrecht (2006).
9. Park, F.C. and Kim, J.W.: Singularity analysis of closed kinematic chains. *ASME J. of Mechanical Design* **121**(1), 32–38 (1999).
10. Zlatanov, D., Bonev, I.A. and Gosselin, C.M.: Constraint singularities of parallel mechanisms. In: *Proc. of the 2002 IEEE Int. Conference on Robotics and Automation*, pp. 496–502, Washington, DC, USA (2002).
11. Zlatanov, D., Fenton, R.G. and Benhabib, B.: Unifying framework for classification and interpretation of mechanism singularities. *ASME J. of Mechanical Design* **117**(4), 566–572 (1995).

Singularity Analysis through Static Analysis

J. Hubert and J.-P. Merlet

INRIA Sophia-Antipolis Mediterrane, 2004 route des lucioles, 06902 Sophia Antipolis Cedex, France, e-mail: julien.hubert@inria.fr; jean-pierre.merlet@sophia.inria.fr

Abstract. Singularity is a major problem for parallel robots as in these configurations the robot cannot be controlled. There may be very large forces/torques in its joints, possibly leading to its breakdown. This issue is clearly a very practical problem and we present in this paper an algorithm which computes the *static workspace* of a planar parallel robot for a given orientation i.e. the set of location of the platform at which the absolute value of all joint forces are smaller or equal to a given threshold

Key words: planar parallel robots, singularity, static workspace.

1 Introduction

Singularity analysis of parallel robots has a long history, starting with the pioneer works of Cauchy [4], Bricard [3] and Borel [2]. In the modern era such configurations have been studied by Hunt [6] and by Gosselin [5]. Since then this issue has been addressed by many authors [7, 18, 21].

Although it exists some kind of singularity where the joint forces/torques could resist to a very large external wrench, it is usually claimed that singularities should be avoided because in the vicinity of some singularities the joint forces may go to infinity, leading to a breakdown of the robot. This has led to major work to determine the singularity loci [8, 9, 13–15], to define a "distance" to a singular pose [17, 19], to determine trajectories that avoid singularities [1, 11, 12, 16], to investigate the relation between singularities and kinematics analysis [20] and finally to determine if a given workspace is singularity-free [10] in spite of the uncertainties in the robot modeling.

In this paper we follow another approach that is led by a very practical consideration: to avoid breaking the robot the main criteria is to avoid exceeding the maximal possible forces/torques in the legs. Hence being given a threshold $\tau_{\max}$ on the joint forces we define the *static workspace* as the set of poses so that

$$-\tau_{\max} \leq \tau \leq \tau_{\max} \tag{1}$$

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 13–20.

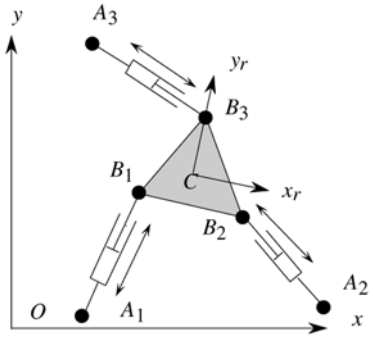


Fig. 1 The 3-RPR robot.

for a given load on the platform. The aim of this paper is to present an algorithm that allows one to compute the boundary of *static workspace* for a given orientation of the platform. This algorithm will be illustrated on the planar 3-RPR parallel robot.

2 Static Analysis

We consider a 3-RPR parallel robot (Figure 1) whose attachment points on the base (platform) are denoted A_i (B_i). The pose of the platform is parametrized by the coordinates in a given reference frame of a point C of the platform and by the rotation angle θ between the x axis of the reference frame and the x_r axis of the mobile frame. The length of leg i is denoted by $\rho_i = \|\mathbf{A_iB_i}\|$. The joint forces vector τ and the wrench $\mathcal{F}$ exerted on the platform are related by $\mathcal{F} = \mathbf{J}^{-T}\tau$, $\mathbf{J^{-T}}$ is the transpose of the kinematic Jacobian $\mathbf{J^{-1}}$ whose lines $\mathbf{J_i}$ are given by:

$$\mathbf{J_i} = \left(\frac{\mathbf{A_iB_i}}{\rho_i} \quad \frac{\mathbf{CB_i} \times \mathbf{A_iB_i}}{\rho_i} \right) \tag{2}$$

We derive from $\mathbf{J^{-1}}$ a matrix $\mathbf{M}$ whose rows are $\mathbf{M_i} = (\mathbf{A_iB_i} \quad \mathbf{CB_i} \times \mathbf{A_iB_i})^T$ while matrices $\mathbf{N_i}$ are obtained by the substitution of the ith column of $\mathbf{M}$ by $\mathcal{F}$. Note that such formulation is valid for many types of parallel robot (for example the Jacobian of the Gough platform has the same layout).

According to Cramer's rule applied on $\mathcal{F} = \mathbf{J}^{-T}\tau$ we get:

$$\tau_i = \rho_i \frac{|\mathbf{N_i}|}{|\mathbf{M}|} \tag{3}$$

Note that at a singularity (i.e $|\mathbf{M}| = 0$), we may still have finite joint forces satisfing (1). Hence in term of static, singularities are not the real issue but more important is their vicinity, which motivates the study of the the regions where $|\tau| \leq \tau_{\max}$, that

will be called the *static workspace*. This study is performed under the assumptions that θ and the wrench applied on the platform have both constant values. It must be noted that under these assumptions we have $\rho_i = \sqrt{(x - U_i)^2 + (y - V_i)^2}$ where U_i, V_i are constants that depend only on the leg and on θ.

3 Border of the Static Workspace

It can be shown that a point belongs to the boundary of the *static workspace* if at least one component τ_i of τ verifies $|\tau_i| = \tau_{\max}$. Let us assume first that $|\mathbf{M}| \neq 0$. If $\tau_i = \tau_{\max}$ or $\tau_i = -\tau_{\max}$ then (3) can be rewritten as

$$\tau_{\max}|\mathbf{M}| - \rho_i|\mathbf{N_i}| = 0, \quad \tau_{\max}|\mathbf{M}| + \rho_i|\mathbf{N_i}| = 0 \tag{4}$$

Curves in the x-y plane that satisfies $\tau_i = \tau_{\max}$ are called *n-type curves* and those that satisfies $\tau_i = -\tau_{\max}$ are called *m-type curves*. These curves are not algebraic as ρ_i is a square root of an algebraic equation in x, y. But it must be noted that m-curves and n-curves satisfy both the equation

$$\tau_{\max}^2|\mathbf{M}|^2 - \rho_i^2|\mathbf{N_i}|^2 = 0 \tag{5}$$

that denotes the *q-curves* and are algebraic of total degree 6.

Now, if we assume that $|\mathbf{M}| = 0$ (i.e. we are at a singularity), then τ_i may be finite only in two cases, namely if $|\mathbf{N_i}| = 0$ or $\rho_i = 0$. Poses at which we have $|\mathbf{M}| = 0$, $|\mathbf{N_i}| = 0$ are labeled as *N-points* while poses at which $|\mathbf{M}| = 0$, $\rho_i = 0$ are denoted by *R-points*. It must be noted that both R and N-points belong simultaneously to the n and m-curves.

Hence the boundary of the *static workspace* is constituted of arcs of m and n-curves with possibly some R and N-points.

4 Algorithm Computing the Boundary of the Static Workspace

4.1 The Key-Points

Let us consider a point M_1 on the n-curve n_1 (i.e. at this pose we have $\tau_1 = \tau_{\max}$) and assume that $|\tau_2|, |\tau_3| \leq \tau_{\max}$. If we move along n_1 we may arrive at a pose M_2 at which we have $|\tau_2| > \tau_{\max}$. This implies that there is pose M_3 between M_1 and M_2 on the curve n_1 such that $|\tau_2| = \tau_{\max}$ and M_3 is an intersection between n_1 and n_2 or m_2. The poses which lie on the intersections of two n-curve n_i or two m-curve m_i or one n-curve and one m-curve for which we have $|\tau_i| = \tau_{\max}$ are called *key-points* and by extension the N and R-points are also considered as key-points.

Now consider the part of an n-curve (or an m-curve) between two successive key-points. If a pose of this part lies on the boundary of the static workspace, then it is easy to show that the whole part is a component of the border.

In summary the set of key-points is consitutued of the R and N-points and the intersection points between the n and m-curves, which can be easily calculated with the q-curves. Indeed the intersections between two such curves are the intersection points between all pairs of corresponding m and n-curves. The intersection points between two algebraic q-curves q_i, q_j may easily be computed with a resultant and attributed to n_i, m_i, n_j or m_j. A similar procedure allows one to calculate the coordinates of the R and N-points.

Note that we may have infinite arcs: to avoid this problem we define a bounding box and draw the border only within it. It may also occur that the list of key-points is empty either because the static workspace has no intersection with the bounding box or if there is no intersection between the n and m-curves and some n and m-curves are fully included in the bounding box. To deal with this case we compute the intersections $\mathcal{I}$ of the n and m-curves with the bounding box. If these intersections are not empty, then the intersection points are added as key-points. Then, we compute arbitrary poses on the curves: if the set $\mathcal{T}$ of such poses are included in, the bounding box is no empty, then we use $\mathcal{T}$ as additional key-points. If $\mathcal{I}$ and $\mathcal{T}$ are empty, then the static workspace has no intersection with the bounding box.

As seen previously determining if part of an n or m-curve belong to a border may be performed by considering an arbitrary pose on the arc and computing the τ at this pose: the $|\tau_i|$ must be all lower or equal to $\tau_{\max}$. The procedure `On_Border(x,y)` returns 1 if x, y is on the border, 0 otherwise. Note also that for the calculation of the border we add a few additional key-points, that will be presented later.

4.2 Determining the Curves

After having computed the key-points, we must determine the arcs of the curves between two successive key-points. Without lack of generality we consider a n-curve n_i. We start from a key-point K_1 with coordinates (x_1, y_1) and look at the closest key-points $K_f(x_f, y_f)$ from K_1 with $x_f > x_1$. We then compute the two tangent unit vectors T_1, T_2 of the n-curve at this point. We consider first T_1 with components (t_x, t_y). If $t_x^2 \geq t_y^2$ we substitute $x = x_n = x_1 + \Delta$, $y = y + 1 + \epsilon$ in n_i where Δ is a small fixed increment. We then solve $n_i = 0$ in ϵ, looking only for "small" solutions (typically of absolute value lower than 10Δ). If we obtain more than one solution, we divide Δ by 2 and start again, while if no solution is obtained we stop the process. If we get only one of such solutions, we obtain a new point on the n-curve with coordinates $(x_n = x_1 + \Delta, y_n = y_1 + \epsilon)$. We use then `On_Border`$(x_n, y_n)$ to determine if the arc we are following is on the border. If this procedure returns 1, then we repeat the process for this new point. Otherwise the process is stopped.

The process is repeated until one of the following cases occur:

- if $x_1 + \Delta > x_f$ we adjust Δ in such way that $x_1 + \Delta = x_f$. We compute then $y_n = y_1 + \epsilon$ and if $y_n = y_f$, then we have determined an arc of curve between K_1, K_f which are successive key-points. If $y_n \neq y_f$ the process continues.
- we compute the tangent unit vectors at x_n, y_n. If at this point $t_y^2 > t_x^2$ we stop the process and store (x_n, y_n) as new additional key-point. The motivation here is to increment by Δ the variable that has locally the largest variation so that the other variable will exhibit only minor variations.

When this process is terminated, we repeat it with $x = x_n = x_1 - \Delta$ instead of $x = x_n = x_1 + \Delta$, in order to determine another arc starting form K_1.

If $t_y^2 > t_x^2$ the principle is the same except that it is the variable y which is incremented by Δ.

Concerning the R-points which require a specific treatment as they have no tangent vectors (indeed the derivatives of the curves with respect to x, y involve $1/\rho$ while at an R-point $\rho = 0$). For an R-point of coordinates x_1, y_1 we use all the small solutions obtained for $x_1 \pm \Delta$, $y_1 \pm \Delta$ as new key-points and the above procedure is not used for the R-points.

5 Examples

The algorithm described in the paper was implemented in `Maple` and was tested for the planar parallel robot 3-RPR whose geometry is defined by $OA_1 = (0, 0)$, $OA_2 = (20, 0)$, $OA_3 = (12, 10)$, $CB_{r1} = (-4, 4)$, $CB_{r2} = (4, -4)$ and $CB_{r3} = (0, 2)$, units being irrelevant as soon as th same one is used for all calculations.

We choose a threshold $\tau_{\max} = 3$ and an increment $\Delta = 0.05$. The curves that are involved and the static workspace border are presented for $\mathcal{F} = (4, 0, 0)$, $\theta = 0.1$ rad in Figure 2, while Figure 3 presents the static workspace for $\theta = -2.91$ rad and $\mathcal{F} = (0, 0, 5)$. Note that the crosses on the second part of figure 2 represent the points used to determine if a region is or not on the boundary of the static workspace.

The computation time for determining the static workspace varies between 50 s and 500 s.

Some interesting results may be noticed on these figures. For example in Figure 3 the static workspace is composed of three components that are connected at point S_1, S_2. These two poses are singular. Hence it is theoretically possible to move from one component to another one by going through the singularity. In practice however the useful workspace is reduced to only one component because the robot control will not allow to generate a trajectory that goes *exactly* through S_i. It must also be noted that the static workspace may have components that extend to infinity.

The effect of the kinematic constraints may be seen on figure 3: here we have imposed that the leg lengths are limited to the range [3, 12]. The resulting workspace is composed of the intersection of the 3 large circles minus the union of the 3 small circles. But if we take into account the static workspace this workspace is reduced to 2 components: a very small area (the one located around $x = -15$ close to the

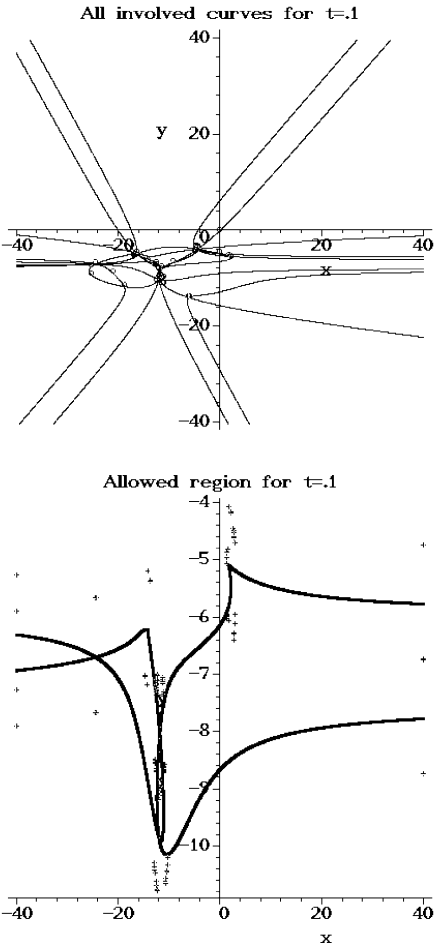


Fig. 2 Curves involved and static workspace border for $\mathcal{F} = (4, 0, 0)$, $\theta = 0.1$ rad.

two small circles having an intersection and a larger one between $x = -10$ and $x = -5$).

6 Conclusion

We have presented in this paper an algorithm which allows one to determine the boundary of the *static workspace* of a planar manipulator for a given orientation and a given external wrench. This approach is another way, based on a physical requirement, to manage singularity. It has been illustrated on a planar parallel robot

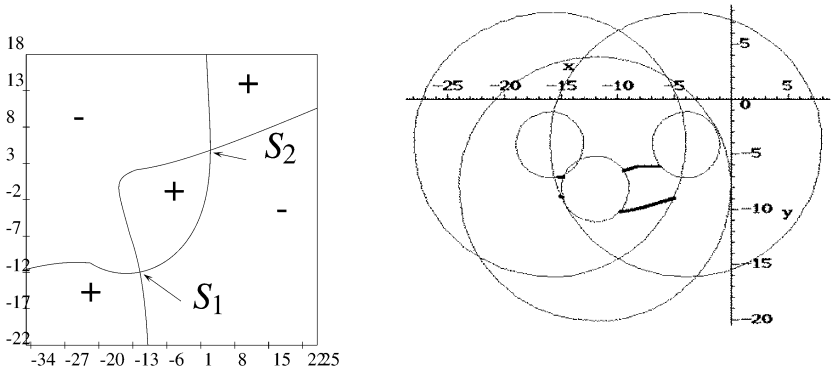


Fig. 3 On the right the static workspace for $\theta = 2.91$ rad and $\mathcal{F} = (0, 0, 5)$ (denoted by "+"). On the right the workspace based on limits on the leg lengths is reduced by the static workspace to 2 components.

but may be used as well on other parallel manipulators and will provide planar cross-sections of the static workspace.

This work may be extended in various ways. It may be interesting to compute the static workspace for a given orientation range. Using our approach we may calculate the *static workspace* for various values of the orientation angle. Then we will compute the intersections of all these static workspaces (which is possible as basically the boundary of the static workspace is a polygonal line and it exists efficient algorithms to compute the intersections of polygons). A similar approach may be used if the wrench applied on the platform lies within some given range.

Clearly with this approach the result will only be an approximation of the static workspace for the orientation or a wrench range. But we will be sure that the real static workspace is included in the resulting polygon(s) and we may then use, for example, interval analysis to refine the result.

A similar method may be used to take modeling uncertainties (e.g. uncertainties on the location of the A_i, B_i) into account. A first result is obtained through our approach, then interval analysis may be used with the initial result as starting point to determine a "safe" static workspace that will be guaranteed to include on poses strictly satisfying the constraint (1) whatever are the tolerances on the location of the A_i, B_i.

Acknowledgement

The research work reported here was made possible by grant ANR SIROPA and Eu ARES NEST grant No. 015653.

References

1. Bhattacharya S., Hatwal H., and Ghosh A., Comparison of an exact and an approximate method of singularity avoidance in platform type parallel manipulators. *Mechanism and Machine Theory* **33**(7), 965–974, 1998.
2. Borel E., Mémoire sur les déplacements à trajectoire sphériques. *Mémoire présentés par divers savants* **33**(1), 1–128, 1908.
3. Bricard R., Mémoire sur la théorie de l'octaèdre articulé. *Journal de Mathématiques pures et appliquées, Liouville*, **3**, 113–148, 1897.
4. Cauchy A., Deuxième mémoire sur les polygones et les polyèdres. *Journal de l'École Polytechnique*, May, 87–98, 1813.
5. Gosselin C. and Angeles J., Singularity analysis of closed-loop kinematic chains. *IEEE Trans. on Robotics and Automation* **6**(3), 281–290, 1990.
6. Hunt K.H., *Kinematic Geometry of Mechanisms*. Clarendon Press, Oxford, 1978.
7. Husty M.L. and Karger A., Architecture singular parallel manipulators and their self-motions. In *Advances in Robot Kinematics*, J. Lenarčič and M.M. Stanisič, Springer, Dordrecht, pp. 355–364, 2000.
8. Mayer St-Onge B. and Gosselin C.M., Singularity analysis and representation of the general Gough–Stewart platform. *Int. J. of Robotics Research* **19**(3), 271–288, 2000.
9. Merlet J.-P., Singular configurations of parallel manipulators and Grassmann geometry. *Int. J. of Robotics Research* **8**(5), 45–56, 1989.
10. Merlet J.-P., A formal-numerical approach for robust in-workspace singularity detection. *IEEE Trans. on Robotics* **23**(3), 393–402, 2007.
11. Nenchev D.N. and Uchiyama M., Singularity-consistent path planning and control of parallel robot motion through instantaneous-self-motion type. In *IEEE Int. Conf. on Robotics and Automation*, Minneapolis, 24–26 April, pp. 1864–1870, 1996.
12. O'Brien J.F. and Wen J.T., Kinematic control of parallel robots in the presence of unstable singularities. In *IEEE Int. Conf. on Robotics and Automation*, Seoul, 23–25 May, pp. 3154–3159, 2001.
13. Pernkopf F., *Workspace Analysis of Stewart–Gough Platforms*. Ph.D. Thesis, Baufakultät, University of Innsbruck, September 2003.
14. Pottmann H., Peternell M., and Ravani B., Approximation in line space. Applications in robot kinematics. In *Advances in Robot Kinematics: Analysis and Control*, J. Lenarčič and M.L. Husty, Springer, Dordrecht, pp. 403–412, 1998.
15. Sefrioui J. and Gosselin C.M., On the quadratic nature of the singularity curves of planar three-degree-of-freedom parallel manipulators. *Mechanism and Machine Theory* **30**(4), 533–551, 1995.
16. Sen S., Dasgupta B., and Mallik A.K., Variational approach for singularity-path planning of parallel manipulators. *Mechanism and Machine Theory* **38**(11), 1165–1183, 2003.
17. Voglewede P.A. and Ebert-Uphoff I., Measuring "closeness" to singularities for parallel manipulators. In *IEEE Int. Conf. on Robotics and Automation*, New Orleans, 28–30 April, pp. 4539–4544, 2004.
18. Wohlhart K., Degrees of shakiness. *Mechanism and Machine Theory* **34**(7), 1103–1126, 1999.
19. Wolf A. and Shoham M., Investigation of parallel manipulators using linear complex approximation. *ASME J. of Mechanical Design* **125**(3), 564–572, 2003.
20. Zein M., Wenger P., and Chablat D., Singular curves and cusp points in the joint space of 3-RPR parallel manipulators. In *IEEE Int. Conf. on Robotics and Automation*, Orlando, 16–18 May, pp. 777–782, 2006.
21. Zlatanov D., Fenton R.G., and Benhabib B., Identification and classification of the singular configurations of mechanisms. *Mechanism and Machine Theory* **33**(6), 743–760, 1998.

Constraint Singularities and the Structural Parameters of Parallel Robots

Grigore Gogu

Mechanical Engineering Research Group, French Institute of Advanced Mechanics and Blaise Pascal University, Clermont-Ferrand, France, e-mail: grigore.gogu@ifma.fr

Abstract. The paper addresses the constraint singularities in connection with the structural parameters of parallel robots. The new formulae of mobility, connectivity, overconstraint and redundancy of parallel robots, recently proposed by the author, are used to characterize the constraint singulatities. By using these formulae, we demonstrate that in a constraint singularity the instantaneous values of the mobility, connectivity of the moving platform and degree of overconstraint increase with no changes in limb connectivity. The constraint singularities are easily identified by inspection without Jacobian calculation.

Key words: constraint singularity, mobility, connectivity, redundancy, overconstraint.

1 Introduction

In a general mathematical sense, a singularity is a point at which a given mathematical model is not defined or fails to be well-behaved in some particular way. In engineering, a singularity is a configuration of a system in which the subsequent behaviour cannot be predicted. In a singular configuration, the underlying equations of the mechanical model exhibit a mathematical singularity. The corresponding motions, forces, or other physical parameters modelled by these equations become undetermined.

Numerous investigations have been conducted on singular configurations of mechanisms, with recent emphasis on parallel robotic manipulators. The identification of singular configurations in parallel robots is of great importance. Near singular configuration, infinitesimal changes in input/output velocities or forces can produce huge variations of outputs/inputs. In singular configurations, the mechanism loses the ability to transmit resolutely motion and force, and becomes uncontrollable; the kinematic and static behaviour of the mechanism change dramatically; the mechanism gains extra degrees of freedom and loses its stiffness. Furthermore, the actuator forces may become very intense; this might lead to robot failure or permanent damage of the robot and surrounding equipment, including a breakdown of the mechanism. Therefore, much effort has been devoted to avoiding singularities

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 21–28.

by a proper design. When they do exist in the workspace, they must be circumvented by proper path planning.

The term of constraint singularities has been recently coined by Zlatanov et al. (2002a) to characterize the configuration of lower mobility parallel manipulators in which both the connectivity of the moving platform and the mobility of the parallel mechanism increase their instantaneous values. This type of singularity was initially identified as a configuration space singularity of the 3-UPU robot at SNU – Seoul National University (Han et al., 2002). In this notation U stands for universal and P for prismatic joints. At its home position this translational parallel robot exhibits finite motions even with all active prismatic joints locked. Various tools have been used to study this singularity: the rank and the condition number of the Jacobian matrix of the loop closure equations (Han et al., 2002), the screw theory (Zlatanov et al., 2002a, 2002b), the augmented Jacobian matrix (Joshi and Tsai, 2002), the linear complex approximation (Wolf and Shoham, 2002; Wolf et al., 2002), Morse function theory and differential forms associated with the constraint functions (Liu et al., 2003).

The 3-UPU mechanism was studied for the first time in Tsai (1996) where the direct position analysis restricted to the search for the translational configurations was addressed. Di Gregorio and Parenti-Castelli (2002) analyzed in analytic form both translation and rotation singularities of this parallel mechanism.

The main objective of this paper is to propose a simple approach to identify and characterize the constraint singularities in connection with the structural parameters of the parallel robots.

2 Structural Parameters of the Parallel Robots

The main structural parameters of the parallel robots are associated with mobility, connectivity, redundancy and overconstraint. We recall briefly the meaning of these parameters. More details can be found in Gogu (2008).

IFToMM terminology defines the mobility or the degree of freedom as the number of independent coordinates required to define the configuration of a kinematic chain or mechanism (Ionescu, 2003).

We note that the classical formulae for a quick calculation of mobility, known as Chebychev–Grübler–Kutzbach formulae do not fit many classical mechanisms and recent parallel robots. These formulae have been recently reviewed in Gogu (2005a) and their limits have been set up in Gogu (2005b). New formulae for quick calculation of the mobility have been proposed in Gogu (2005c, 2005d) and demonstrated via the theory of linear transformations. A recent development of these contributions can be found in Gogu (2008).

The connectivity between two links of a mechanism represents the number of independent finite and/or infinitesimal displacements allowed by the mechanism between the two links.

The number of overconstraints of a mechanism is given by the difference between the maximum number of joint kinematic parameters that could lose their independence in the closed loops, and the number of joint kinematic parameters that actually lose their independence in the closed loops.

The redundancy is given by the difference between the mobility of the parallel mechanism and the connectivity of the moving platform.

Let us consider the case of a parallel mechanism $F \leftarrow G_1 - G_2 - \cdots G_k$ in which the end-effector $n \equiv n_{G_i}$ is connected to the reference link $1 \equiv 1_{G_i}$ by k simple or complex kinematic chains G_i ($1_{G_i} - 2_{G_i} - \cdots n_{G_i}$) called limbs (legs). The mechanism $F \leftarrow G_1 - G_2 - \cdots G_k$ is characterized by:

- R_{G_i} – the vector space of relative velocities between the extreme links n_{G_i} and 1_{G_i} in the kinematic chain G_i disconnected from the mechanism F,
- R_F – the vector space of relative velocities between the extreme links $n \equiv n_{G_i}$ and $1 \equiv 1_{G_i}$ in the mechanism $F \leftarrow G_1 - G_2 - \cdots G_k$,
- $S_{G_i} = \dim(R_{G_i})$ – the connectivity between extreme the links n_{G_i} and 1_{G_i} in the kinematic chain G_i disconnected from the mechanism F,
- $S_F = \dim(R_F)$ – the connectivity between the extreme links $n \equiv n_{G_i}$ and $1 \equiv 1_{G_i}$ in the mechanism $F \leftarrow G_1 - G_2 - \cdots G_k$.

These parameters are related by the new formulae for mobility M, connectivity S_F, overconstraint N, and redundancy T of the parallel mechanism $F \leftarrow G_1 - G_2 - \cdots G_k$ (Gogu, 2008):

$$M = \sum_{j=1}^{p} f_j - r, \tag{1}$$

$$N = 6q - r, \tag{2}$$

$$T = M - S_F, \tag{3}$$

where

$$S_F = \dim(R_F) = \dim(R_{G_1} \cap R_{G_2} \cap \cdots \cap R_{G_k}), \tag{4}$$

$$r = \sum_{j=1}^{k} S_{G_j} - S_F + r_l, \tag{5}$$

and

$$r_l = \sum_{j=1}^{k} r_l^{G_j}, \tag{6}$$

where p represents the total number of joints, q is the total number of independent closed loops in the sense of graph theory, f_j is the mobility of the jth joint, r is the number of joint parameters that lose their independence in the mechanism F, $r_l^{G_j}$ is the number of joint parameters that lose their independence in the closed loops of the limb G_j, and r_l is the total number of joint parameters that lose their independence in the closed loops that may exist in the limbs of the mechanism F.

We note that the intersection in Eq. (4) is consistent if the operational velocity spaces R_{G_i} are defined by the velocities of the same point situated on the end-effector. This point is called the characteristic point, and is denoted in this paper by H.

The connectivity S_F of the end-effector $n \equiv n_{G_i}$ in the mechanism $F \leftarrow G_1 - G_2 - \cdots G_k$ is less than or equal to the mobility M of the mechanism F.

The basis of the vector space R_F of relative velocities between the end-effector $n \equiv n_{G_i}$ and the reference link $1 \equiv 1_{G_i} \equiv 0$ in the mechanism $F \leftarrow G_1 - G_2 - \cdots G_k$ does not vary with the position of the characteristic point on $n \equiv n_{G_i}$.

When there are various ways to choose the basis of the operational spaces, the bases of R_{G_i} in Eq. (4) are selected such as the minimum value of S_F is obtained. By this choice, the result of Eq. (4) fits in with general mobility definition as the minimum value of the instantaneous mobility.

Equations (1–6) are valid for any parallel mechanism in which no joint belongs to more than one limb. This condition can be expresses by

$$p = \sum_{j=1}^{k} p_{G_j}, \tag{7}$$

where p_{G_j} is the number of joints of jth limb.

In a singular configuration of the parallel mechanism $F \leftarrow G_1 - G_2 - \cdots G_k$, at least one of the structural parameters in Eq. (5) is instantaneously altered. We denote by an anterior superior index i the instantaneous values of the structural parameters defined above. We note that, in general, any structural parameter used in Eqs. (1–6) can be affected by singular configurations excepting f_j and q.

3 Constraint Singularities

In this section, we discuss the nature of constraint singularities, by using Eqs. (1–6), and we demonstrate the relation between these singularities and the degree of overconstraint of the parallel mechanism and the connectivity of the limbs.

Property 1. If the vector space of relative velocities between the extreme links n_{G_i} and 1_{G_i} in the kinematic chain G_i disconnected from the mechanism F, accept various bases, the connectivity of the moving platform can increase instantaneously its value ${}^{i}S_F > S_F$, with no instantaneous change in leg connectivity (${}^{i}S_{G_i} = S_{G_i}$).

This property results directly from Eq. (4) and shows that a constraint singularity may occur when the vector space of relative velocities between the extreme links n_{G_i} and 1_{G_i} in the kinematic chain G_i disconnected from the mechanism F, accept various bases at least for one of the limbs.

Property 2. An instantaneous increase of the connectivity of the moving platform (${}^{i}S_F - S_F$) is accompanied by an identical increase of the mobility of the parallel mechanism (${}^{i}M_F - M_F$) when no instantaneous changes occur in limb connectivity and the number of joint parameters that lose their independence in the closed loops that may exist in limb G_j

$$ {}^{i}M_F - M_F = {}^{i}S_F - S_F \tag{8} $$

when ${}^{i}S_{G_i} = S_{G_i}$ and ${}^{i}r_l = r_l$. This property results directly from Eqs. (1) and (5).

Property 3. An instantaneous increase of the connectivity of the moving platform (${}^{i}S_F - S_F$) is accompanied by an identical increase of the degree of overconstraint of the parallel mechanism (${}^{i}N - N$) when no instantaneous changes occur in leg connectivity and the number of joint parameters that lose their independence in the closed loops that may exist in limb G_j

$$ {}^{i}N - N = {}^{i}S_F - S_F \tag{9} $$

when ${}^{i}S_{G_i} = S_{G_i}$ and ${}^{i}r_l = r_l$. This property results directly from Eqs. (1) and (5).

Property 4. An instantaneous increase of the connectivity of the moving platform (${}^{i}S_F - S_F$) accompanied by an identical increase of the mobility of the parallel mechanism (${}^{i}M_F - M_F$) does not affect the redundancy of the parallel mechanism.

This property results directly from Eq. (3).

The four properties characterizing the constraint singularities are illustrated with respect to the well known 3-UPU mechanism in Figure 1a. We will show that the constraint singularities of this mechanism can be easily identified by inspection with no need to calculate the augmented Jacobian. The parallel mechanism has three simple legs (no closed loops exist inside a leg), that is ${}^{i}r_l = r_l = 0$. The axes of the first revolute joints of each leg are coplanar and form the fixed platform plane. The axes of the last revolute joints of each leg are also coplanar and form the moving platform plane. The relative position of the three revolute joints in the two planes is the same but the dimensions of the mobile and fixed platforms are different (see Figure 1a). Each universal joint is composed by two revolute joints with perpendicular axes. Each leg has $S_{G_i} = 5$ and each vector space R_{G_i} can have one of the following bases: $(\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_x, \boldsymbol{\omega}_y)$, $(\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_x, \boldsymbol{\omega}_z)$ and $(\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_y, \boldsymbol{\omega}_z)$. Three different values can be obtained for the connectivity between the moving and the fixed platforms of 3-UPU mechanism. The minimum value $S_F = 3$ is associated with the mechanism in Figure 1a where the moving platform have just three independent translations. For example if we chose $(R_{G_1}) = (\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_x, \boldsymbol{\omega}_y)$, $(R_{G_2}) = (\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_x, \boldsymbol{\omega}_z)$ and $(R_{G_3}) = (\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_y, \boldsymbol{\omega}_z)$, Eqs. (1–5) give $M = 3$, $N = 0$ and $T = 0$.

For certain configurations the moving platform can have ${}^{i}S_F = 4$ or 5. These instantaneous values are associated with the constraint singularities. In the configuration in Figure 1b, the axes of the first two revolute joints of two limbs are coplanar, and the axes of the last two revolute joints of the same limbs are also coplanar.

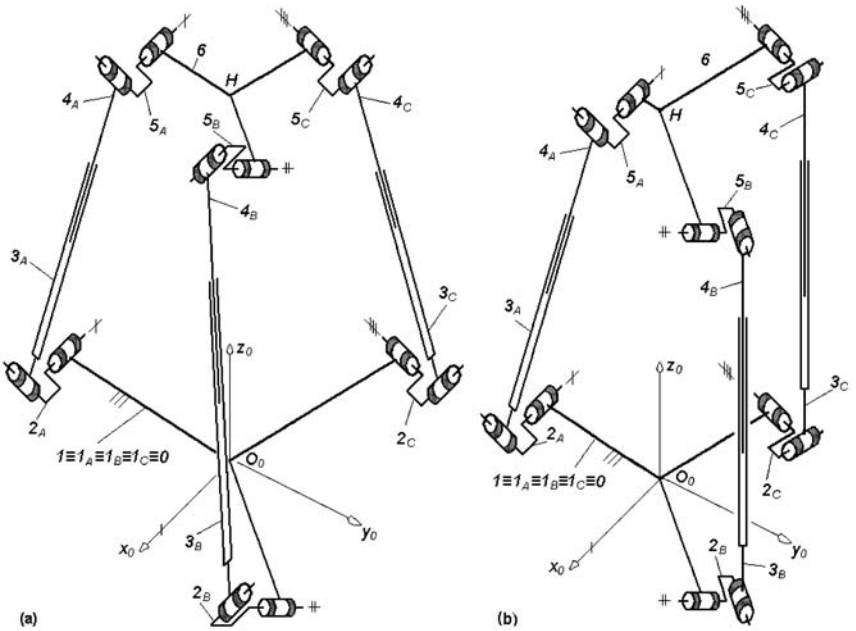


Fig. 1 Two solutions of 3-UPU parallel mechanisms: general solution with no constraint singularities (a) and a solution with a constraint singularity (b).

The two planes to which the revolute joints belong are parallel. In this configuration, the vector spaces of relative velocities between the extreme links n_{G_i} and 1_{G_i} in the kinematic chain G_i disconnected from the mechanism F have the following bases $({}^iR_{G_1}) = (\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_x, \boldsymbol{\omega}_z)$, or $({}^iR_{G_1}) = (\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_y, \boldsymbol{\omega}_z)$ and $({}^iR_{G_2}) = ({}^iR_{G_3}) = (\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_x, \boldsymbol{\omega}_y)$ with $S_{G_i} = 5$. Equation (4) gives ${}^iS_F = 4$. The moving platform has instantaneously four independent velocities: three translations and one rotation. Equations (1–3) give ${}^iM = 4$, ${}^iN = 1$ and ${}^iT = 0$.

In the configurations in Figure 2, the axes of the first two revolute joints of the three limbs are coplanar, and the axes of the last two revolute joints of the three limbs are also coplanar. The two planes to which the revolute joints belong are parallel. In this configuration, the vector spaces of relative velocities between the extreme links n_{G_i} and 1_{G_i} in the kinematic chain G_i disconnected from the mechanism F have the same bases, for example $({}^iR_{G_1}) = ({}^iR_{G_2}) = ({}^iR_{G_3}) = (\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z, \boldsymbol{\omega}_x, \boldsymbol{\omega}_y)$ with $S_{G_i} = 5$. Equation (4) gives ${}^iS_F = 5$. The moving platform has instantaneously five independent velocities: three translations and two rotations. Equations (1–3) give ${}^iM = 5$, ${}^iN = 2$ and ${}^iT = 0$. The moving platform and the base are identical in Figure 2a. The constraint singularity in Figure 2b was firstly observed on SNU parallel robot (Han et al., 2002).

The constraint singularities in Figures 2a and 2b have been identified by Joshi and Tsai (2002) by calculating the rank of the augmented (overall) Jacobian. The constraint singularity in Figure 1b is identified for the first time by using the method presented in this paper.

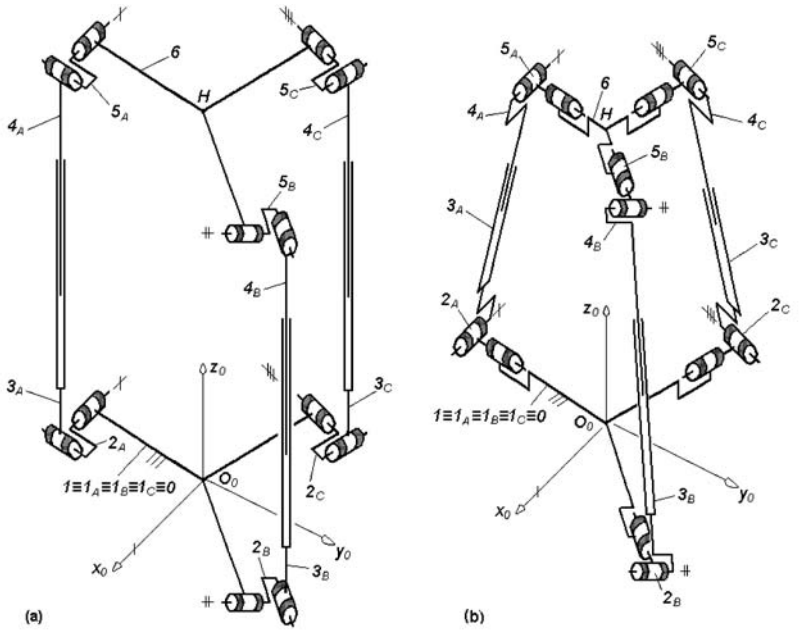


Fig. 2 Two solutions of 3-UPU parallel mechanisms with constraint singularities.

4 Conclusions

Constraint singularities are used to characterize the configuration of lower mobility parallel manipulators in which both the connectivity of the moving platform and the mobility of the parallel mechanism increase their instantaneous values. In this paper, we have explained the nature of these singularities, by using the new formulae of mobility, connectivity, overconstraint and redundancy of parallel robots, recently proposed by the author, and we have demonstrate the relation between these singularities and the structural parameters of the parallel robots. In this way, the constraint singularities can be easily identified by inspection with no need to calculate the augmented Jacobian.

References

Di Gregorio, R., Parenti-Castelli, V. (2002), Mobility analysis of the 3-UPU parallel mechanism assembled for a pure translational motion, *Transactions of the ASME Journal of Mechanical Design* **124**, 259–264.

Gogu, G. (2005a), Mobility of mechanisms: A critical review, *Mechanism and Machine Theory* **40**, 1068–1097.

Gogu, G. (2005b), Chebychev–Grubler–Kutzbach's criterion for mobility calculation of multi-loop mechanisms revisited via theory of linear transformations, *European Journal of Mechanics, A – Solids* **24**, 427–441.

Gogu, G. (2005c), Mobility and spatiality of parallel robots revisited via theory of linear transformations, *European Journal of Mechanics, A – Solids* **24**, 690–711.

Gogu G. (2005d), Mobility criterion and overconstraints of parallel manipulators, in *Proceedings of International Workshop on Computational Kinematics*, Cassino, Italy.

Gogu, G. (2008), *Structural Synthesis of Parallel Robots*, Springer, Dordrecht.

Han, C., Kim, J., Kim, J., Park, F.C. (2002), Kinematic sensitivity analysis of the 3-UPU parallel mechanism, *Mechanism and Machine Theory* **37**, 787–798.

Ionescu, T.G. (2003), Terminology for mechanisms and machine science, *Mechanism and Machine Theory* **38**, 597–901.

Joshi, S.A., Tsai, L.W. (2002), Jacobian analysis of limited-DOF parallel manipulators, *Transactions of the ASME Journal of Mechanical Design* **124**, 254–258.

Liu, G., Lou, Y., Li, Z. (2003), Singularities of parallel manipulators: A geometric treatment, *IEEE Trans Robotics Automation* **19**(4), 579–594.

Tsai, L.-W. (1996), Kinematics of a three-dof platform with three extensible limbs, in *Recent Advances in Robot Kinematics*, Lenarčič, J., Parenti-Castelli, V. (Eds), Kluwer Academic Publishers, Dordrecht, pp. 401–410.

Wolf, A., Shoham, M. (2003), Investigation of parallel manipulators using linear complex approximation, *Transactions of the ASME Journal of Mechanical Design* **125**, 564–572.

Wolf, A., Shoham, M., Park, F.C. (2002) Investigation of singularities and self-motions of the 3-UPU robot, in *Advances in Robot Kinematics: Theory and Applications*, Lenarčič J., Thomas F. (Eds), Kluwer Academic Publishers, Dordrecht, pp. 165–174.

Zlatanov, D., Bonev, I.A., Gosselin, C.M. (2002a), Constraint singularities of parallel mechanisms, in *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, D.C., USA, pp. 496–502.

Zlatanov, D., Bonev, I.A., Gosselin, C.M. (2002b), Constraint singularities as configuration space singularities, in Lenarčič J., Thomas F. (Eds), *Advances in Robot Kinematics: Theory and Applications*, Kluwer Academic Publishers, Dordrecht.

Forward Kinematics and Singularity Analysis of a 3-RPR Planar Parallel Manipulator

Xianwen Kong

Department of Mechanical Engineering, School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh EH14 4AS, U.K., e-mail: x.kong@hw.ac.uk

Abstract. This paper first deals with the forward kinematics (forward displacement analysis) of a class of 3-RPR planar parallel manipulator with congruent equilateral base and moving platform using an algebraic approach. Conditions on the inputs under which the manipulator has different number of solutions to the forward kinematics, including a double solution, two distinct solutions or infinite solutions (self-motion), are identified. The Type 2 singularity analysis is then performed for this parallel manipulator. The result, given in [1] without detailed proof, regarding the Type 2 singularities with a non-zero orientation angle is then proved.

Key words: parallel manipulator, forward kinematics, singularity, self-motion, Cardanic motion.

1 Introduction

Recently, a 3-RPR planar parallel manipulator with congruent equilateral base and moving platform (Figure 1) has been proposed in [1]. A comprehensive kinematic study on this parallel manipulator has been performed using a geometric approach. It was pointed out, without detailed proof (see the last paragraph in [1, p. 224]), that the only Type 2 singular configurations, for which the base and the moving platform do not coincide, belong to self motions. In such singular configurations, the legs intersect at the circumcircle of the base.

For a better understanding of this parallel manipulator, the forward kinematics (forward displacement analysis) and the singularity analysis will be investigated systematically using an algebraic approach.

2 Loop Closure Equation

The 3-RPR planar parallel manipulator (Figure 1) is composed of a base and moving platform connected by three legs. The base ($A_1A_2A_3$) and the moving platform ($B_1B_2B_3$) are congruent equilateral triangles. Each leg is composed of an R, a P

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 29–38.

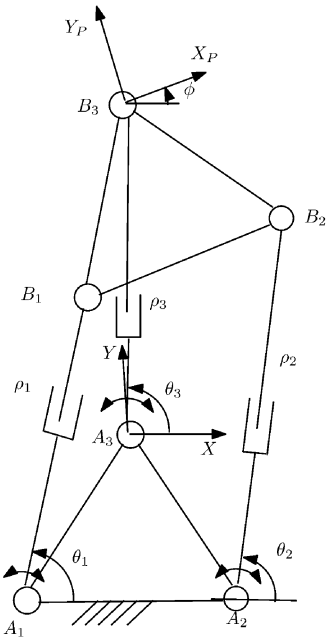


Fig. 1 3-RPR planar parallel manipulator.

and an R joint in sequence. The positive direction of each P joint is indicated by an arrow in the figures except for Figure 1. All the three R joints located on the base are actuated. Two coordinate systems, $A_3 - XY$ and $B_3 - X_P Y_P$, are fixed on the base and the moving platform, respectively. The location of the moving platform can be represented with (x, y), the position of O_P, and ϕ, the orientation angle of the moving platform.

The loop closure equations of loops $A_3B_3B_1A_1$ and $A_3B_3B_2A_2$ in complex number form are [3]

$$\begin{cases} \rho_3 e^{i\theta_3} = e^{-i(2\pi/3)} + \rho_1 e^{i\theta_1} - e^{i(\phi - 2\pi/3)} \\ \rho_3 e^{i\theta_3} = e^{-i(\pi/3)} + \rho_2 e^{i\theta_2} - e^{i(\phi - \pi/3)} \end{cases}$$

Multiplying the above equation with $e^{-i\theta_1}$ and $e^{-i\theta_2}$ respectively, we obtain

$$\begin{cases} \rho_3 e^{i(\theta_3 - \theta_1)} = e^{-i(\theta_1 + 2\pi/3)} + \rho_1 - e^{i(\phi - \theta_1 - 2\pi/3)} \\ \rho_3 e^{i(\theta_3 - \theta_2)} = e^{-i(\theta_2 + \pi/3)} + \rho_2 - e^{i(\phi - \theta_2 - \pi/3)} \end{cases} \tag{1}$$

Taking the imaginary part of Eq. (1), we have

$$\begin{cases} \rho_3 \sin(\theta_3 - \theta_1) = -\sin(\theta_1 + 2\pi/3) - \sin(\phi - \theta_1 - 2\pi/3) \\ \rho_3 \sin(\theta_3 - \theta_2) = -\sin(\theta_2 + \pi/3) - \sin(\phi - \theta_2 - \pi/3) \end{cases} \tag{2}$$

3 Forward Kinematics

Since x and y can be readily determined for a specified ρ_3, the forward kinematics is reduced to the solution of Eq. (2) for ρ_3 and ϕ.

Rewriting Eq. (2), we have

$$\begin{cases} \sin(\phi - \theta_1 - 2\pi/3) = -\rho_3 \sin(\theta_3 - \theta_1) - \sin(\theta_1 + 2\pi/3) \\ \sin(\phi - \theta_2 - \pi/3) = -\rho_3 \sin(\theta_3 - \theta_2) - \sin(\theta_2 + \pi/3) \end{cases} \tag{3}$$

i.e.

$$R \begin{bmatrix} \sin\phi \\ \cos\phi \end{bmatrix} = - \begin{bmatrix} \rho_3 \sin(\theta_3 - \theta_1) \\ \rho_3 \sin(\theta_3 - \theta_2) \end{bmatrix} - \begin{bmatrix} \sin(\theta_1 + 2\pi/3) \\ \sin(\theta_2 + \pi/3) \end{bmatrix} \tag{4}$$

where

$$R = \begin{bmatrix} \cos(\theta_1 + 2\pi/3) & -\sin(\theta_1 + 2\pi/3) \\ \cos(\theta_2 + \pi/3) & -\sin(\theta_2 + \pi/3) \end{bmatrix} \tag{5}$$

Equation (4) can be regarded as a set of linear equations in $\sin\phi$ and $\cos\phi$. Therefore, it can be solved according to the following two cases:

Case 1

$$\det(R) = \sin(\theta_1 - \theta_2 + \pi/3) \neq 0 \tag{6}$$

Case 2

$$\det(R) = \sin(\theta_1 - \theta_2 + \pi/3) = 0 \tag{7}$$

3.1 Case 1

Since Eq. (6) is satisfied, we obtain the following equation by solving Eq. (4):

$$\begin{cases} \sin\phi = \rho_3 \det(S)/\det(R) \\ \cos\phi = 1 + \rho_3 \det(T)/\det(R) \end{cases} \tag{8}$$

where

$$S = \begin{bmatrix} \sin(\theta_3 - \theta_1) & \sin(\theta_1 + 2\pi/3) \\ \sin(\theta_3 - \theta_2) & \sin(\theta_2 + \pi/3) \end{bmatrix} \tag{9}$$

$$T = \begin{bmatrix} \cos(\theta_1 + 2\pi/3) & -\sin(\theta_3 - \theta_1) \\ \cos(\theta_2 + \pi/3) & -\sin(\theta_3 - \theta_2) \end{bmatrix} \tag{10}$$

Substitution of Eq. (8) into $\sin^2\phi + \cos^2\phi = 1$, we obtain

$$a_2 \rho_3^2 + a_1 \rho_3 = 0 \tag{11}$$

where

$$a_2 = \det^2(S) + \det^2(T) \tag{12}$$

$$a_1 = 2\sin(\theta_1 - \theta_2 + \pi/3)\det(T) = 2\det(R)\det(T) \tag{13}$$

Equation (11) can be solved according to the following three sub-cases.

Case 1a

$$\begin{cases} a_2 = 0 \\ a_1 = 0 \end{cases} \tag{14}$$

Case 1b

$$\begin{cases} a_2 \neq 0 \\ a_1 = 0 \end{cases} \tag{15}$$

Case 1c

$$a_1 \neq 0 \tag{16}$$

3.1.1 Case 1a

Equation (14) is equivalent to

$$\begin{cases} \det(S) = 0 \\ \det(T) = 0 \end{cases} \tag{17}$$

Substituting Eq. (17) into Eqs. (12) and (13), we obtain that $a_1 = 0$ and $a_2 = 0$. From Eq. (11), we obtain that there are infinite solutions for ρ_3.

Substituting Eq. (17) into Eq. (8), we have

$$\begin{cases} \sin\phi = 0 \\ \cos\phi = 1 \end{cases} \tag{18}$$

i.e.,

$$\phi = 0 \tag{19}$$

Substituting Eq. (18) into Eq. (3), we obtain

$$\begin{cases} \rho_3 \sin(\theta_3 - \theta_1) = 0 \\ \rho_3 \sin(\theta_3 - \theta_2) = 0 \end{cases} \tag{20}$$

i.e.,

$$\begin{cases} \sin(\theta_3 - \theta_1) = 0 \\ \sin(\theta_3 - \theta_2) = 0 \end{cases} \tag{21}$$

In summary, in Case 1a in which Eq. (21) is satisfied, there are infinite solutions to the forward kinematics, and the moving platform undergoes self-motion – translation along $A_i B_i$ with $\phi = 0$ (Figure 2a).

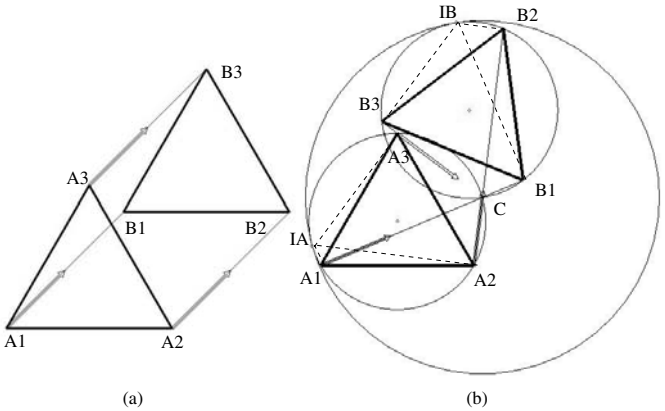


Fig. 2 Forward kinematics of 3-RPR planar parallel manipulator with infinite solutions: (a) Case 1a translation as selfmotion, and (b) Case 2a Cardanic motion as selfmotion.

3.1.2 Case 1b

Equation (15) is equivalent to

$$\begin{cases} \det(S) \neq 0 \\ \det(T) = 0 \end{cases} \tag{22}$$

Solving Eq. (11), one obtain the following double solution to ρ_3:

$$\rho_3 = 0 \tag{23}$$

Substituting the above equation into Eq. (8), we have

$$\begin{cases} \sin\phi = 0 \\ \cos\phi = 1 \end{cases} \tag{24}$$

i.e.,

$$\phi = 0 \tag{25}$$

Therefore, in Case 1b, in which Eqs. (6) and (22) are satisfied, there is a double solution to the forward kinematics: the moving platform coincides with the base (Figure 3a).

3.1.3 Case 1c

Equation (16) is equivalent to

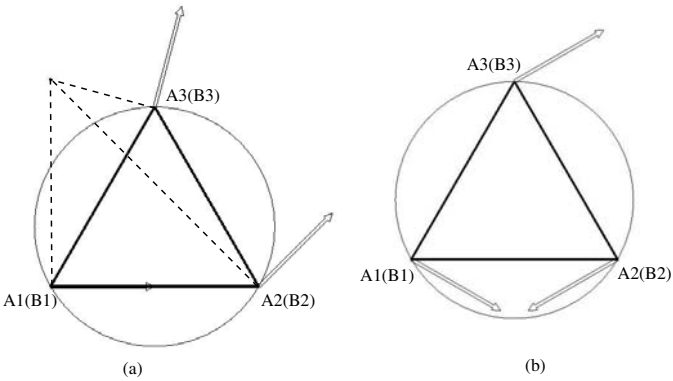


Fig. 3 Forward kinematics of 3-RPR planar parallel manipulator with a double solution: (a) Case 1b, and (b) Case 2b.

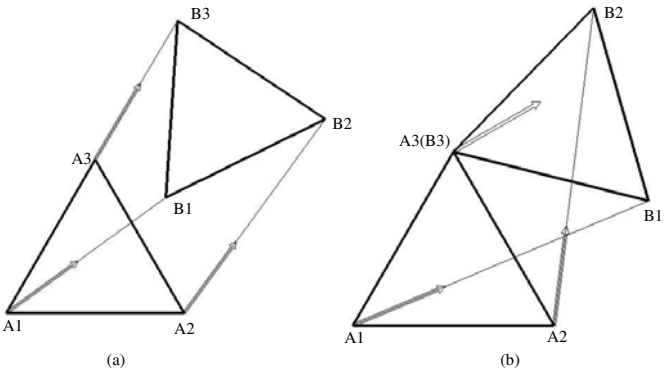


Fig. 4 Forward kinematics of 3-RPR planar parallel manipulator with two distinct solutions: (a) Case 1c, and (b) Case 2b.

$$\det(T) \neq 0 \tag{26}$$

Solving Eq. (11), we obtain two distinct solutions for ρ_3:

$$\rho_3 = 0 \tag{27}$$

$$\rho_3 = -a_1/a_2 \tag{28}$$

For each value of ρ_3 (Eqs. (27) and (28)), one unique value for ϕ can be obtained using Eq. (8).

In Case 1c (Figure 4a) in which Eqs. (6) and (26) are satisfied, there are two distinct solutions to the forward kinematics (Eqs. (27) and (28) and (8)).

3.2 Case 2

In this case, Eq.(7) is satisfied. Substituting Eq. (7) into the second equation of (3), it becomes

$$\begin{aligned}&\sin(\phi-\theta_1-2\pi/3)\cos(\theta_1-\theta_2+\pi/3)\\&\quad=-\rho_3\sin(\theta_3-\theta_2)-\sin(\theta_1+2\pi/3)\cos(\theta_1-\theta_2+\pi/3)\end{aligned}\tag{29}$$

Comparing Eq.(29) with the first equation of (3), we obtain that Eq. (3) have solutions if

$$\rho_3[\sin(\theta_3-\theta_2)-\sin(\theta_3-\theta_1)\cos(\theta_1-\theta_2+\pi/3)]=0\tag{30}$$

Equation (30) leads to

$$\sin(\theta_3-\theta_2)=\sin(\theta_3-\theta_1)\cos(\theta_1-\theta_2+\pi/3)\tag{31}$$

or

$$\begin{cases}\sin(\theta_3-\theta_2)\neq\sin(\theta_3-\theta_1)\cos(\theta_1-\theta_2+\pi/3)\\ \rho_3=0\end{cases}\tag{32}$$

Therefore, the forward kinematics in Case 2 can be further divided into two subcases: Case 2a – Equation (31) is satisfied, and Case 2b – Equation (32) is satisfied.

3.2.1 Case 2a

If Eq. (31) is satisfied, the two equations in (3) are identical. In this case, the moving platform can undergo a finite motion which can be described using either the first (or the second) equation of (3), which is in fact the Cardanic motion (Figure 2b).

3.2.2 Case 2b

Substitution of Eq. (32) into the first equation of (3), we have

$$\sin(\phi-\theta_1-2\pi/3)=-\sin(\theta_1+2\pi/3)\tag{33}$$

which leads two solutions if $\theta_1\neq-\pi/6$ and $5\pi/6$:

$$\phi=0\tag{34}$$

$$\phi=2\theta_1+\pi/3\tag{35}$$

or a double solution if $\theta_1=-\pi/6$ or $5\pi/6$:

$$\phi=0.\tag{36}$$

3.3 Summary

From the previous sections, it is learned that there may be infinite solutions, a double solution or two distinct solutions to the forward kinematics for different cases of inputs. In this section, the above results will be summarized. Geometric characteristics for different cases will also be revealed. From brevity, $A_iB_i^{\perp}$ is used to represent the perpendicular to A_iB_i passing through A_i, and circumcircle A is used to denote the circumcircle of points A_1, A_2 and A_3.

1. Equations (7) and (31) are satisfied (Case 2a).
   Considering that the base is an equilateral triangle, we can prove using the property of inscribed angles in the circumcircle A that Eq. (7) means lines A_1B_1 and A_2B_2 intersect at a common point, denoted by C, on the circumcircle A. From Eq. (31), we obtain that A_3B_3 should be parallel to one and only one of the two angle bisectors of the angles formed by lines A_1B_1 and A_2B_2. Using again the property of inscribed angles, we obtain that A_3B_3 passes through C. Therefore, if Eqs. (7) and (31) are satisfied, all the lines A_iB_i intersect at a common point, C, on the circumcircle A, and if the common point coincide with A_i, then line A_iB_i must be tangent to the circumcircle A. We can prove that the reverse is also true.
   Similarly, we can prove that except for $\theta_1 = -\pi/6$ or $5\pi/6$, all the $A_iB_i^{\perp}$ intersect at a common point on the circumcircle A if and only if Eqs. (7) and (31) are satisfied. If $\theta_1 = -\pi/6$ or $5\pi/6$, all the $A_iB_i^{\perp}$ intersect at A_3, there is no guarantee that (31) is satisfied.
   In summary, all the lines A_iB_i intersect at a common point on the circumcircle A, and if the common point coincides with A_i, then line A_iB_i must be tangent to the circumcircle A. The moving platform can undergo finite motion – Cardanic motion.
2. Equation (21) is satisfied (Case 1a).
   In this case, all lines A_iB_i are parallel. There are infinite solutions to the forward kinematics, and the moving platform undergoes self-motion – translation along A_iB_i with $\phi = 0$ (Figure 2a).
3. The case which satisfies (Case 1c)

$$\begin{cases} \det(R) \neq 0 \\ \det(S) \neq 0 \\ \det(T) = 0 \end{cases} \tag{37}$$

   or (Case 2b)

$$\begin{cases} \sin(\theta_1 - \theta_2 + \pi/3) = 0 \\ \sin(\theta_3 - \theta_2) \neq \sin(\theta_3 - \theta_1)\cos(\theta_1 - \theta_2 + \pi/3) \\ \theta_1 = -\pi/6 \quad \text{or} \quad 5\pi/6 \end{cases} \tag{38}$$

Equation (37) means, geometrically, that all the $A_iB_i^{\perp}$ intersect at (a) A_1 or A_2 if not all A_iB_i intersect at a point on the circumcircle A, and (b) a finite point not on the circumcircle A. Equation (38) means that all the $A_iB_i^{\perp}$ intersect at A_3 and not all A_iB_i intersect at a point on the circumcircle A.
In summary, in this case, all the $A_iB_i^{\perp}$ intersect at (a) A_i if not all A_iB_i intersect at a point on the circumcircle A, and (b) a finite point not on the circumcircle A. There is a double solution to the forward kinematics.

4. The other cases, there are two distinct solutions to the forward kinematics.

4 Type 2 Singularity Analysis

Type 2 singularity (also uncertainty singularity) occurs when the structure that is obtained from the parallel manipulator by blocking all its actuated joints has a positive instantaneous (or full-cycle) DOF [4].

Taking all θ_i as constants and differentiating Eq. (2), we have

$$\begin{cases} \sin(\theta_3-\theta_1)\delta\rho_3 = -\cos(\phi-\theta_1-2\pi/3)\delta\phi \\ \sin(\theta_3-\theta_2)\delta\rho_3 = -\cos(\phi-\theta_2-\pi/3)\delta\phi \end{cases} \tag{39}$$

Thus, Type 2 singularities happen if and only if

$$\begin{vmatrix} \sin(\theta_3-\theta_1) & \cos(\phi-\theta_1-2\pi/3) \\ \sin(\theta_3-\theta_2) & \cos(\phi-\theta_2-\pi/3) \end{vmatrix} = 0 \tag{40}$$

i.e.

$$\det(S)\sin\phi + \det(T)\cos\phi = 0 \tag{41}$$

One can verify that Eq. (41) is satisfied in the cases that there are infinite solutions or a double solution to the forward kinematics. Equation (41) is not satisfied in the cases that there are two distinct solutions to the forward kinematics. Due to space limitation, only the non-trival solution of two distinct solutions (Eqs. (8) and (28)) in Cases 1c will be discussed in the following.

Substitution of Eqs. (8), (28), (12) and (13) into $\det(S)\sin\phi + \det(T)\cos\phi$, we obtain using Eqs. (26)

$$\det(S)\sin\phi + \det(T)\cos\phi = -\det(T) \neq 0 \tag{42}$$

Therefore, the manipulator is not singular at the non-trival solution of the two distinct solutions in Cases 1c.

In summary, singularities happen in three cases:

1. All the lines A_iB_i intersect at a common point, C, on the circumcircle A (Figure 2b), and if the common point coincide with A_i, then line A_iB_i must be tangent to the circumcircle A.

In this case, the moving platform can undergo finite motion – Cardanic motion. All the perpendiculars to A_iB_i passing through B_i (A_i) also intersect at a common point, IB(IA), on the circumcircle of points B_1, B_2 and B_3 (A_1, A_2 and A_3).

2. Lines A_iB_i ($i = 1,\ 2,\ 3$) are parallel (Figure 2a).
   In this case, the moving platform can undergo finite motion – translation along lines A_iB_i.
3. All $A_iB_i^{\perp}$ intersect at (a) A_i if not all A_iB_i intersect at a point on the circumcircle A, and (b) a finite point not on the circumcircle A. In this case, there is a double solution to the forward kinematics (Figure 3). The moving platform has an instantaneous DOF.

We have thus proved that the result, given in [1] without detailed proof, that the only Type 2 singular configurations, for which the base and the moving platform do not coincide, belong to self-motions.

5 Summary

A systematic approach to the forward kinematics has been presented for the class of 3-RPR planar parallel manipulator with congruent equilateral base and moving platform. The conditions on the inputs under which this planar parallel manipulator has different number of solutions to the forward kinematics, including a double solution, two distinct solutions or a set of continuous motion, have been identified. Type 2 singularity analysis has also been investigated systematically. The result, given in [1] without detailed proof, regarding the Type 2 singularities with a non-zero orientation angle has been proved.

Although it is not as elegant as the geometric approach presented in [1], the algebraic approach presented in this paper is rigorous. Some results, such as the conditions on the inputs under which this planar parallel manipulator has a double solution, might be hard to obtain using the geometric approach.

References

1. Chablat, D., Wenger, P. and Bonev, I.A., Self motions of a special 3RPR planar parallel robot. In *Advances in Robot Kinematics*, J. Lenarčič and B. Roth (Eds.), pp. 221–228, Springer (2006).
2. Merlet, J.P., Direct kinematics of planar parallel manipulators. In *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, Vol. 4, pp. 3744–3749, Institute of Electrical and Electronics Engineers (1996).
3. Kong X., Displacement analysis of a 3-RP Assur III group. *Mechanical Design and Research*, **4**, 11–12 (1994) [in Chinese].
4. Kong, X. and Gosselin, C.M., Uncertainty singularity analysis of parallel manipulators based on the instability analysis of structures. *The International Journal of Robotics Research*, **20**(11), 847–856 (2001).

Geometric Algebra Approach to Singularity of Parallel Manipulators with Limited Mobility

Tanio K. Tanev

Central Laboratory of Mechatronics and Instrumentation, Bulgarian Academy of Sciences, Acad. G. Bonchev Str., Bl. 1, 1113 Sofia, Bulgaria, e-mail: tanev_tk@hotmail.com

Abstract. Geometric algebra is employed for the analysis of the singularity of parallel manipulators with limited mobility. The rotations and translations of vectors and screws are performed in the degenerate geometric algebra $G_{3,0,1}$. The condition for singularity is obtained using the language of geometric algebra. The approach is applied to two parallel manipulators with limited mobility.

Key words: parallel manipulator, geometric algebra, kinematics, singularity.

1 Introduction

This paper presents an application of geometric algebra for the analysis of the singularity of parallel manipulators that do not have full mobility, i.e. spatial parallel manipulators with less than six degrees of freedom (dof) or planar ones with less than three dof. The analysis of the singular configurations of the parallel manipulators is an essential part of the process of design and control. In a singular configuration the moving platform of the parallel manipulator has an uncontrollable instant mobility and the manipulator can not sustain a certain wrench applied to the moving platform.

Recently, some "non-standard" methods have been introduced to robot kinematics. For example, the Grassmann geometry was used by Merlet (1989) and the Grassmann–Cayley algebra has been applied to robotics by several researchers: White (1994) analysed the motion of serial robot using Grassmann–Cayley algebra. The Grassmann–Cayley algebra was employed in Staffetti and Thomas (2000) and Ben-Horin and Shoham (2006). The Clifford algebra was used in Collins and McCarthy (1998) and Selig (2000). In Zamora-Esquivel and Bayro-Corrochano (2006) and in Tanev (2006) the geometric algebra was applied.

The Grassmann and Clifford algebras were created in the 19th century. In the second half of the 20th century Clifford algebras have been "rediscovered" and further developed into a unified language named "geometric algebra" in Hestenes (1999), Lasenby et al. (2000), Dorst and Mann (2002), and some other authors.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 39–48.

In this paper, the geometric algebra is used for obtaining the singularity conditions for parallel manipulators with fewer than six degrees of freedom. This approach is applied to two parallel manipulators – a simple planar one and a spatial 5-dof parallel manipulator.

2 Kinematics of Parallel Manipulators Using Geometric Algebra

The different types of geometric algebra distinguished by the different signatures can be denoted by $G_{p,g,r} = G(p, q, r)$. This geometric algebra has $n = q + q + r$ orthonormal basis vectors e_i $(i = 1, \ldots, n)$ which obey the following rule:

$$e_i \cdot e_j = \begin{cases} 1, & i = j = 1, \ldots, p, \\ -1, & i = j = p+1, \ldots, p+q, \\ 0, & i = j = p+q+1, \ldots, p+q+r, \\ 0, & i \neq j, \end{cases}$$

$$e_i \wedge e_i = 0. \tag{1}$$

In this paper, the transformations for the kinematic analysis are performed in the degenerate geometric algebra (Hestenes et al., 1999). In this case, the translation can be represented as a spinor:

$$T = e^{e_0 \mathbf{a}/2} = 1 + \frac{1}{2} e_0 \mathbf{a}, \tag{2}$$

where e_0 $(e_0 \cdot e_0 = 0)$ is a null vector orthogonal to R^3; $\mathbf{a}$ is a vector in G_3.

Here, the point x is represented as a trivector in $G_{3,0,1}$ similar to the form given in Selig (2000), i.e.,

$$x = (1 + e_0 \mathbf{x}) I_3, \tag{3}$$

where $I_3 = e_1 e_2 e_3$ is the unit pseudoscalar of G_3; $\mathbf{x} = a_1 e_1 + a_2 e_2 + a_3 e_3$ is a vector in G_3.

The points denoted as italic characters are represented by vectors in $G_{3,0,1}$, and points denoted as boldface characters are represented by vectors in G_3.

The rigid displacement can be written in spinor representation, i.e.,

$$Q = T R, \tag{4}$$

where the spinor

$$R = e^{-(1/2) I_3 \mathbf{a}} = \cos\left(\frac{1}{2}\mathbf{a}\right) - I_3 \sin\left(\frac{1}{2}\mathbf{a}\right).$$

Thus, the linear transformation is written as:

$$Q(x) = QxQ^{\dagger}, \tag{5}$$

where $Q^{\dagger} = R^{\dagger}T^{\dagger}$ is the reverse of Q.

This representation has the great advantage of reducing the group composition to the geometric product.

The screw axes (lines) of the joints can be expressed in the geometric algebra as follows: any oriented line $\mathbf{L}$ is uniquely determined by given its direction $\mathbf{u}$ and its moment $\mathbf{m}$ and in the geometric algebra G_3 of 3-D vector space V^3 with the basis $\{e_1, e_2, e_3\}$ it can be written as (Hestenes, 1999):

$$\begin{aligned}\mathbf{L} &= \mathbf{u} + \mathbf{m} \\ &\equiv \mathbf{u} + \mathbf{r} \wedge \mathbf{u} = u_1e_1 + u_2e_2 + u_3e_3 + m_1e_2 \wedge e_3 + m_2e_3 \wedge e_1 + m_3e_1 \wedge e_2,\end{aligned} \tag{6}$$

where $\mathbf{r}$ is the position vector of a point on the line; u_i $(i = 1, 2, 3)$ and m_i $(i = 1, 2, 3)$ are scalar coefficients.

The transformation of a line can be performed in the same way as a vector (see Selig, 2000) and for that reason the line can be written as a bivector in $G_{3,0,1}$, i.e.,

$$\begin{aligned}L^{(4)} &= (\mathbf{u} + \mathbf{m}e_0)I_3 \\ &= u_1e_2 \wedge e_3 + u_2e_3 \wedge e_1 + u_3e_1 \wedge e_2 + m_1e_1 \wedge e_0 + m_2e_2 \wedge e_0 + m_3e_3 \wedge e_0,\end{aligned} \tag{7}$$

where the superscript (4) indicates that the screw is written in $G_{3,0,1}$.

A general screw can be expressed in $G_{3,0,1}$ in a similar way, i.e.,

$$S^{(4)} = v_1e_2 \wedge e_3 + v_2e_3 \wedge e_1 + v_3e_1 \wedge e_2 + b_1e_1 \wedge e_0 + b_2e_2 \wedge e_0 + b_3e_3 \wedge e_0, \tag{8}$$

where v_i $(i = 1, 2, 3)$ and b_i $(i = 1, 2, 3)$ are scalar coefficients.

Then, the transformation of a screw (or line) can be written as

$$S^{(4)} = QS^{(4)}Q^{\dagger}. \tag{9}$$

2.1 Velocity

In this section, the screws are expressed as vectors in G_6. In other words, in the geometric algebra G_6 of 6-D vector space V_6 with the basis $\{e_1, e_2, e_3, e_4, e_5, e_6\}$, a screw can be written as a vector (grade 1), i.e.,

$$S = v_1e_1 + v_2e_2 + v_3e_3 + b_1e_4 + b_2e_5 + b_3e_6, \tag{10}$$

where the coefficients are the same as in Eq. (8).

The following notation of a screw is adopted: an upper case letter without superscript (S, D) denotes a screw written as a vector in G_6 of 6-D space, otherwise a superscript indicates the type of the geometric algebra in which the screw is de-

scribed; letters with a tilde mark ($\tilde{S}$) denote the elliptic polars of the screws S (Lipkin and Duffy, 1985).

The moving platform and the base of a parallel manipulator are connected with n-legs, which can be considered as serial chains. The velocity of the moving platform can be expressed as a linear combination of the joint instantaneous twists (for example, see Rico and Duffy, 1996):

$$V_p = \sum_{i=1}^{f} {}^{j}\omega_i \, {}^{j}S_i, \quad j = 1, 2, \ldots, n, \tag{11}$$

where ${}^{j}\omega_i$ denotes the joint rate and ${}^{j}S_i$ represents the normalized screw associated with the ith joint axis of the jth leg; f is the dof of the jth leg. The left leading superscript denotes the leg number.

A leg with full mobility and a leg with less than six dof could be treated in a similar way. For that reason the necessary extra dummy joints are added to the leg with less than six dof so that it becomes a leg with full mobility. The dummy joints are considered as driven but locked ones. Then, taking the outer product of five screws of the jth leg gives the following 5-blade:

$${}^{j}A_k = {}^{j}S_1 \wedge {}^{j}S_2 \wedge \cdots \wedge {}^{j}S_{k-1} \wedge {}^{j}S_{k+1} \wedge \cdots \wedge {}^{j}S_6, \tag{12}$$

where ${}^{j}S_i$ are the normalized joint axes of the jth leg.

The 5-blade ${}^{j}A_k$ from Eq. (12) involves five screws (out of six with the exception of the ${}^{j}S_k$ screw), where the kth joint is active. In a non-degenerate space, the dual of a blade represents the orthogonal complement of the subspace represented by the blade. The dual of the above 5-blade ${}^{j}A_k$ is given by the following geometric product:

$${}^{j}D_k = {}^{j}A_k I_6^{-1} = (-1)^{n(6-n)} I_6^{-1} \, {}^{j}A_k, \tag{13}$$

where $I_6 = e_1e_2e_3e_4e_5e_6$ is a unit pseudoscalar of the G_6 and I_6^{-1} is its inverse; $n = 5$ (in case of 6-dof limb).

Pre-multiplying (inner product) both sides of Eq. (11) by ${}^{j}D_k$ one obtains:

$${}^{j}\omega_k = \frac{1}{{}^{j}D_k \cdot {}^{j}S_k} \, {}^{j}D_k \cdot V_p \quad \text{or} \quad {}^{j}\omega_k = \frac{1}{{}^{j}\tilde{R}_k \cdot {}^{j}S_k} \, {}^{j}R_k \cdot \tilde{V}_p, \tag{14}$$

where ${}^{j}R_k \equiv {}^{j}\tilde{D}_k$ is a screw reciprocal to the joint screws ${}^{j}S_1, {}^{j}S_2, \ldots, {}^{j}S_{k-1}, \ldots, {}^{j}S_6$, and $\tilde{V}_p$ is the velocity of the moving platform with interchanged primary and secondary parts.

The result in Eq. (14) is obtained having in mind that ${}^{j}S_i \cdot {}^{j}D_k = 0$ ($i \neq k$) and ${}^{j}S_k \cdot {}^{j}D_k = {}^{j}c_k$ (providing the joint screws of the jth leg are linearly independent); ${}^{j}c_k$ is a scalar.

3 Wrenches of Constraints and Singularity

Let n be the number of the manipulator legs and m ($m = 6 - q$) be the degrees of freedom of the parallel manipulator. We suppose that the remaining q degrees of freedom are represented by dummy joints (or driven but locked joints) and associated with them dummy screws. So, in non-singular configuration the driven joints and the geometry (or the dummy joints) of the manipulator sustain a general wrench applied to the moving platform. Therefore, the singular configuration can occur when all dual (or reciprocal, respectively) screws ${}^{j}D_k$ (${}^{j}R_k$) from Eq. (14), representing active and dummy joints, are linearly dependent. Using the language of the geometric algebra, the condition of singularity for the parallel manipulator with less than six degrees of freedom (but with dummy joints) can be expressed in the following way

$$D_{a_1} \wedge \cdots \wedge D_{a_k} \wedge D_{d_1} \wedge \cdots \wedge D_{d_q} = 0, \quad k + q = 6, \tag{15}$$

where D_{a_i} is a dual vector (grade 1-blade) associated to the ith active joint and D_{d_i} is a dual vector (grade 1-blade) associated to the ith dummy joint. In this case each leg has a full mobility. Here the dummy joints are considered as active but locked.

In the following sections, the approach is applied to two particular parallel manipulators.

3.1 Example of Four-Bar Mechanism as a One-dof Parallel Manipulator

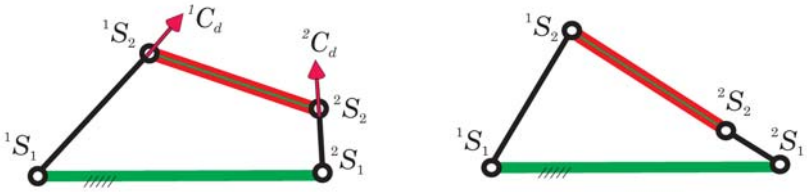
Firstly, in order to illustrate the approach, it is applied to a very simple example, i.e. to the four-bar mechanism (Figure 1a), whose singular configurations are known (e.g., Zlatanov et al., 2002). It is considered as a planar parallel manipulator with two RR-legs and the coupler as a moving platform. The mechanism has one driven joint with the joint axis ${}^{1}S_1$. In order to have full mobility, we suppose that one dummy joint is added to each leg. Then, the duals corresponding to the active and dummy joints for the first and the second leg, respectively, are as follows:

$${}^{1}D_1 = ({}^{1}S_2 \wedge {}^{1}S_d \wedge e_{126})I_6^{-1}; \quad {}^{1}D_d = ({}^{1}S_1 \wedge {}^{1}S_2 \wedge e_{126})I_6^{-1}, \tag{16}$$

$${}^{2}D_d = ({}^{2}S_1 \wedge {}^{2}S_2 \wedge e_{126})I_6^{-1}, \tag{17}$$

where $e_{126} = e_1 \wedge e_2 \wedge e_6$ is a 3-blade representing the restricting subspace, i.e., it restricts the mechanisms to move only in the horizontal (X–Y) plane; ${}^{j}D_d$ denotes the dual corresponding to the dummy joint and ${}^{j}S_d$ is a screw associated with the dummy joint.

After applying some identities of the geometric algebra (see Hestenes et al., 1999) one obtains



(a) The manipulator and the constraint forces

(b) A singular configuration

Fig. 1 Four-bar mechanism (one-dof parallel manipulator).

$$^1D_1 \wedge {}^1D_d = c(^1S_2 \wedge e_{126})I_6^{-1}, \tag{18}$$

where $c = {}^1D_1 \cdot {}^1S_1$ is a scalar.

The blade from Eq. (18) is a blade of non-freedom for the first leg. In this case, the wrenches of constraint associated with the dummy joints for the first and the second leg, respectively, are $^1C_d = {}^1\tilde{D}_d$, $^2C_d = {}^2\tilde{D}_d$ (derived from Eqs. (16) and (17)) and the third one can be obtained by factoring the 2-blade from Eq. (18).

In this case the wrenches of constraints are pure forces, Notice, that the two constraint forces ^1C_d and ^2C_d, associated with the dummy joints, are unique (along the legs, Figure 1a).

The condition for singularity of the manipulator can be written as

$$^1D_1 \wedge {}^1D_d \wedge {}^2D_d = 0 \quad \text{or} \quad {}^1\tilde{D}_1 \wedge {}^1\tilde{D}_d \wedge {}^2\tilde{D}_d = 0. \tag{19}$$

Again, applying the identities of the geometric algebra and keeping in mind Eq. (18), the left-hand blade (the singularity condition) from Eq. (19) becomes

$$\begin{aligned} &[c(^1S_2 \wedge e_{126})I_6^{-1}] \wedge [(^2S_1 \wedge {}^2S_2 \wedge e_{126})I_6^{-1}] \\ &= -c(^2S_1 \wedge {}^2S_2 \wedge e_{126} \wedge {}^1S_2)I_6^{-1}e_{126}I_6^{-1} \\ &= c(^2S_1 \wedge {}^2S_2 \wedge {}^1S_2 \wedge e_{126})e_{126} = 0. \end{aligned} \tag{20}$$

Therefore, bearing in mind that $e_1 \wedge e_2 \wedge e_6 \neq 0$, it is clear from Eq. (20) that the condition for singularity can be written as

$$^2S_1 \wedge {}^2S_2 \wedge {}^1S_2 = 0. \tag{21}$$

Eq. (21) implies that the mechanism is in singular configuration if the three lines (joint axes, which are parallel) are linearly dependent, i.e., lie in a single plane, defined by any two of the lines (Figure 1b). Eq. (21) involves only the screw axis of the passive joints. Therefore, in case of changing the driven joint (for example from $^1S_1$ to $^2S_1$), the configuration shown in Figure 1b will be no longer singular.

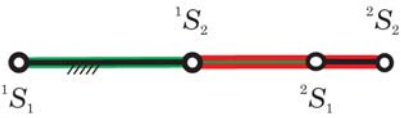


Fig. 2 Constraint singular configuration of the four-bar mechanism.

If we consider the blade formed only by the duals associated with the dummy joints, the condition for the so-called constraint singularity (the term introduced by Zlatanov et al., 2002) can be obtained, i.e.,

$$^1D_d \wedge {}^2D_d = 0 \quad \text{or} \quad {}^1\tilde{D}_d \wedge {}^2\tilde{D}_d = 0. \tag{22}$$

Therefore, the constraint singularity occurs when these two lines $^1C_d = {}^1\tilde{D}_d$ and $^2C_d = {}^2\tilde{D}_2$ coincide (Figure 2).

From Eqs. (19) and (22) it can be seen that the constraint singularity is a subset of general singularity. It is clear from Eqs. (16), (17) and (22) that the condition for constraint singularity involves all joint axes, which fact implies that the mechanism remains in constraint singular configuration even when the driven joint is changed.

3.2 Singularity of a 5-dof Parallel Manipulator

In this section a type of parallel manipulator with five degrees of freedom is introduced and its singular configurations are analyzed using the geometric algebra. The considered parallel manipulator has four legs; the first leg has RRPRR structure and the other three legs have identical SPS structure (Figure 3). The driven (active) joints are the four prismatic joints of the legs and a revolute joint R attached to the base of the RRPRR leg. In this case, the RRPRR (R $\perp$ R $\perp$ P $\perp$ R $\perp$ R) leg has two driven (active) joints: the first one (R – revolute joint attached to the base) and the prismatic joint (P).

The SPS legs have full mobility and each one has one driven joint (the P joint). In this case the SPS (or UPS) leg has only one possible dual screw, or reciprocal screw to the joint axis associated with the U and S joints. This reciprocal screw is a line along the SPS leg. The RRPRR leg has five degrees of freedom and in order to have full mobility one extra dummy joint (denoted by a superscript d in the equations) is added, which can be considered as active but locked. The dual screws associated with the active and dummy joints of the first (RRPRR) leg are as follows:

$$^1D_1 = (^1S_2 \wedge {}^1S_3 \wedge {}^1S_4 \wedge {}^1S_5 \wedge {}^1S_d)I_6^{-1}, \tag{23}$$

$$^1D_3 = (^1S_1 \wedge {}^1S_2 \wedge {}^1S_4 \wedge {}^1S_5 \wedge {}^1S_d)I_6^{-1}, \tag{24}$$

$$^1D_d = (^1S_1 \wedge {}^1S_2 \wedge {}^1S_3 \wedge {}^1S_4 \wedge {}^1S_5)I_6^{-1}, \tag{25}$$

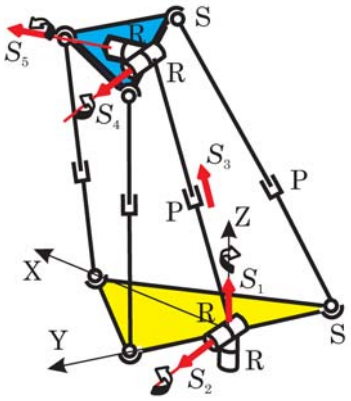


Fig. 3 The 5-dof parallel manipulator.

where ${}^1S_1$ is the axis of the joint attached to the base and ${}^1S_5$ is the axis of the joint connected to the moving platform of the RRPRR leg (Figure 3).

Applying the identities of the geometric algebra for the outer product of the duals from Eqs. (23), (24) and (25) one obtains:

$$D_{13d} = {}^1D_1 \wedge {}^1D_3 \wedge {}^1D_d = \lambda({}^1S_2 \wedge {}^1S_4 \wedge {}^1S_5)I_6^{-1}, \tag{26}$$

where $\lambda = ({}^1S_1 \cdot {}^1D_1)({}^1S_d \cdot {}^1D_d)$ is a scalar; the above result is obtained bearing in mind that ${}^1D_i \cdot {}^1S_k = 0\ (i \neq k)$ and ${}^1D_i \cdot {}^1S_k \neq 0\ (i = k)$.

In fact, the 3-blade from Eq. (26) is a blade of non-freedom for the RRPRR leg. One of the wrenches of constraints (${}^1C_d = {}^1\tilde{D}_d$) is uniquely defined by Eq. (25).

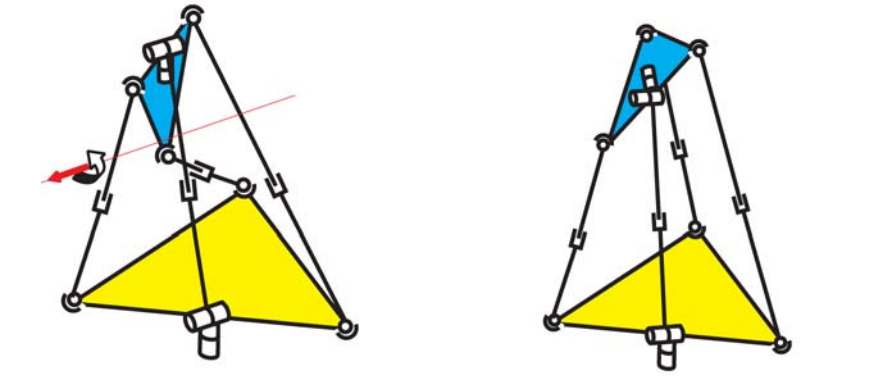
The algebraic condition for singularity can be written as follows:

$$\left\{[({}^1S_2 \wedge {}^1S_4 \wedge {}^1S_5)I_6^{-1}] \wedge {}^2D \wedge {}^3D \wedge {}^4D\right\}I_6^{-1} = 0, \tag{27}$$

where ${}^2D = {}^2\tilde{C}$, ${}^3D = {}^3\tilde{C}$ and ${}^4D = {}^4\tilde{C}$ are duals associated with the three SPS legs: ${}^jD = {}^jS_1 \wedge {}^jS_2 \wedge {}^jS_4 \wedge {}^jS_5 \wedge {}^jS_6$ $(j = 2, 3, 4)$. The missing joint screw axis ${}^jS_3$ is associated with the active P joint of the SPS (UPS) legs. The wrenches of constraint ${}^2C$, ${}^3C$ and ${}^4C$ for the three SPS legs can be easily obtained and in fact they are lines along the legs.

The singular configurations of the parallel manipulator can be algebraically derived from Eq. (27). Expanding Eq. (27) leads to an algebraic equation in terms of the five joint variables of the RRPRR leg (all joint screws of the parallel manipulator are expressed as functions of these five variables). The solutions of this equation give the singular configuration of the manipulator. The expanded equation is not listed here because of the limited space. Several singular configurations have been identified. Two types of singular configurations are shown in Figure 4.

The uncontrollable motion of the moving platform for the first singular configuration (Figure 4a) is a pure rotation, which axis intersects all four legs, is parallel to



(a) Uncontrollable motion - pure rotation

(b) Uncontrollable motion - a general screw motion

Fig. 4 Singular configurations of the 5-dof parallel manipulator.

two R joint axes of the RRPRR leg and perpendicular to P joint axis of the RRPRR leg. In the second singular configuration (Figure 4b) the uncontrollable motion is a general screw motion.

4 Conclusions

The presented approach proves to be effective in determining the singularity condition for parallel manipulators with limited mobility. This approach is applied to two parallel manipulators, which singular configurations are obtained. It has been shown that the equation for the singularity (the condition for singularity) involves the screws which represent all and only passive joints of the manipulators. This geometric algebra approach provides a good geometrical insight and efficiency in dealing with robot kinematics and singularity of parallel manipulators with fewer than six degrees of freedom.

References

Ben-Horin, P. and Shoham, M. (2006), Singularity of a class of Gough–Stewart platforms with concurrent joints, in *Advances in Robot Kinematics*, Lenarčič J. and Roth B. (Eds), Springer, the Netherlands, pp. 265–274.

Collins, C.L. and McCarthy, J.M. (1998), The quartic singularity surfaces of planar platforms in the Clifford algebra of the projective plane, *Mechanism and Machine Theory* **33**(7), 931–944.

Dorst, L. and Mann, S. (2002), Geometric algebra: A computational framework for geometrical applications (Part 1), *IEEE Computer Graphics and Applications* **22**(3), 24–31.

Hestenes, D. (1999), *New Foundations for Classical Mechanics*, second edition, Kluwer Academic Publishers, Dordrecht, the Netherlands.

Hestenes, D., Li, H. and Rockwood, A. (1999), New algebraic tools for classical geometry, in G. Sommer (Ed.), *Geometric Computing with Clifford Algebra*, Springer, Berlin.

Lasenby, J., Lasenby, A.N. and Doran, C.J.L. (2000), A unified mathematical language for physics and engineering in the 21st century, *Philosophical Transactions of the Royal Society of London A* **358**, 21–39.

Lipkin, H. and Duffy, J. (1985), The elliptic polarity of screws, *ASME Journal of Mechanisms, Transmissions, and Automation in Design* **107**, 377–387.

Merlet, J-P. (1989), Singular configurations of parallel manipulators and Grassmann geometry, *International Journal of Robotics Research* **8**(5), 45–56.

Rico, J.M. and Duffy, J. (1996), An application of screw algebra to the acceleration analysis of serial chains, *Mechanism and Machine Theory* **31**(4), 445–457.

Selig, J.M. (2000), Clifford algebra of points, lines and planes, *Robotica* **18**, 545–556.

Staffetti, E. and Thomas, F. (2000), Analysis of rigid body interactions for compliant motion tasks using the Grassmann-Cayley algebra, in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)*, Vol. 3, pp. 2325–2332.

Tanev, T.K. (2006), Singularity analysis of a 4-dof parallel manipulator using geometric algebra, in *Advances in Robot Kinematics, Mechanism and Motion*, Lenarčič, J. and Roth, B. (Eds), Springer, Dordrecht, the Netherlands, pp. 275–284.

White, N. (1994), Grassmann–Cayley algebra and robotics, *Journal of Intelligent and Robotics Systems* **11**, 91–107.

Zamora-Esquivel, J. and Bayro-Corrochano, E. (2006), Kinematics and grasping using conformal geometric algebra, in *Advances in Robot Kinematics, Mechanism and Motion*, Lenarčič, J. and Roth, B. (Eds), Springer, Dordrecht, the Netherlands, pp. 473–480.

Zlatanov, D., Bonev, I.A. and Cosselin, C.M. (2002), Constraint singularities of parallel mechanisms, in *Proceedings International Conference on Robotics and Automation, ICRA '02*, IEEE, Vol. 1, pp. 496–502.

SinguLab – A Graphical User Interface for the Singularity Analysis of Parallel Robots Based on Grassmann–Cayley Algebra

Patricia Ben-Horin[1], Moshe Shoham[1], Stéphane Caro[2], Damien Chablat[2] and Philippe Wenger[2]

[1]*Department of Mechanical Engineering, Technion – Israel Institute of Technology, Haifa, Israel, e-mail: {patbh, shoham}@tx.technion.ac.il*
[2]*Institut de Recherche en Communications et Cybernétique de Nantes, 1, Rue de la Noe, 44321 Nantes, France, e-mail: {stephane.caro, damien.chablat, philippe.wenger}@irccyn.ec-nantes.fr*

Abstract. This paper presents SinguLab, a graphical user interface for the singularity analysis of parallel robots. The algorithm is based on Grassmann–Cayley algebra. The proposed tool is interactive and introduces the designer to the singularity analysis performed by this method, showing all the stages along the procedure and eventually showing the solution algebraically and graphically, allowing as well the singularity verification of different robot poses.

Key words: singularity, Grassmann–Cayley algebra, parallel robot, software.

1 Introduction

Singularity of parallel manipulators has been thoroughly investigated using different methods, mainly including line geometry, screw theory, and Jacobian determinants analysis. Recently, Grassmann–Cayley algebra (GCA) has been used for singularity analysis too.

SinguLab is the first version of a tool for singularity analysis of parallel robots. The aim of this user interface is to provide the designer an automatic tool for the analysis, geometric interpretation and visualization of singularities. It enables the user to determine the singularities of a large range of parallel robots and gives him some guidelines of GCA.

SinguLab was developed within the framework of SIROPA[1] – a French national project, the aim of which is to develop knowledge about the direct-kinematics singularities of parallel robots and to transmit this knowledge to the end-users – during a sojourn stay of the first author at IRCCyN.

[1] https://wiki-sop.inria.fr/wiki/bin/view/Coprin/SIROPA

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 49–58.

1.1 Grassmann–Cayley Algebra

The algorithm used in SinguLab is based on GCA. For space limitations we only introduce the basic concepts and the readers are referred to Ben-Horin and Shoham (2006a) and reference therein for further details on this topic. The basic elements of this algebra are called *extensors*, which in fact are symbolically denoted Plücker coordinates of vectors. Two basic operations that play an essential role in GCA involving extensors are the *join* and *meet* operators. The first is associated with the union of two vector spaces, and the latter has the same geometric meaning as the intersection of two vector spaces. Further, special determinants called *brackets* are also defined in GCA. The brackets, of which columns are vectors, satisfy special product relations called *syzygies*, which are useful to manipulate and compare bracket expressions. The Grassmann–Cayley algebra functions under the projective space P^d, in which points are represented by homogeneous coordinates and lines are represented by Plücker coordinates.

As mentioned above, the extensors are vectors that represent geometric entities, and are characterized by their *step*. Extensors of step 1, 2 and 3 stand for a point, a line and a plane, respectively. Assumming two extensors $\mathbf{A}$ and $\mathbf{B}$, of step k and h, respectively, defined in the d-projective space, the *join* and *meet* operations are written as follows:

$$\mathbf{A} \vee \mathbf{B} = a_1 \vee a_2 \vee \cdots \vee a_k \vee b_1 \vee \cdots \vee b_h = a_1 a_2 \ldots a_k b_1 \ldots b_h, \tag{1}$$

$$\begin{aligned} \mathbf{A} \wedge \mathbf{B} &= \sum_{\sigma} \operatorname{sgn}(\sigma)[\alpha_{\sigma(1)} a_{\sigma(2)} \ldots a_{\sigma(d-h)} b_1 \ldots b_h] a_{\sigma(d-h+1)} \ldots a_{\sigma(k)} \\ &= \sum_{\sigma} \operatorname{sgn}(\sigma)[\dot{a}_1 \dot{a}_2 \ldots \dot{a}_{d-h} b_1 \ldots b_h] \dot{a}_{d-h+1} \ldots \dot{a}_k, \end{aligned} \tag{2}$$

where the sum in Eq. (2) is taken over all permutations σ of $\{1, 2, \ldots, k\}$ such that $\sigma(1) < \sigma(2) < \cdots < \sigma(d-h)$ and $\sigma(d-h+1) < \sigma(d-h+2) < \cdots < \sigma(k)$. Incidences between geometric entities are obtained as extensors of step 0 (scalars). Some examples of incidences in 3D-space are the *meet* of four planes, the *meet* of two lines and the *meet* of a line with two planes. Three *meet* examples are written in GCA as follows:

- Meet of four planes: $abc \wedge def \wedge ghi \wedge jkl = [\dot{a}def][\dot{b}ghi][\dot{c}jkl]$.
- Meet of two lines: $ab \wedge cd = [abcd]$.
- Meet of a line and two planes: $gh \wedge abc \wedge def = [\dot{g}abc][\dot{h}def]$.

Let us consider a finite set of 1-extensors $\{a_1, a_2, \ldots, a_d\}$ defined in the d-dimensional vector space over the field Υ, V, where $a_i = x_{1,i}, x_{2,i}, \ldots, x_{d,i}$ $(1 \leq i \leq d)$. The bracket of these extensors is the determinant of the matrix, of which columns are vectors a_i $(1 \leq i \leq d)$:

$$[a_1, a_2, \ldots, a_d] = \begin{vmatrix} x_{1,1} & x_{1,2} & \ldots & x_{1,d} \\ \vdots & \vdots & \ldots & \vdots \\ x_{d,1} & x_{d,2} & \ldots & x_{d,d} \end{vmatrix}. \tag{3}$$

For example, the bracket of points a, b, c and d defined in the 3D space is written as:

$$[\mathbf{abcd}] = \begin{vmatrix} a_x & b_x & c_x & d_x \\ a_y & b_y & c_y & d_y \\ a_z & b_z & c_z & d_z \\ 1 & 1 & 1 & 1 \end{vmatrix}.$$

From a geometrical point of view, the value of this bracket represents six times the volume of the tetrahedron of vertices **a**, **b**, **c** and **d**.

2 Algorithm

The procedure behind SinguLab follows the next steps:

a. Determination of the robot structure.
b. Writing the singularity equation in terms of GCA.
c. Identification of the geometrical entities involved in the singularity condition according to the algebraic equation.
d. Depending on the entities found, the algorithm finds the geometrical condition in terms of GCA. The singularity condition, if feasible, is shown to the user by means of a geometrical statement in algebraic form with a graphical visualization of the geometric entities comprising the singularity.

2.1 Determination of the Robot Structure

The available options in this version are all the possible Gough–Stewart platforms (GSPs). There are 35 different GSPs if concurrent joints on the platform or on the base are considered (Faugere and Lazard, 1995). In the next version of SinguLab, other types of parallel robots will be analyzed by means of the method explained in Sections 2.5 and 3.

The robot structure is determined by the user with six lines as 2-extensors according to their endpoints on the platform and the base. Two concurrent joints have the same label. Once the structure is defined, a schematic of the robot appears.

2.2 Singularity Equation

The singularity analysis is performed using a coordinate-free invariant version of the Jacobian matrix determinant written in terms of GCA, which is suitable for robots of motion ruled by six pure forces, represented by six zero-pitch screws. This coordinate-free version of the Jacobian determinant was derived by McMillan and White (1991), after proposing a significantly larger expression by White (1983) eight years before.

A paradigm of robots ruled by six pure forces is the general GSP. The moving platform of GSP is connected to the base through six spherical-prismatic-universal chains, the spherical and universal joints being placed anywhere on both platforms. The coordinate-free version of the Jacobian determinant of this robot with legs ab, cd, ef, gh, ij, and kl ($a, \ldots, l$ denoting the endpoints of the lines) has 24 monomials as follows:

$$\begin{aligned}[[ab, cd, ef, gh, ij, kl]] = &-[abcd][efgi][hjkl] + [abcd][efhi][gjkl] + [abcd][efgj][hikl] \\ &- [abcd][efhj][gikl] + [abce][dfgh][ijkl] - [abde][cfgh][ijkl] - [abcf][degh][ijkl] \\ &+ [abdf][cegh][ijkl] - [abce][dghi][fjkl] + [abde][cghi][fjkl] + [abcf][dghi][ejkl] \\ &+ [abce][dghi][fikl] - [abdf][cghi][ejkl] - [abde][cghj][fikl] - [abcf][dghj][eikl] \\ &+ [abdf][cghj][eikl] + [abcg][defi][hjkl] - [abdg][cefi][hjkl] - [abch][defi][gjkl] \\ &- [abcg][defj][hikl] + [abdh][cefi][gjkl] + [abdg][cefj][hikl] + [abch][defj][gikl] \\ &- [abdh][cefj][gikl]. \end{aligned} \tag{4}$$

Each term (monomial) in Eq. (4) is a multiplication of three brackets.

The singularity condition arises when the right-hand side of (4) is equal to zero. This is the basic equation used in the singularity analysis in this paper. It is to be noted that we use Eq. (4) instead of another shorter version having 16 monomials (Downing et al., 2002). The main advantage of Eq. (4) over the shorter version is the order of the points in each bracket, which is lexicographically in both rows and columns. This fact significantly facilitates the manipulation and comparison of monomials, operations needed for the derivation of the geometric condition of the singularity equation.

For the remaining 34 GSP combinations, the singularity equation is significantly reduced since many monomials vanish due to the appearance of equal points in some brackets. For most of these structures, this equation enables the geometrical explanation of the singularity condition using GCA tools. The reduction of the original equation, however, may be differently obtained if different order of legs is taken to substitute the left-hand side of Eq. (4). For example, for the 3-3 GSP, the following leg definition leads to two and four monomials if the leg order is altered:

$$[[ab, af, cb, cd, ed, ef]] = [abfc][acde][bdef] + [abfd][acbe][cdef], \tag{5}$$

$$[[ab, cd, af, cb, ed, ef]] = -[abcd][afce][bdef] + [abcd][afbe][cdef]$$

$$+ [abcf][dcbe][adef] - [abdc][cafe][bdef]. \tag{6}$$

Although Eqs. (5) and (6) are equivalent, the lowest number of monomials is recommended in order to avoid long calculations. Therefore, if the number of monomials obtained after the user definition is more than 4, then he is led to use the automatic function to find the shortest form. This function runs all the possible orders and returns the first shortest form. The next step is to find the interchangeable points within the monomials in order to identify the geometrical entities involved in the singularity condition.

2.3 Identification of Interchangeable Points

The objective of this stage is to automatically find the geometric entities involved in the singularity condition. These entities may be lines, planes or tetrahedrons. The method to find them is based on the first stage of the Cayley factorization performed by White (1991).

White's algorithm deals only with multilinear expressions, which are those containing each point in each monomial only once (Eq. (4) without any substitution is a multilinear example). According to his algorithm a pair of points is interchangeable if the expression after replacing all the appearances of both with each other, summed with the original expression is equal to zero:

$$P(a, b, \ldots) + P(b, a, \ldots) = 0, \tag{7}$$

where P is the expression containing all the monomials (for example, Eq. (4)). This process is performed for all possible pairs of points, using the straightening algorithm (White, 1991).

Unfortunately, our expressions are never multilinear. Unlike the general GSP, of which singularity has no special geometrical explanation with this method, all other structures have at least one point appearing at least twice in each monomial. Until now, no algorithm of Cayley factorization for non-multilinear polynomials is known and it still remains an open problem. Our approach is as follows.

First, we assume that if a point appears more than once in each monomial, then each appearance belongs to a different geometric entity. Each monomial has three brackets, each bracket containing four points, thus 12 points are part of geometric entities that have to be identified. From the definition of the *meet* operation (Eq. (2)), to obtain a monomial of brackets of four points the geometric entities involved may be 2- or 3-extensors (lines or planes). Otherwise, a *meet* including a 4-extensor (tetrahedron) and another entity would lead to a 5-bracket. Still, a monomial of 4-brackets may result from a bracket containing a tetrahedron and two other brackets resulting from a meet of lines and planes. Accordingly, when the potential entities to be searched are lines, planes and tetrahedrons, the following groups can be found: (a) six lines; (b) two planes and three lines; (c) four planes; (d) one tetrahedron, two

planes and one line; (e) one tetrahedron and four lines; (f) two tetrahedrons and two lines; and (g) three tetrahedrons.

To avoid the non-multilinearity problem, the order of searching is as follows:

1. The first entities to be searched are the tetrahedrons. These are searched as common brackets (having the four points of the tetrahedron) in all the monomials. If the equation has more than one monomial, then it is searched if there is a common bracket appearing in all the monomials. If such a bracket exists, then the tetrahedron recognition is done, and the remaining equation continues the search procedure. To have three tetrahedrons there has to be only one monomial in the equation, where each bracket consists of the points of each tetrahedron. If this is the case, then the procedure is completed, resulting in three possible coplanar tetrahedrons, according to the points appearing in each bracket in the monomial.

2. The second stage is to look for planes, which are represented as triplets of interchangeable points. These triplets are searched as points that appear together in one bracket in every monomial. Any pair within such a triplet is interchangeable since replacing them one by another means a permutation, leading to a general sign change in all the monomials, and thus satisfying Eq. (7).

3. Once all the triplets in the previous stage were recognized, their labels acquire a star ($a \rightarrow a^*$) to distinguish them from the other same labels appearing in other brackets in the monomials, due to the non-multilinearity. The next stage consists in searching among the remaining points, pairs that satisfy Eq.(7). In many cases it is not necessary to use the straightening algorithm to verify this condition. For the cases in which it would be necessary, unfortunately we cannot use this algorithm because of the non-multilinearity of the expressions. Therefore in these cases the points that remain without being identified to any entity are left in parentheses and are treated as follows. If three planes were already identified then the residual letters will be treated as possible part of a fourth plane (case c). The same occurs with the residual letter if three planes and one line were found. If two planes and a line were identified, then the residual letters will be referred as possible pairs of lines to correspond to case b.

2.4 Singularity Solution and Visualization

This stage provides the singularity condition as a geometrical incidence between the entities that were already identified. The *union* and *intersection* of geometric entities in terms of GCA are obtained by means of the *join* and *meet* operators, respectively.

Ben-Horin and Shoham (2006a, 2006b, 2007) found the geometrical conditions for 31 from the 34 regular GSPs. According to the number and types of entities identified in the previous section, this stage verifies if, algebraically, the respective condition is equal to a geometric incidence, some examples of which are shown in Section 1. If both are equal, then the solution is written according to the points that were defined before, and the geometric entities involved are shown in the robot figure.

Once the singularity condition is obtained, a new field appears in the window with a value standing for the singularity condition in the actual configuration. When this value, which changes depending on the robot pose, becomes smaller than a predefined ε, it means closeness to singularity and a warning message appears. In the robot figure the user is able to move the platform in 6 DOFs in order to visualize the poses that satisfy the singularity condition. The singularity test is performed using the simplest form according to the condition obtained. For example, the four intersecting planes condition is verified by calculating the condition number of the 4×4 matrix containing the coefficients of the planes. The condition for two planes to intersect simultaneously a line is verified by calculating the condition number of the matrix having the four homogeneous coordinates of four points: two of them lie on the line of intersection of the planes, and the remaining two lie on the other line.

For the interested users, the identification of the interchangeable points and the verification of the singularity condition can be performed manually. By default, the automatic mode is applied.

2.5 Applications

The structures suitable to be analyzed by this method and used with this tool include a long list of robots in addition to all the GSP structures. Their analyses are performed knowing the equivalent lines of action applied to the platform by the legs. For 6-DOF robots these equivalent lines of action are the reciprocal screws to the passive joints of legs (McCarthy, 2000). The topology of the lines of action must be equal to one of the 34 combinations of legs arrangement presented in Faugere and Lazard (1995) since they are all the different combinations existing to define six legs connecting two platforms when two and three concurrent joints are taken into consideration.

Manipulators with lower DOF having a spherical joint in each leg can be also analyzed by this tool. For their analysis the 6×6 Jacobian matrix, which contains a Jacobian of actuations and a Jacobian of constraints as sub-matrices, is needed (Joshi and Tsai, 2002). Once the rows of both sub-matrices are identified, these rows actually being the wrench screws applied to the platform, the topologically equivalent GSP can be identified and the singularity condition can be found by means of SinguLab.

In a future version of SinguLab, an automatic specification of other parallel robots than GSPs with graphical tools will be available. The use of SinguLab is performed according to the interactive instructions in the main window. These instructions follow the steps listed in Section 2. This guidance, showing each result algebraically, graphically and as a geometric statement, enables non-experts to obtain the singularity in an easy manner.

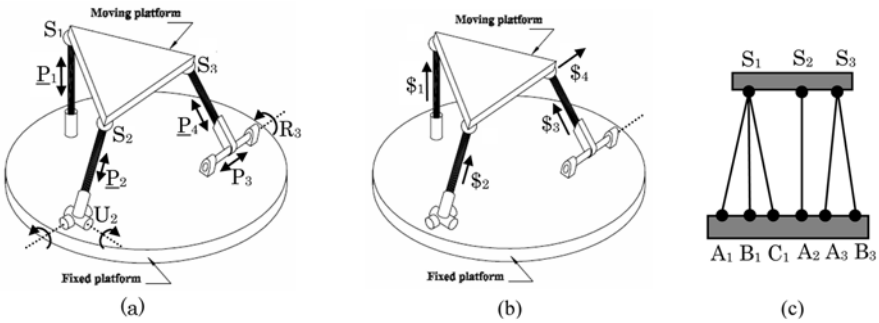


Fig. 1 (a) Four-DOF robot from Gallardo-Alvarado et al. (2006), (b) reciprocal screws, (c) equivalent structure.

3 Example

We exemplify the tool by means of the singularity analysis of a 4-DOF robot, first presented in Gallardo-Alvarado et al. (2006). This robot consists of three different legs, having PS, UPS and PRPS (or CPS) kinematic chains, respectively (the underlined labels stand for actuated joints), see Figure 1a.

The constraint screws of leg 1 are reciprocal to the spherical and prismatic joints. Thus they form a two-system perpendicular to the prismatic and passing through the spherical joint. Then the reciprocal screw to the passive joint of leg 1, $\$_1$, is a screw directed along the leg. In leg 2 the passive joints are the spherical and the universal joints thus their reciprocal screw, $\$_2$, is directed along the leg that connects them. The third leg has two prismatic actuators P_3 and P_4. The reciprocal screws to the passive joints form a two-system of zero pitch, being a planar pencil with center at the spherical joint and containing the axis of the revolute joint. Particularly, $\$_3$ and $\$_4$ pass through the spherical joint center and are directed along the leg and parallel to the revolute joint axis, respectively. With these screws known, the equivalent structure to be entered into SinguLab is as appears in Figure 1c. The interface analyzing this robot is shown in Figure 2, where the singularity condition is that at least one of the tetrahedrons composed by $S_1A_1B_1C_1$, $S_1S_3A_3B_3$ and $S_1S_3S_2A_2$ (according to the labels in Figure 1c) is coplanar.

4 Conclusion

This paper presents SinguLab, a graphical user interface for the singularity analysis of parallel robots. The theoretical background behind this analysis is based on Grassmann–Cayley algebra, which provides a coordinate-free approach for treating geometric entities and their incidences. The identification of the geometric entities and the singularity conditions are made automatically. The results are based on pre-

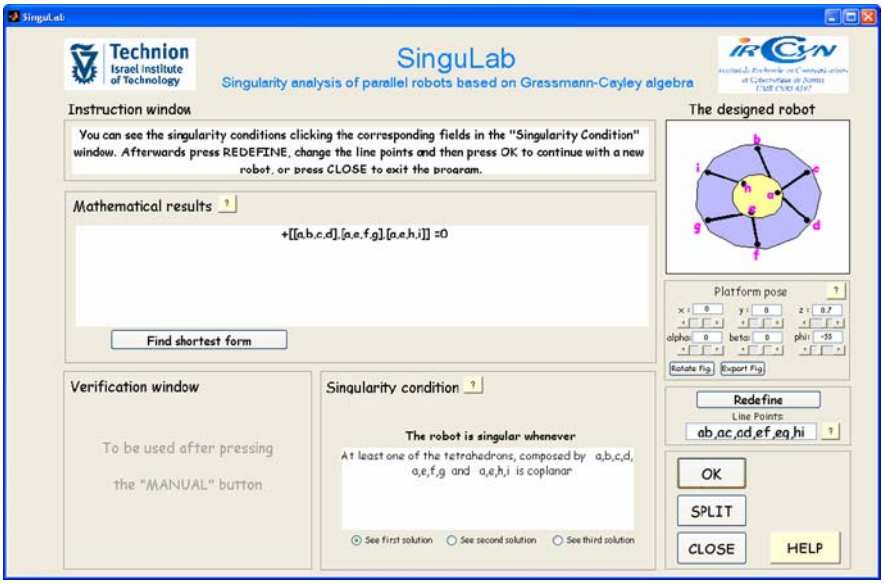


Fig. 2 SinguLab interface.

vious studies on certain classes of Gough–Stewart platforms. This interface is suitable for a broad range of parallel robots. The current version can be used only for robots actuated with SPS chains. For those with serial chains, an equivalent structure has to be predefined with their reciprocal screws. Accordingly, the topological arrangement of the lines of action is the robot definition input. For lower-DOF parallel robots, the reciprocal screws standing for the actuation and for the constraints on the platform have to be identified first. Therefore, we come up with the 6×6 Jacobian matrix and the same GCA approach thereafter. In future versions, the analysis of these robots will be incorporated in the software to provide a fully automatic tool for robot designers.

Acknowledgment

This work is supported in part by the French Research Agency A.N.R. (Agence Nationale pour la Recherche).

References

Ben-Horin P. and Shoham M. (2006a), Singularity analysis of a class of parallel robots based on Grassmann–Cayley algebra, *Mechanism and Machine Theory* **41**, 958–970.

Ben-Horin P. and Shoham M. (2006b), Singularity of a class of Gough–Stewart platforms with three concurrent joints, in *Advances in Robot Kinematics: Mechanisms and Motion*, J. Lenarčič and B. Roth (Eds.), Springer, Dordrecht, pp. 265–274.

Ben-Horin P. and Shoham M. (2007), Application of Grassmann–Cayley algebra to geometrical interpretation f parallel robot singularities, *The International Journal of Robotics Research*, submitted.

Downing D.M., Samuel A.E. and Hunt K.H. (2002), Identification of the special configurations of the octahedral manipulator using the pure condition, *The International Journal of Robotics Research* **21**(2), 147–159.

Gallardo-Alvarado J., Rico-Martinez J.M. and Alici G. (2006), Kinematics and singularity analyses of a 4-dof parallel manipulator using screw theory, *Mechanism and Machine Theory* **41**, 1048–1061.

Faugere J.C. and Lazard D. (1995), Combinatorial classes of parallel manipulators, *Mechanism and Machine Theory* **30**, 765–776.

Joshi, S. and Tsai, L.W. (2002), Jacobian analysis of limited-DOF parallel manipulators, *Journal of Mechanical Design* **124**, 254–258.

McCarthy J.M. (2000), *Geometric Design of Linkages*, Springer, New York.

McMillan T. and White N. (1991), The dotted straightening algorithm, *Journal of Symbolic Computation* **11**, 471–482.

White N. (1983), The bracket of 2-extensors, *Congressus Numerantium* **40**, 419–428.

White N. (1991), Multilinear Cayley factorization, *Journal of Symbolic Computation* **11**, 421–438.

Singularity Analysis of Limited-DOF Parallel Manipulators Using Grassmann–Cayley Algebra

Daniel Kanaan, Philippe Wenger and Damien Chablat

Institut de Recherche en Communication et Cybernétique de Nantes, 1 rue de la Noë, 44321 Nantes, France, e-mail: {daniel.kanaan, philippe.wenger, damien.chablat}@irccyn.ec-nantes.fr

Abstract. This paper characterizes geometrically the singularities of limited-DOF parallel manipulators. The geometric conditions associated with the dependency of six Plücker vector of lines (finite and infinite) constituting the rows of the inverse Jacobian matrix are formulated using Grassmann–Cayley algebra. Manipulators under consideration do not need to have a passive spherical joint somewhere in each leg. This study is illustrated with three example robots.

Key words: parallel manipulator, singularity, Grassmann–Cayley algebra, screw theory.

1 Introduction

Parallel singularities are critical configurations in which a parallel manipulator loses its stiffness and gains one and more degrees of freedom (DOF). These singular configurations can be found using analytical, numerical or geometrical methods (Merlet, 2005). The inverse Jacobian matrix of a 6-DOF parallel manipulator has Plücker coordinate vectors of finite lines as its rows. These lines are wrenches of actuation describing the instantaneous forces of actuation applied by the actuators on the moving platform. For parallel manipulators of type Gough–Stewart, parallel singularities occur when lines within legs of the manipulator are linearly dependent, which Merlet (1989) analyzed using Grassmann line geometry. Hao and McCarthy (1998) used screw theory to define conditions for line-based singularities. They focused on 6-DOF parallel manipulators, each leg having at least one actuator and their last three joints equivalent to a passive spherical joint. These conditions ensure that each supporting leg of the system can apply only a pure force to the platform so that it is possible to characterize all singular configurations in terms of the geometry of linearly dependent sets of lines. Ben-Horin and Shoham (2005, 2006) analyzed parallel singularities of two classes of 6-DOF parallel manipulators using Grassmann–Cayley algebra (GCA). They used the superbracket and the Grassmann–Cayley operators to obtain geometric conditions for singularity, namely, when four planes defined by the direction of the joint axes and the location of the spherical joint are concurrent in a point.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 59–68.

Contrary to 6-DOF manipulators, each leg of a limited-DOF parallel manipulator has connectivity less than 6 and, in turn, constraints partly the motion of the moving platform. When a leg loses its ability to constrain the moving platform, a so-called constraint singularity occurs. Joshi and Tsai (2002) developed a methodology for the inverse Jacobian of limited-DOF parallel manipulators with the theory of reciprocal screws. A 6×6-inverse Jacobian matrix was defined, the rows of which are wrenches that provide information about both architecture[1] and constraint singularities. These wrenches, also known as governing lines, are actuation and constraint wrenches applied to the moving platform.

In this paper, we focus on limited-DOF parallel manipulators, i.e. that have less than 6-DOF and we are interested in identifying their parallel singularities using GCA. The application of GCA is enlarged to include limited-DOF parallel manipulators, in which the inverse Jacobian is a 6×6 matrix having not only finite lines (zero pitch screws) as its rows but also infinite lines (infinite pitch screws). In this work, manipulators under consideration do not need to have a passive spherical joint somewhere in each of their legs. The results are illustrated with three example robots.

2 Grassmann–Cayley Algebra

Originally developed by H. Grassmann as a calculus for linear varieties, GCA has two operators, namely the join, denoted by $\vee$ and the meet, denoted by $\wedge$. These two operators are associated with union and intersection between vector subspaces of extensors. These extensors represent geometric entities such as points, lines, planes, etc. in the projective space. GCA makes it possible to work at the symbolic level, where points and lines are represented in a coordinate-free form by their homogeneous and Plücker coordinates, respectively.

GCA is used in this study to transform the singular geometric conditions defined as the dependency between six lines in the 3-dimensional projective space $\mathbb{P}^3$, into coordinate free algebraic expressions involving 12 points selected on the axes of these lines. For further details on GCA concept and its application to robotics, see for example (White, 1994, 2005; Staffetti and Thomas, 2000; Ben-Horin and Shoham, 2006) and references therein.

2.1 Projective Space

The 3-dimensional projective space $\mathbb{P}^3$ can be considered as the union of $\mathbb{R}^3$ with a set of ideal points that are the intersections of parallel lines and planes. The set of all such points forms a plane known as the plane at infinity, Π_∞. This plane may

[1] Following Joshi and Tsai's (2002) terminology, an architecture singularity occurs when the inverse Jacobian matrix is rank-deficient and the Jacobian of constraints is full-rank.

be thought of as the set of all directions, since all lines with the same particular direction intersect Π_∞ in the same unique point and all parallel planes intersect Π_∞ in the same unique line. Hence every line at infinity meets every other line at infinity, and they are therefore all in one plane.

2.2 Bracket Ring

Consider a finite set of points $\{e_1, e_2, \ldots, e_d\}$ defined in the d-dimensional vector space, V, over the field $\mathbb{R}^3$. Each point e_i is represented by a d-tuple by using homogeneous coordinates, where $e_i = e_{1,i}, e_{2,i}, \ldots, e_{d,i}$ $(1 \leq i \leq d)$. The bracket of these points is defined as the determinant of the matrix $\mathbf{M}$ whose columns are the homogeneous coordinates of these points e_i $(1 \leq i \leq d)$:

$$[e_1, e_2, \ldots, e_d] = \det \mathbf{M} = \begin{vmatrix} e_{1,1} & e_{1,2} & \ldots & e_{1,d} \\ \vdots & \vdots & \ldots & \vdots \\ e_{d,1} & e_{d,2} & \ldots & e_{d,d} \end{vmatrix}. \tag{1}$$

The brackets form a subalgebra of the GCA, called the bracket ring or bracket algebra. The brackets satisfy the following relations:

(1) $[e_1, e_2, \ldots, e_d] = 0$ if any $e_i = e_j$ with $i \neq j$, or $e_1, e_2, \ldots, e_d$ are dependent.
(2) $[e_1, e_2, \ldots, e_d] = \text{sign}(\sigma)[e_{\sigma 1}, e_{\sigma 2}, \ldots, e_{\sigma d}]$ for any permutation σ of $\{1, 2, \ldots, d\}$.
(3) $[e_1, e_2, \ldots, e_d][f_1, f_2, \ldots, f_d] = \sum_{j=1}^{d} [f_j, e_2, \ldots, e_d][f_1, f_2, \ldots, f_{j-1}, e_1, f_{j+1}, \ldots, f_f]$.

All relations among the brackets are consequences of relations of the above three types. The relations of the third type are called Grassmann–Plücker relations or syzygies, and they correspond to generalized Laplace expansions by minors (White, 1975).

2.3 The Superbracket Decomposition

The inverse Jacobian matrix of a parallel manipulator has the Plücker coordinates of six lines in the projective space $\mathbb{P}^3$ as its rows. The superjoin of these six vectors in $\mathbb{P}^5$ corresponds to the determinant of their six Plücker coordinate vectors up to scalar multiple, which is the superbracket in GCA $\Lambda(V^{(2)})$ (White, 1983). Thus, a singularity occurs when these six Plücker coordinate vectors are dependent, which is equivalent to a superbracket equal to zero.

White (1983) and McMillan (1990) used the theory of projective invariants to decompose the superbracket into expression having brackets involving 12 points selected on the axes of these lines. Let $[ab, cd, ef, gh, ij, kl]$ be the superbracket

having six 2-extensors as its column. These extensors represent lines ab, cd, ef, gh, ij, kl in the projective space, respectively. The expression of this superbracket taken from Ben-Horin and Shoham (2006) is as follows:

$$\begin{aligned}[ab\ cd\ ef\ gh\ ij\ kl] = {} & [a\ b\ c\ d][e\ f\ \overset{1}{g}\ \overset{2}{i}][\overset{1}{h}\ \overset{2}{j}\ k\ l] \\ & - [a\ b\ \overset{3}{c}\ \overset{4}{e}][\overset{3}{d}\ \overset{4}{f}\ g\ h][i\ j\ k\ l] \\ & - [a\ b\ \overset{5}{c}\ \overset{6}{e}][\overset{5}{d}\ g\ h\ \overset{7}{i}][\overset{6}{f}\ \overset{7}{j}\ k\ l] \\ & + [a\ b\ \overset{8}{c}\ \overset{9}{g}][\overset{8}{d}\ e\ f\ \overset{10}{i}][\overset{9}{h}\ \overset{10}{j}\ k\ l], \end{aligned} \tag{2}$$

where

$$[a\ b\ c\ d][e\ f\ \overset{1}{g}\ \overset{2}{i}][\overset{1}{h}\ \overset{2}{j}\ k\ l]$$

denotes

$$\sum_{1,2} \text{sign}(1,2)\ [a\ b\ c\ d][e\ f\ \overset{1}{g}\ \overset{2}{i}][\overset{1}{h}\ \overset{2}{j}\ k\ l]$$

1, 2 are permutations of the 2-element sets $\{g, h\}$, $\{i, j\}$, respectively.

Equation (2) may be transformed into a linear combination (sum) of 24 bracket monomials, where each bracket monomial is a product of 3 brackets. The monomials in Eq. (2) may be found in McMillan (1990) and Ben-Horin and Shoham (2006).

3 Singularity Geometric Conditions

The aim of this study is to enlarge the application of line geometry to include limited-DOF manipulators whose legs apply actuation and constraint wrenches to the moving platform. These manipulators do not need to have a passive spherical joint anywhere along their legs and their 6×6-inverse Jacobian matrix may have Plücker coordinate vectors of finite and infinite lines as its rows. In this section, we use GCA and the superbracket decomposition to determine the singularity geometric conditions of two classes of parallel manipulators having inverse Jacobian matrices equivalent to those obtained for 3-UPU translational and the parallel module of the Verne machine shown in Figures 1 and 2.

3.1 Singularity Analysis of the 3-UPU Translational Manipulator

The 3-UPU manipulator was studied in Di Gregorio and Parenti-Castelli (1998), Joshi and Tsai (2002), Wolf and Shoham (2003), and Merlet (2005). The moving platform controlled by three linear actuators along the three legs can perform a translational motion when the axes of the base universal joints are parallel to those of the platform universal joints of the same leg. Thus each leg i having connectivity

equal to 5 applies one actuation force, $\hat{\mathbf{F}}_i = [\mathbf{s}_i^T \quad (\mathbf{r}_i \times \mathbf{s}_i)^T]^T$, and one constraint moment, $\hat{\mathbf{M}}_i = [\mathbf{0}_{1\times 3} \quad \mathbf{n}_i^T]^T$, to the moving platform, where $\mathbf{s}_i$ is a unit vector in the direction of the line of application of the force $\hat{\mathbf{F}}_i$ (leg i), $\mathbf{r}_i$ is the position vector of a point on this line and $\mathbf{n}_i$ is the cross product of the two vectors associated with the axes of the base U joint of leg i represented by that the direction of the torque associated with the constraint moment $\hat{\mathbf{M}}_i$. Each actuation force, $\hat{\mathbf{F}}_i$, is a zero pitch screw reciprocal to all joint screws (a joint screw stands for a twist screw associated with the joint) of leg i except for the joint screw associated with the actuated prismatic joint of the same leg. Each constraint moment, $\hat{\mathbf{M}}_i$, is an infinite pitch screw reciprocal to all joint screws of leg i. These actuation forces and constraint moments have Plücker coordinate vectors of finite and infinite lines, respectively, in the 3-dimensional projective space. As a result, the 6×6-inverse Jacobian matrix will have the Plücker coordinate vectors of 3-finite lines (actuation forces) and 3-infinite lines (constraint moments) as its rows. The dependency between these lines is related to the degeneration of the inverse Jacobian matrix, which is equivalent to a superbracket equal to zero. The transpose of the inverse Jacobian of the 3-UPU can be expressed as follows (Joshi and Tsai, 2002):

$$\mathbf{J}^{-T} = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \mathbf{s}_3 & \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} \\ \mathbf{r}_1 \times \mathbf{s}_1 & \mathbf{r}_2 \times \mathbf{s}_2 & \mathbf{r}_3 \times \mathbf{s}_3 & \mathbf{n}_1 & \mathbf{n}_2 & \mathbf{n}_3 \end{bmatrix}$$
$$= \left[\hat{\mathbf{F}}_1 \;\; \hat{\mathbf{F}}_2 \;\; \hat{\mathbf{F}}_3 \;\; \hat{\mathbf{M}}_1 \;\; \hat{\mathbf{M}}_2 \;\; \hat{\mathbf{M}}_3\right]. \tag{3}$$

Let ab, cd, ef, be the finite lines representing the 3-actuation forces $\mathbf{F}_i$ ($i = 1, 2, 3$), where a, b, c are finite point and b, d, f are points at infinity. All these points are expressed with their homogeneous coordinates, where $b = (\mathbf{s}_1^T \quad 0)^T$, $d = (\mathbf{s}_2^T \quad 0)^T$, $f = (\mathbf{s}_3^T \quad 0)^T$. On the other hand, since every line at infinity meets every other line at infinity, the three constraint moments $\mathbf{M}_i$ ($i = 1, \ldots, 3$) can be represented by three infinite lines gh, gi and hi, respectively. According to Eq. (2) and due to the repetition of points in the same bracket, we simplify the superbracket expression into a reduced number of non-zero monomial terms and the superbracket decomposition of our manipulator, in turn, reduces to:

$$[ab, cd, ef, gh, gi, hi] = [abdf][cghi][eghi], \tag{4}$$

where $[abdf] = (\mathbf{s}_1 \times \mathbf{s}_2) \cdot \mathbf{s}_3$ and $[eghi] = [cghi] = (\mathbf{n}_1 \times \mathbf{n}_2) \cdot \mathbf{n}_3$.

Thus, the manipulator under study is at a singularity whenever vectors $\mathbf{s}_i$ or $\mathbf{n}_i$ ($i = 1, \ldots, 3$) verify Eq. (4), which occurs respectively when triangle ghi or bdf vanishes (its surface is equal to zero). These two conditions include the cases (c–h) in Figure 1. Each vertex of triangle bdf is a point at infinity representing the direction of an actuation force. In case (c, d), the 3 points at infinity b, d and f are coincident, so three vectors $\mathbf{s}_i$ are coplanar. Case (d, e) occurs when at least two of the three points b, d, f coincide and so at least two of the three forces are parallel. Each side of the triangle ghi is a line at infinity defined as the intersection of a family of parallel planes at infinity, with $\mathbf{n}_i$ as the normal to these planes. Case (f, g) occurs

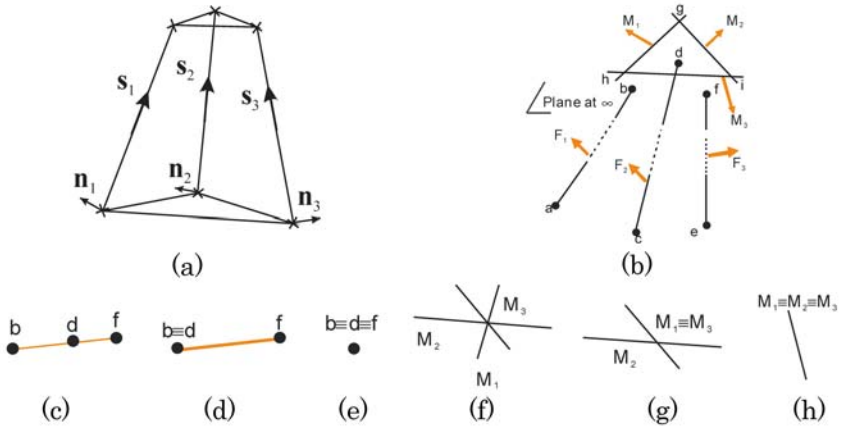


Fig. 1 Geometric conditions for parallel singularities.

when the three lines at infinity intersect at the same point, which means that three vectors $\mathbf{n}_i$ are coplanar. In case (g, h), at least two lines at infinity are collinear, which means that their corresponding planes are parallel and so at least two of the vectors $\mathbf{n}_i$ are parallel.

These conditions stand for all manipulators with an inverse Jacobian matrix consisting of three actuation forces and three constraint moments, like for example the 3-UPU translational manipulator and the linear Delta robot (Clavel, 1988). Notice that another condition may appear when at least one constraint moment degenerates, meaning that at least one line gh, gi or hi degenerates to a point. This condition is more related to the arrangement of the joint within each leg. For the 3-UPU manipulator, this case can only happen if a universal joint is replaced by two revolute joints with parallel axes.

3.2 Singularity Analysis of the Verne parallel Module

The parallel module of the Verne machine consists of three legs, leg I, II and III (Figure 2a). Each leg uses pairs of rods linking a prismatic joint to the moving platform through two pairs of spherical joints. Legs II and III are two identical parallelograms. Leg I differs from the other two legs in that $ac \neq bd$. Leg I does not remain planar (rod directions define skew lines) as the machine moves, unlike what arises in the other two legs that are articulated parallelograms. The movement of the moving platform is generated by the slide of three actuators along three vertical guideways. We suppose that we are out of the serial singularities and we are interested only on studying the parallel singularities.We can thus consider that the transpose of the inverse Jacobian of this manipulator can be expressed as follows:

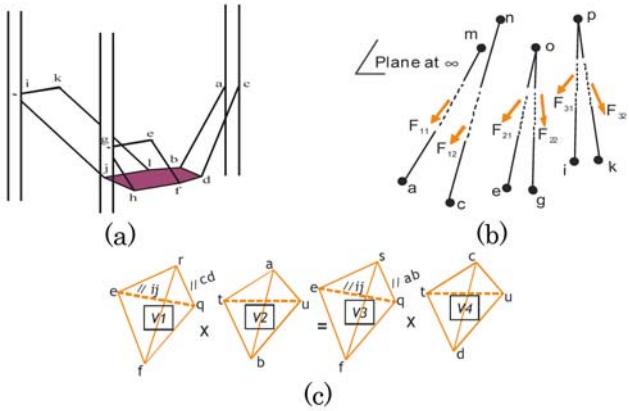


Fig. 2 Singularity geometrical conditions of the Verne parallel module.

$$\mathbf{J}^{-T} = \left[\hat{\mathbf{F}}_{11}\ \ \hat{\mathbf{F}}_{12}\ \ \hat{\mathbf{F}}_{21}\ \ \hat{\mathbf{F}}_{22}\ \ \hat{\mathbf{F}}_{31}\ \ \hat{\mathbf{F}}_{32}\right]. \tag{5}$$

The rows of $\mathbf{J}^{-1}$ are zero pitch screws, $\hat{\mathbf{F}}_{ij} = [\mathbf{s}_{ij}^T\ \ (\mathbf{r}_{ij} \times \mathbf{s}_{ij})^T]^T$ $(i = 1, \ldots, 3,$ $j = 1, 2)$, represented by finite lines along the six rods, where $\mathbf{s}_{ij}$ is a unit vector in the direction of the rod j within the leg i and $\mathbf{r}_{ij}$ is the position vector of a point on this rod. Each of those screws is an actuation force, which is reciprocal to all the joint screws of the rod j within the leg i except for the joint screw associated with the actuated prismatic joint of the same leg. Notice that the pair of actuation forces within legs II and III are parallel, so $\mathbf{s}_{21} = \mathbf{s}_{22}$ and $\mathbf{s}_{31} = \mathbf{s}_{32}$. The singularity geometrical conditions are associated with the dependency between these six actuation forces supported by the rods of the Verne parallel module.

Let ab, cd, ef, gh, ij, kl, be respectively the finite lines representing the six actuation forces, where points $a, b, \ldots, l$ are finite points located at the center of spherical joints (Figure 2). We suppose that lines ef, gh are parallel and intersect at infinity as well as for lines ij, kl. Thus, the superbracket decomposition of these lines reduces to:

$$[ab, cd, ef, gh, ij, kl] = [am, cn, eo, go, ip, kp], \tag{6}$$

where $m = b - a$, $n = d - c$, $ef \wedge gh = o = f - e = h - g$ and $ij \wedge kl = p = j - i = l - k$.

The shortest form of the superbracket decomposition using the algorithm in Ben-Horin et al. (2008) will result in the following non-zero monomial terms:

$$\begin{aligned}&[oegm][oncp][aipk] - [oega][oncp][mipk]\\ &\quad - [oegn][omap][cipk] + [oegc][omap][nipk].\end{aligned} \tag{7}$$

After collecting equal brackets and applying the shuffles relation (see White, 2005) caused by the permutation of m, a and n, c, we obtain:

$$[o\ n\ c\ p][o\ e\ g\ \dot{m}][\dot{a}\ i\ p\ k] - [o\ m\ a\ p][o\ e\ g\ \dot{n}][\dot{c}\ i\ p\ k] = 0. \tag{8}$$

Applying the following relation, $[(b-a)cde] = [bcde] - [acde]$, we are able to replace some points at infinity by finite points and obtain the following expression:

$$\begin{aligned}[ab\ cd\ ef\ gh\ ij\ kl] = {} & [o\ n\ c\ p][f\ e\ g\ \dot{b}][\dot{a}\ i\ j\ k] \\ & - [o\ m\ a\ p][f\ e\ g\ \dot{d}][\dot{c}\ i\ j\ k].\end{aligned} \tag{9}$$

Using the meet operator relations (see White, 2005), we prove that:

$$\begin{aligned}[f\ e\ g\ \dot{b}][\dot{a}\ i\ j\ k] &= (ijk \wedge feg) \wedge ba = tu \wedge ba = [tuba] \quad \text{and} \\ [f\ e\ g\ \dot{d}][\dot{c}\ i\ j\ k] &= (ijk \wedge feg) \wedge dc = tu \wedge dc = [tudc]\end{aligned} \tag{10}$$

where tu is the line of intersection of leg II and III planes including such as: $\mathbf{tu} = (\mathbf{ef} \times \mathbf{eg}) \times (\mathbf{ij} \times \mathbf{ik})$.

On the other hand, it is easy to prove that:

$$[oncp] = [fepn] = [feqr] \quad \text{and} \quad [omap] = [feqs], \tag{11}$$

where $q = e + p$, $r = q + n$ and $s = q + m$.

Finaly, the invariant algebraic expression related to the existence of parallel singularities of the parallel module of the Verne machine can be stated as:

$$[feqr][tuba] - [feqs][tudc] = 0, \tag{12}$$

where $[tuba] = \mathbf{tu} \cdot (\mathbf{ub} \times \mathbf{ab})$, $[tudc] = \mathbf{tu} \cdot (\mathbf{ud} \times \mathbf{cd})$, $[feqr] = \mathbf{cd} \cdot \mathbf{N}$ and $[feqs] = \mathbf{ab} \cdot \mathbf{N}$ with $\mathbf{N} = \mathbf{ef} \times \mathbf{ij}$.

The above expression is geometrically equivalent to the difference between the volume products of two pairs of tetrahedrons with vertices expressed as function of points $a, b, \ldots, l$ as shown in Figure 2c. This geometric condition includes the following cases: (i) planes of legs II and III are coplanar or parallel ($\|\mathbf{tu}\| = \mathbf{0}$); (ii) ef and ij are parallel ($\|\mathbf{N}\| = 0$); (iii) ef, cd are parallel and ij, ab are parallel or ef, ab are parallel and ij, cd are parallel; (iv) ab and cd intersect with tu, in this case the six actuation forces form a singular linear complex; (v) ef, cd are parallel and tu, cd are coplanar or ij, cd are parallel and tu, cd are coplanar or ij, ab are parallel and tu, ab are coplanar or ef, ab are parallel and tu, ab are coplanar; (vi) the 6 actuation forces form a general linear complex expressed by

$$[\mathbf{cd} \cdot \mathbf{N}][\mathbf{tu} \cdot (\mathbf{ub} \times \mathbf{ab})] - [\mathbf{ab} \cdot \mathbf{N}][\mathbf{tu} \cdot (\mathbf{ud} \times \mathbf{cd})]. \tag{13}$$

The singularity conditions of the Delta robot are obtained for the particular when $\mathbf{cd} = \mathbf{ab}$, thus Eq. (13) is reduced to:

$$[\mathbf{ab} \cdot \mathbf{N}][\mathbf{tu} \cdot (\mathbf{db} \times \mathbf{ab})] = [(\mathbf{s}_1 \times \mathbf{s}_2) \cdot \mathbf{s}_3][(\mathbf{n}_1 \times \mathbf{n}_2) \cdot \mathbf{n}_3]. \tag{14}$$

4 Conclusions

This paper showed how GCA can be used to determine the geometric conditions associated with the singular configurations of limited-DOF parallel robots whose legs can transmit forces and moments to the moving platform. Three example robots were analyzed as illustrative examples. This method provides a physical meaning and a geometrical interpretation of singular configurations for a familly of parallel manipulators, which is of interest for the designer of new robots.

References

1. Ben-Horin, P. and Shoham, M. (2005), Singularity analysis of parallel robots based on Grassmann–Cayley algebra, in *International Workshop on Computational Kinematics*, Cassino, May 4–6.
2. Ben-Horin, P. and Shoham, M. (2006), Singularity condition of six degree-of-freedom three-legged parallel robots based on Grassmann–Cayley algebra, *IEEE Trans. Robotics* **22**(4), 577–590.
3. Ben-Horin, P., Shoham, M., Caro, S., Chablat, D. and Wenger, P. (2008), A graphical user interface for the singularity analysis of parallel robots based on Grassmann–Cayley algebra, in *Advances in Robot Kinematics*, this volume.
4. Clavel, R. (1988), DELTA, A fast robot with parallel geometry, in *Proceedings of 18th International Symposium on Industrial Robots*, Lausanne, pp. 91–100.
5. Di Gregorio, R. and Parenti-Castelli, V. (1998), A translational 3-DOF parallel manipulator, in *Advances in Robot Kinematics*, Lenarčič, J., Husty, M.L. (Eds.), Kluwer Academic Publishers, Dordrecht, pp. 49–58.
6. Joshi, S.A. and Tsai, L.W. (2002), Jacobian analysis of limited-DOF parallel manipulators, *ASME J. Mech. Des.* **124**, 254–258.
7. Hao, F. and McCarthy, J. (1998), Conditions for line-based singularities in spatial platform manipulators, *J. of Robotic Systems* **15**(1), 43–55.
8. Kanaan, D., Wenger, P. and Chablat, D. (2006), Workspace analysis of the parallel module of the VERNE machine, *Problems of Mechanics* **25**(4), 26–42.
9. McMillan, T. (1990), *Invariants of Antisymmetric Tensors*, PhD Dissertation, University of Florida.
10. Merlet, J.P. (1989), Singular configurations of parallel manipulators and Grassmann geometry, *Int. J. Robotics Res.* **8**(5), 45–56.
11. Merlet, J.P. (2006), *Parallel Robots*, Second Edition, Springer, Dordrecht.
12. Pottmann H., Peternell M. and Ravani B., (1999), An introduction to line geometry with applications, *Comput. Aided Design* **31**, 3–16.
13. Staffetti, E. and Thomas, F. (2000), Analysis of rigid body interactions for compliant motion tasks using the Grassmann–Cayley algebra, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robotic System*.
14. White, N. (1975), The bracket ring of a combinatorial geometry I, *Trans. Amer. Math. Soc.* **202**, 79–95.
15. White, N. (1983), The bracket of 2-extensors, *Congressus Numerantium* **40**, 419–428.
16. White, N. (1994), Grassmann–Cayley algebra and robotics, *J. Intelligent and Robotics Systems* **11**, 91–107.
17. White, N. (2005), Grassmann–Cayley algebra and robotics applications, in *Handbook of Geometric Computing*, Part VIII, Eduardo Bayro Corrochano (Ed.), Springer, Berlin/Heidelberg, pp. 629–656.

18. Wolf, A. and Shoham, M. (2003), Investigation of parallel manipulators using linear complex approximation, *ASME J. Mech. Des.* **125**, 564–572.
19. Zhao, T.-S., Dai, J.S. and Huang, Z. (2002), Geometric analysis of overconstrained parallel manipulators with three and four degrees of freedom, *JSME Int. Journal, Ser. C, Mech. Systems, Mach. Elem. Manuf.* **45**(3), 730–740.

DESIGN OF ROBOTS AND MECHANISMS

On the Design of Fully Constrained Parallel Cable-Driven Robots

M. Gouttefarde[1], S. Krut[1], O. Company[1], F. Pierrot[1] and N. Ramdani[1,2]

[1]*LIRMM, CNRS – Université Montpellier 2, France,*
e-mail: {gouttefarde,krut,company,pierrot}@lirmm.fr
[2]*INRIA Sophia-Antipolis, France, e-mail: nacim.ramdani@inria.fr*

Abstract. This work deals with the design of parallel cable-driven robots having more cables than degrees of freedom (DOF). Compared to parallel robots with rigid links, this issue has a distinctive property: the requirement of keeping the cables taut. In this paper, it is shown how numerical tools, recently introduced elsewhere, can be used to solve the important practical problem of finding geometries of robots such that a given prescribed workspace is fully included in the wrench-feasible workspace. Then, in order to improve the solving procedure of this problem, it is shown that another workspace is of interest: the wrench-closure workspace. Some of its relationships with the wrench-feasible workspace are highlighted.

Key words: cable-driven robot, parallel robots, design, workspace.

1 Introduction

Many existing works deal with the limitation of the workspace of parallel cable-driven robots induced by the unilateral nature of the forces applied by the cables on the mobile platform. Most of them propose methods allowing to determine the workspace of these robots, for instance, by means of a discretization method [7] or by delineating its boundary [2, 9]. But very few of them tackle the difficult design problem of finding robots having a prescribed workspace in which the cables remain taut. In Section 2 of this paper, it is shown how numerical methods recently introduced in [3] allow to work in the design framework of the so-called parameter space approach [6] in order to find geometries of fully constrained parallel cable-driven robots for which a prescribed workspace is fully included in the wrench-feasible workspace (WFW). The WFW is appropriate for this type of cable-driven robots since it ensures that the end-effector can exert or balance wrenches in all directions of the task space with tension forces in the cables remaining below given limits [1]. An example presented in Section 3 illustrates the proposed method. Furthermore, in Section 4, this example is used to motivate the interest of considering, together with the WFW, another workspace: the wrench-closure workspace (WCW). In the context of design dealt with in this paper, an important distinction between these two

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 71–78.

workspaces being that the WCW depends only on the geometry of the robot. Two useful relationships between the WFW and the WCW are highlighted and proved leading to a two-step approach to the design problem considered in this work. Finally, Section 5 illustrates briefly this approach and Section 6 concludes the paper.

2 Design Method

The relationship between the tensions in the m cables and the wrench $\mathbf{f}$ applied by these cables on a reference point C of the mobile platform is given by [4, 8]

$$\mathbf{W}\boldsymbol{\tau} = \mathbf{f}, \tag{1}$$

where $\boldsymbol{\tau}$ is the vector whose components are the tensions in the cables and $\mathbf{W}$ is the $n \times m$ wrench matrix, n denoting the number of DOF.

Let the n-dimensional vector $\mathbf{X}$ denotes the pose of the mobile platform – for instance, in the case of a 6-DOF robot, three components of $\mathbf{X}$ give the position of C and the three others are angles defining the orientation of the mobile platform. Moreover, let $\mathbf{P}$ be a vector of parameters defining completely, possibly with some other constant parameters, the geometry of the robot, that is, the positions of the actuated reels at the base and the positions of the cables attachment points on the mobile platform. Then, the wrench matrix $\mathbf{W}$ depends on both $\mathbf{X}$ and $\mathbf{P}$: $\mathbf{W} = \mathbf{W}(\mathbf{X}, \mathbf{P})$.

Generally, the unilateral nature of the forces applied by the cables on the mobile platform limits the workspace of parallel cable-driven robots. One can tackle this problem by means of the WFW which is defined as follows [1].

Definition (WFW) *The WFW is the set of poses* $\mathbf{X}$ *for which, for any wrench* $\mathbf{f}$ *in* $[\mathbf{f}]$*, there exists a vector of cable tensions* $\boldsymbol{\tau}$ *in* $[\boldsymbol{\tau}]$ *such that* $\mathbf{W}\boldsymbol{\tau} = \mathbf{f}$.

In this definition, $[\mathbf{f}]$ is the (bounded) set of wrenches that the cables must apply on the mobile platform, called the *required set of wrenches*, and $[\boldsymbol{\tau}]$ is the *set of allowed cable tensions*

$$[\boldsymbol{\tau}] = \left\{ \boldsymbol{\tau} \mid \tau_i \in [\tau_{i_{\min}}, \tau_{i_{\max}}], \;\; 0 \leq \tau_{i_{\min}} < \tau_{i_{\max}}, \;\; 1 \leq i \leq m \right\}. \tag{2}$$

In [3], we proposed a numerical method based on interval analysis which allows us to determine, for a given robot geometry $\mathbf{P}$, whether or not a given box workspace $[\mathbf{X}]$ is fully included in the WFW. A straightforward extension of this method can handle sets $[\mathbf{P}]$ of robot geometries. The corresponding implementation in C++ provides a procedure denoted `FullyIncluded()` which takes as arguments a set of robot geometries $[\mathbf{P}]$ and the box workspace $[\mathbf{X}]$. When the procedure returns 1, the method ensures that

$$\forall\, \mathbf{P} \in [\mathbf{P}], \forall\, \mathbf{X} \in [\mathbf{X}], \forall\, \mathbf{f} \in [\mathbf{f}], \exists\, \boldsymbol{\tau} \in [\boldsymbol{\tau}] \text{ such that } \mathbf{W}\boldsymbol{\tau} = \mathbf{f}. \tag{3}$$

Input: $\mathcal{B}$, [**X**], [**f**], [$\boldsymbol{\tau}$], ε
Output: $\mathcal{L}_{\text{in}}$, $\mathcal{L}_{\text{out}}$, $\mathcal{L}_{\text{neg}}$
1. $\mathcal{L} \leftarrow \mathcal{B}$ % Initialize list $\mathcal{L}$ with the search box $\mathcal{B}$
2. **while** $\mathcal{L} \neq \emptyset$ **do**
3. [**P**] $\leftarrow$ `Extract`($\mathcal{L}$)
4. included $\leftarrow$ `FullyIncluded`([**P**], [**X**])
5. **if** included=1 **then**
6. $\mathcal{L}_{in} \leftarrow$ [**P**]
7. **else if** included=-1 **then**
8. $\mathcal{L}_{\text{out}} \leftarrow$ [**P**] % Pruning
9. **else** % included=0
10. **if** Size([**P**]) $> \epsilon$ **then**
11. $\mathcal{L} \leftarrow$ `Bisect`([**P**]) % Branching
12. **else** % [**P**] is too small to be bisected
13. $\mathcal{L}_{neg} \leftarrow$ [**P**]
14. **end if**
15. **end if**
16. **end while**

Fig. 1 A branch-and-prune algorithm to find robots having a prescribed workspace fully included in the WFW.

In words: for all the robots geometries **P** in [**P**] the box workspace [**X**] is completely included in the WFW. If it returns -1, the method ensures that

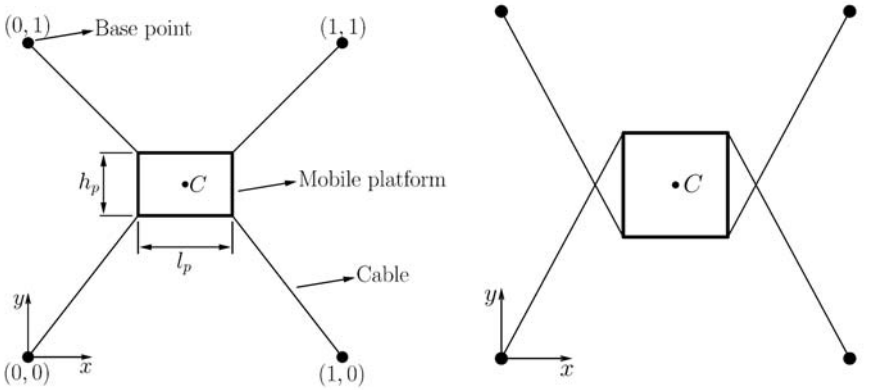
$$\forall\, \mathbf{P} \in [\mathbf{P}], \exists\, \mathbf{X} \in [\mathbf{X}], \exists\, \mathbf{f} \in [\mathbf{f}] \text{ such that } \forall\, \boldsymbol{\tau} \in [\boldsymbol{\tau}],\ \mathbf{W}\boldsymbol{\tau} \neq \mathbf{f}, \tag{4}$$

i.e., for all **P** in [**P**] the box workspace [**X**] is not fully included in the WFW – at least one pose $\mathbf{X} \in [\mathbf{X}]$ lies outside of the WFW. Finally, the procedure returns 0 when it is not able to conclude.

With the tool provided by the procedure `FullyIncluded`(), the design framework introduced in [5] and [6, chapter 11] can be applied to the problem of finding parallel cable-driven robots which have a prescribed workspace [**X**] fully included in the WFW. A basic realization of this design methodology is depicted in Figure 1 where a branch-and-prune algorithm is used in order to explore the space of design parameters **P** – which are geometric parameters in this paper.

3 Illustration

In order to illustrate the algorithm shown in Figure 1, let us consider the 3-DOF planar parallel cable-driven robot shown in Figure 2(a). Its pose is given by $\mathbf{X} = (x, y, \phi)^T$ where x and y are the coordinates of point C and ϕ is the angle defining the orientation of the mobile platform. The design parameters that characterize the geometry of the platform are its length l_p and its height h_p. The actuated reels fixed to the base are located at the vertices of a square of side length 1 m. Hence, in this example, $\mathbf{P} = (l_p, h_p)^T$ defines completely the geometry of the robot since the



(a) 3-DOF planar parallel cable-driven robot

(b) Geometry obtained for $l_p = 0.2$ and $h_p = -0.2$ (m)

Fig. 2 An illustration example.

platform is taken rectangular and the geometry of the base is fixed. Note that we let l_p and h_p be negative in order to be able to consider robots with crossed cables. For instance, the robot obtained for $l_p = 0.2$ m and $h_p = -0.2$ m is shown in Figure 2(b).

The design problem to be solved in this example consists in finding the design parameters $\mathbf{P} = (l_p, h_p)^T$ such that the box workspace

$$[\mathbf{X}] = ([0.3, 0.7], [0.3, 0.7], [-\pi/6, \pi/6])^T \tag{5}$$

is fully included in the WFW defined by $[\mathbf{f}] = ([-10, 10], [-10, 10], [-0.5, 0.5])^T$ and $[\boldsymbol{\tau}] = ([1, 55], [1, 55], [1, 55], [1, 55])^T$. The first two components of the required set of wrenches $[\mathbf{f}]$ correspond to ranges of forces (N) and the last one to a range of moments (N.m) – hence, $[\mathbf{f}]$ has the shape of a box. The four components of the box of allowed cable tensions $[\boldsymbol{\tau}]$ are ranges of forces (N).

To solve this problem, the algorithm shown in Figure 1 manages a list $\mathcal{L}$ of boxes $[\mathbf{P}]$ of design parameters. Depending on the result of the procedure `FullyIncluded()`, the current box $[\mathbf{P}]$ is put in the list $\mathcal{L}_{in}$, in the list $\mathcal{L}_{\text{out}}$ (pruning) or else, if not too small, $[\mathbf{P}]$ is bisected yielding two smaller boxes which are added to $\mathcal{L}$ (branching). Thanks to the properties of the procedure `FullyIncluded()`, briefly described in Section 2, the boxes in $\mathcal{L}_{in}$ provide an approximation of the set of design parameters $\mathbf{P}$ for which $[\mathbf{X}]$ is fully included in the WFW. $\mathcal{L}_{\text{out}}$ provides also an approximation but of design parameters $\mathbf{P}$ for which $[\mathbf{X}]$ is not fully included in the WFW. The accuracy of these two approximations depends on the value of the threshold ϵ. The smaller is ϵ, the better is the accuracy at the cost of a higher computation time.

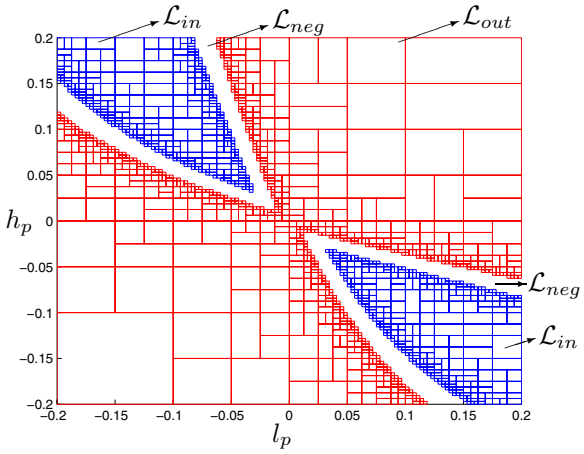


Fig. 3 A design problem solved with the algorithm shown in Figure 1 (for $\epsilon = 0.005$).

For the example introduced above, if the list $\mathcal{L}$ is initialized with the search box of design parameters $\mathcal{B} = [\mathbf{X}]_{\text{ini}} = ([-0.2, 0.2], [-0.2, 0.2])^T$, the result obtained is shown in Figure 3.

4 Relationship between the WFW and the WCW

Note from Figure 3 that only robots with crossed cables, i.e., robots for which l_p is negative or h_p is negative, are solutions to the design problem introduced in Section 3. Now, assume that the designer would prefer not to have crossed cables for practical reasons. Then, a question is whether the set of allowed cable tensions $[\boldsymbol{\tau}]$ can be enlarged so as to find robots with non-crossed cables solving the design problem. Or else, is the set of geometric parameters considered too restrictive? Obviously, in order to try to answer this type of questions, the algorithm of Figure 1 can be applied again with larger sets $[\boldsymbol{\tau}]$. But a more systematic and efficient approach is obtained by considering the WCW.

Definition (WCW) *The WCW is the set of poses* $\mathbf{X}$ *for which, for any wrench* $\mathbf{f}$ *in* $\mathbb{R}^n$*, there exists a vector of cable tensions* $\boldsymbol{\tau} \geq \mathbf{0}$ *in* $\mathbb{R}^m$ *such that* $\mathbf{W}\boldsymbol{\tau} = \mathbf{f}$*, where the notation* $\boldsymbol{\tau} \geq \mathbf{0}$ *means that all the components of* $\boldsymbol{\tau}$ *are nonnegative.*

A well-known characterization of the WCW is [8]: $\mathbf{X}$ belongs to the WCW if and only if the wrench matrix $\mathbf{W}$ has full rank and its null space contains a vector $\mathbf{z} > \mathbf{0}$ ($\mathbf{Wz} = \mathbf{0}$). This characterization shows that *the WCW depends only on the robot geometry* $\mathbf{P}$ since the two conditions that characterize a pose belonging to the WCW are properties of $\mathbf{W}$ only whereas the WFW as defined in section 2 depends on the geometry of the robot but also on $[\mathbf{f}]$ and $[\boldsymbol{\tau}]$.

Moreover, let us prove two properties which highlight the relationship between the WFW and the WCW. Since the focus of this paper is on fully constrained parallel cable-driven robots, henceforth, it is assumed that the robot must exert or balance wrenches in all directions. Then, mathematically speaking, the required set of wrenches $[\mathbf{f}]$ contains a neighborhood of the null wrench $\mathbf{0}$, that is, an open ball of radius $r > 0$ centered at $\mathbf{0}$. The following two properties hold.

Property 1 *The WFW is a subset of the WCW.*

Proof. Consider a pose $\mathbf{X} \in$ WFW and any wrench $\mathbf{f}$ in $\mathbb{R}^n$. There exists a scalar $\alpha > 0$ such that $\|\alpha\mathbf{f}\| < r$. Hence, $\alpha\mathbf{f} \in [\mathbf{f}]$ and there exists $\boldsymbol{\tau} \in [\boldsymbol{\tau}]$ (thus $\boldsymbol{\tau} \geq \mathbf{0}$) such that $\mathbf{W}\boldsymbol{\tau} = \alpha\mathbf{f}$ for $\mathbf{X} \in$ WFW. Consequently, $\mathbf{W}(\boldsymbol{\tau}/\alpha) = \mathbf{f}$ where $\boldsymbol{\tau}/\alpha \geq \mathbf{0}$: $\mathbf{X} \in$ WCW.

Property 2 *For any bounded set of wrenches* $[\mathbf{f}]$ *and any pose* $\mathbf{X} \in$ *WCW, there exists a (finite) set of allowed cable tensions* $[\boldsymbol{\tau}]$ *such that* $\mathbf{X} \in$ *WFW.*

Proof. $\mathbf{X} \in$ WCW implies that there exists $\boldsymbol{\tau} > \mathbf{0}$ such that $\mathbf{W}\boldsymbol{\tau} = \mathbf{0}$ and, since $\text{rank}(\mathbf{W}) = n$, that for all $\mathbf{f}$ in $[\mathbf{f}]$, $\boldsymbol{\tau}^+ = \mathbf{W}^+\mathbf{f}$ exists and satisfies $\mathbf{W}\boldsymbol{\tau}^+ = \mathbf{f}$ where $\mathbf{W}^+ = \mathbf{W}^T(\mathbf{W}\mathbf{W}^T)^{-1}$ is a generalized inverse. Now, since $[\mathbf{f}]$ is bounded, the set $\left\{\boldsymbol{\tau}^+ = \mathbf{W}^+\mathbf{f} \mid \mathbf{f} \in [\mathbf{f}]\right\}$ is bounded and this set translated by the vector $\alpha\boldsymbol{\tau}$, denoted $\mathcal{S} = \left\{\boldsymbol{\tau}^+ + \alpha\boldsymbol{\tau} \mid \mathbf{f} \in [\mathbf{f}]\right\}$, is bounded too. Moreover, by correctly choosing α, namely

$$\alpha = \max_i \left(\max_{\mathbf{f}\in[\mathbf{f}]}((\tau_{\min} - \tau_i^+)/\tau_i)\right), \quad \tau_{\min} \geq 0 \tag{6}$$

$\mathcal{S}$ is fully included in the positive orthant of $\mathbb{R}^m$ and, thus, there exists a set of cable tensions $[\boldsymbol{\tau}]$ satisfying Eq. (2) such that $\mathcal{S} \subset [\boldsymbol{\tau}]$. Finally, for all $\mathbf{f}$ in $[\mathbf{f}]$, $\boldsymbol{\tau}^+ + \alpha\boldsymbol{\tau} \in [\boldsymbol{\tau}]$ and $\mathbf{W}(\boldsymbol{\tau}^+ + \alpha\boldsymbol{\tau}) = \mathbf{f}$: $[\boldsymbol{\tau}]$ is such that $\mathbf{X} \in$ WFW.

These developments motivate a two-step approach to the design problem introduced in Sections 2 and 3.

Step 1: find the vectors of geometric parameters $\mathbf{P}_{WCW}$ of robots for which the prescribed workspace $[\mathbf{X}]$ is fully included in the WCW.
Step 2: among the vectors $\mathbf{P}_{WCW}$, find those for which $[\mathbf{X}]$ is fully included in the WFW.

This two-step approach is completely justified by Property 1. Indeed, for any given vector of parameters $\mathbf{P}$, if $[\mathbf{X}]$ is not fully included in the WCW, it is necessarily not fully included in the WFW since the WFW is a subset of the WCW. Hence, the WCW depending only on the vector of geometric parameters $\mathbf{P}$, Step 1 allows to remove from the search all robots which are not solutions to the design problem due to an inadequate geometry. Then, Step 2 searches among the remaining robots those for which the set of allowed cable tensions $[\boldsymbol{\tau}]$ is large enough for $[\mathbf{X}]$ to be fully included in the WFW. Moreover, if no such robots are found, Property 2 insures that $[\boldsymbol{\tau}]$ can always be enlarged so as to find robots, among those not eliminated at Step 1, such that $[\mathbf{X}]$ is fully included in the WFW. Note also that Step 2 can be applied several times for various $[\boldsymbol{\tau}]$ without considering again and again vectors $\mathbf{P}$, namely those eliminated at Step 1, for which $[\mathbf{X}]$ is never fully included in the WFW whatever the size of the set $[\boldsymbol{\tau}]$.

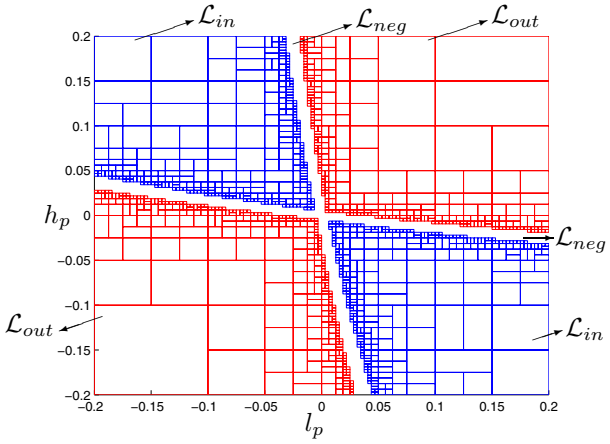


Fig. 4 Result of Step 1 for the example of Section 3. $\mathcal{L}_{in} \equiv$ approximation of robots geometries such that $[\mathbf{X}] \subset$ WCW.

5 Example

The numerical method presented in [3] has been adapted to the case of the WCW enabling the application of the algorithm of Figure 1 to the case of the WCW and, thus, the practical realization of Step 1 of Section 4.

For instance, the application of Step 1 to the example of Section 3 yields the result shown in Figure 4. The boxes of the list $\mathcal{L}_{in}$ provide an approximation of the geometric parameters $\mathbf{P}_{WCW}$ of robots such that the prescribed workspace $[\mathbf{X}]$ is completely included in the WCW. On the contrary, the boxes of the list $\mathcal{L}_{\text{out}}$ correspond to geometric parameters of robots for which $[\mathbf{X}]$ is not fully included in the WCW. According to Property 1 of Section 4, whatever the size and shape of the sets $[\mathbf{f}]$ and $[\boldsymbol{\tau}]$, $[\mathbf{X}]$ is not fully included in the WFW of a robot whose vector of geometric parameters lies in $\mathcal{L}_{\text{out}}$. Note that the question asked at the beginning of Section 4 can now be answered: as seen in Figure 4, for the problem at hand, no robot with non-crossed cables – l_p and h_p nonnegative – have $[\mathbf{X}]$ fully included in the WCW. Again, by Property 1, whatever the size of the sets $[\mathbf{f}]$ and $[\boldsymbol{\tau}]$, none of the robots with non-crossed cables will have $[\mathbf{X}]$ fully included in the WFW due to the considered set of geometries $\mathbf{P}$ which is too restrictive.

Finally, the realization of Step 2 of Section 4 consists in applying the branch-and-prune algorithm presented in Figure 1 with the list $\mathcal{L}$ initialized with the list of boxes $\mathcal{L}_{in}$ shown in Figure 4 instead of the search box $\mathcal{B}$ (Step 1 in Figure 1). This allows to exclude from the search the parts of the l_p-h_p plane covered by the boxes of $\mathcal{L}_{\text{out}}$ (shown in Figure 4) which can never correspond to geometric parameters $\mathbf{P}$ for which $[\mathbf{X}]$ is fully included in the WFW. This property of the two-step approach proposed in Section 4 is especially useful when Step 2 is repeated several times for different WFW, for example, when, for a given required set of wrenches $[\mathbf{f}]$, we search for the set of allowed cable tensions $[\boldsymbol{\tau}]$ with the smallest $\tau_{\max}$ for which

there exist geometric parameters $\mathbf{P}$ such that $[\mathbf{X}]$ is fully included in the WFW. To this end, Step 2 can be repeated several times with different sets of tensions $[\boldsymbol{\tau}]$, beginning the process with a large $[\boldsymbol{\tau}]$. In fact, in this case, the computationally intensive branch-and-prune algorithm presented in Figure 1 is called each time Step 2 is repeated. Therefore, Step 1 allows to avoid considering again and again the boxes of $\mathcal{L}_{\text{out}}$ which, whatever the set $[\boldsymbol{\tau}]$, never correspond to robots for which the prescribed workspace $[\mathbf{X}]$ is fully included in the WFW and, consequently, Step 1 avoids many useless computations.

6 Conclusions

This paper shows how numerical methods, introduced elsewhere, can be used to determine solutions to the design problem of finding geometries of fully constrained parallel cable-driven robots for which a given prescribed workspace is fully included in the WFW. In this context, the study of the WCW is of interest since, contrary to the WFW, it depends only on the geometry of the cable-driven robot. Moreover, the WFW is proved to be a subset of the WCW. These two properties allow to utilize the WCW in order to filter out, once and for all, robots geometries for which the prescribed workspace is not fully included in the WFW whatever the size of the set of allowed cable tensions.

References

1. Bosscher, P., Riechel, A.T. and Ebert-Uphoff, I., Wrench-feasible workspace generation for cable-driven robots, *IEEE Trans. on Robotics* **22**(5), 890–902 (2006).
2. Fattah, A. and Agrawal, S.K., On the design of cable-suspended planar parallel robots, *ASME J. Mech. Design* **127**(5), 1021–1028 (2005).
3. Gouttefarde, M., Merlet J.-P. and Daney, D., Wrench-feasible workspace of parallel cable-driven mechanisms, in *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1492–1497, Roma, Italy (2007).
4. Hiller, M., Fang, S., Mielczarek, S., Verhoeven, R. and Franitza, D., Design, analysis and realization of tendon-based parallel manipulators, Mechanism and Machine Theory, **40**(4), pp. 429–445 (2005).
5. Merlet, J.-P., Optimal design of robots, in *Proc. Robotics: Science and Systems*, Cambridge, MA, USA (2005).
6. Merlet, J.-P., *Parallel Robots*, 2nd ed., Springer, Dordrecht (2006).
7. Pham, C.B., Yeo, S.H., Yang, G., Kurbanhusen, M.S. and Chen, I-M., Force-closure workspace analysis of cable-driven parallel mechanisms, *Mech. and Mach. Theory* **41**, 53–69 (2006).
8. Roberts, R.G., Graham, T. and Lippitt, T., On the inverse kinematics, statics, and fault tolerance of cable-suspended robots, *Journal of Robotic Systems* **15**(10), 581–597 (1998).
9. Stump, E. and Kumar, V., Workspaces of cable-actuated parallel manipulators, *ASME J. Mech. Design* **128**(1), 159–167 (2006).

Synthesis of Part Orienting Devices for Spatial Assembly Tasks

P. Larochelle

Robotics & Spatial Systems Lab, Department of Mechanical and Aerospace Engineering, Florida Institute of Technology, Melbourne, FL 32901-6975, U.S.A., e-mail: pierrel@fit.edu

Abstract. A novel dimensional synthesis technique for solving the mixed exact and approximate motion synthesis problem for spherical RR kinematic chains is presented. The methodology uses an analytic representation of the spherical RR dyad's rigid body constraint equation in combination with classical geometric constructions for exact motion synthesis to yield designs that exactly reach two of the prescribed orientations while approximating the remaining. The result is a mixed exact and approximate motion dimensional synthesis technique that is applicable to spherical open and closed kinematic chains. Here, we specifically address the design of spherical RR open and 4R closed chains since they form the foundation of a new class of devices being developed called PODs or Part Orienting Devices. An example that demonstrates the synthesis technique is included.

Key words: spherical mechanisms, exact motion synthesis, approximate motion synthesis.

1 Introduction

As a product is assembled in an automated factory, both the product and its individual parts are picked up, reoriented and inserted into subassemblies or fixtures. For a complex product, the number of manipulations could run into the thousands. Parts are picked out of bins and placed into assemblies. Partial assemblies are rotated to allow additional parts to be added. Fasteners are inserted to hold everything together.

Typically, designers of assembly lines seek to keep the manipulations as simple as possible. Rotations about vertical or horizontal axes are preferred, often of 90 or 180 (deg). These tasks have a well established set of solutions. However, operations which involve a translation along and/or a rotation about an axis which is not vertical or horizontal is more challenging to the designer. Additional constraints on the trajectory of the object (e.g. obstacle avoidance, or part meshing) increase the difficulties. One solution is to use devices with a high number of degrees of freedom, such as industrial robots. Robots can perform the tasks, but at penalties in costs, setup time, and maintenance. A second solution is to use a cascading series

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 79–87.

of simple one degree of freedom devices. Creating this manipulation pipeline takes a longer design time and is often more art than science.

Part Orienting Devices (PODs) offer another alternative. The synthesis algorithm presented here is part of ongoing efforts directed at realizing the capability to design these devices for spatial assembly tasks. These low degree-of-freedom devices are capable of producing the necessary spatial reorientations often required in spatial assembly tasks. Hence, PODs provide an alternative for solving spatial assembly tasks that might otherwise require a robot or multiple single degree-of-freedom devices.

A well known result from screw theory is that moving an object from one spatial location to another does not require six degrees of freedom. In fact, such motions can be accomplished with a single degree of freedom twist about a screw axis. However, it is rare that this solution is practical due to the location of the screw axis within the workspace and the collisions and interferences between objects that result. PODs are low degree-of-freedom machines that are a compromise between the 6 or more degree of freedom industrial robot and the single degree of freedom twisting motion. Here, the focus is on utilizing the spherical 4R closed chain architecture to serve as the motion generator for a class of PODs to achieve two desired orientations exactly while approximating a set of guiding orientations that take the workpiece from one exact orientation to the other.

In related works [7] present the derivation of the constraint manifold for spherical RR dyads using the image space representation of displacements. Their work was an extension of the ideas presented in [8]. In [11] the design of spherical mechanisms to approximate spatial locations is presented. A robust synthesis algorithm for spherical motion generation was presented by Al-Widyan and Angeles [1]. More recently, [2] present the synthesis of spherical 4R mechanisms for 5 prescribed orientations. Related ongoing efforts at the University of Dayton to advance the design of PODs have been reported in [3, 6]. The methodology used here for performing the dimensional synthesis for mixed exact and approximate orientation rigid body guidance is based upon the works of Tsai and Roth [10] and McCarthy [5].

This paper proceeds as follows. First, the geometry and kinematics of the spherical RR dyad and the spherical 4R closed chains are reviewed. Next, the synthesis algorithm for solving the mixed exact and approximate motion synthesis problem for spherical RR kinematic chains is presented. Finally, an example POD design is presented; the synthesis of a spherical 4R closed chain to achieve two prescribed orientations exactly while approximating three guiding orientations.

2 Synthesis Algorithm

A spherical 4R closed chain may be viewed as the combination of two spherical RR dyads where each dyad consist of two R joints; one fixed and the other moving, see Figure 1. The approach taken here is to synthesize two dyads separately and then join their floating links to yield a kinematic closed chain. Let the fixed axis be specified by the vector $\mathbf{u}$ measured in the fixed reference frame F and let the

moving axis be specified by $\mathbf{v}$ measured in the moving frame M. Moreover, let $\mathbf{l}$ define the moving axis $\mathbf{v}$ in the fixed frame F so that, $\mathbf{l} = [A]\mathbf{v}$ where $[A]$ is the element of SO(3) that defines M with respect to F. Because the link is rigid, the angle between the two axes of the dyad remains constant. This geometric constraint may be expressed analytically as,

$$\mathbf{u} \cdot \mathbf{l} = \mathbf{u} \cdot [A]\mathbf{v} = \cos\alpha. \tag{1}$$

This constraint equation is the foundation of the synthesis algorithm presented below. In order to solve the mixed exact and approximate synthesis problem we first solve the exact synthesis problem for 3 prescribed orientations.

2.1 Exact Synthesis for Three Orientations

First, we select the moving axis $\mathbf{v}$. Second, we write Eq. (1) for each of the desired orientations, $[A]_i$, $i = 1, 2, 3$. Finally, we subtract the first equation from the remaining two to arrive at a linear system of equations,

$$[P]\mathbf{u} = \mathbf{b}, \tag{2}$$

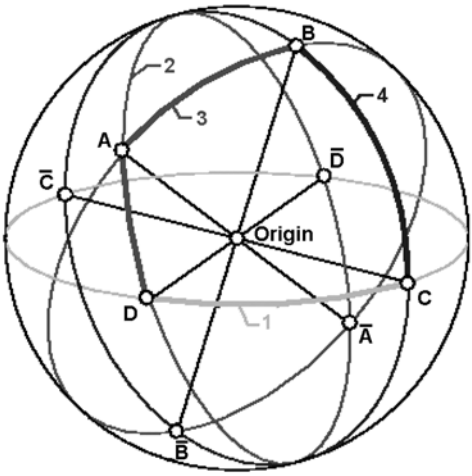
where

$$[P] = \begin{bmatrix} \mathbf{v}^T([A]_2 - [A]_1)^T \\ \mathbf{v}^T([A]_3 - [A]_1)^T \\ 0 \quad 0 \quad 1 \end{bmatrix},$$

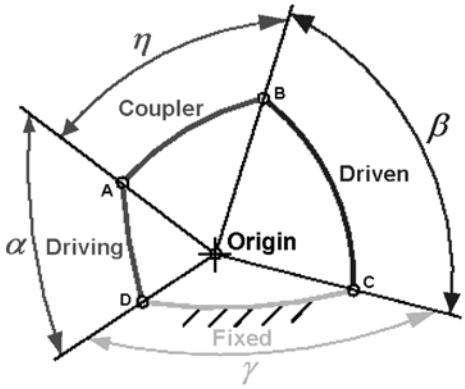
$\mathbf{b} = [0\ 0\ 1]^T$, and $\mathbf{u}$ is the desired fixed axis. Note that we must solve Eq. (2) for each prescribed moving axis to find its corresponding fixed axis. Moreover, note that since we are using 3-vectors to define the axes when in fact they are simply directions that only require 2 coordinates, the last row of $[P]$ is chosen to yield the vector $\mathbf{u}$ that is the intersection of the fixed axis with the $z = 1$ plane. In the event that $[P]$ is rank deficient (i.e. when the fixed axis does not intersect the $z = 1$ plane) simply change the last row to any vector that does not lie in this plane (e.g. $[1\ 0\ 0]^T$).

2.2 Mixed Synthesis Algorithm

In the problem considered here we have 2 orientations to reach exactly and n orientations that serve to guide the body from one exact orientation to the other. First a desired moving axis $\mathbf{v}$ is selected. Next, we seek a corresponding fixed axis for the dyad. The fixed axis is found by solving n 3 orientation problems to yield a set of fixed axes $\mathbf{u}_i$, $i = 1, 2, \ldots, n$. The 3 orientation problems are derived from the 2 exact orientations along with 1 of the guiding orientations. Hence, there are n unique



a) Great Circles



b) Spherical Linkage Nomenclature

Fig. 1 Spherical 4R geometry and nomenclature.

3 orientation problems. We select the fixed axis $\mathbf{u}$ that is their normalized sum,

$$\mathbf{u} = \frac{\sum \mathbf{u}_i}{\| \sum \mathbf{u}_i \|}. \tag{3}$$

It is beneficial to discuss the geometry underlying this approach. Consider the synthesis of a spherical RR dyad for two exact orientations. Associated with the desired moving axis is a great circle that is defined by the set of all fixed axis that solve the problem. Now consider another orientation and one of the original 2 orientations.

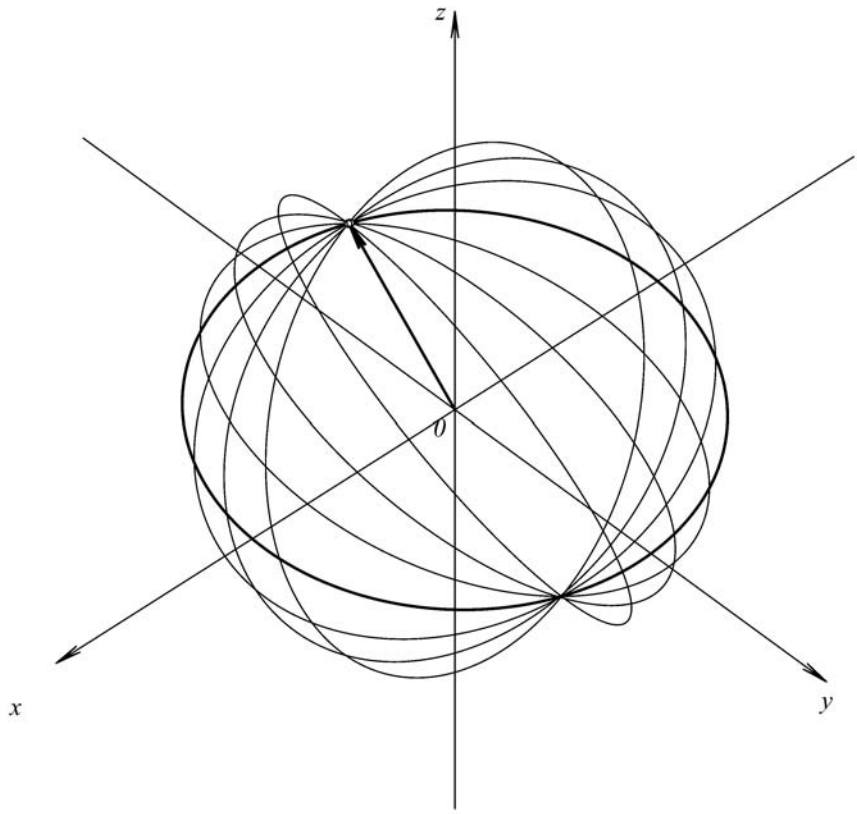


Fig. 2 The great circles associated with dyad #1.

For a desired moving axis there is again a great circle that represents all of the solution fixed axes. Fixed axes that guide a body through all 3 orientations must lie at the intersections of these two great circles. Generally, these great circles intersect in two points that define line; hence there is 1 unique fixed axis associated with 3 spherical orientations and a choice of moving axis. Finally, consider the exact 3 orientation problem. The desired fixed axis lies at the intersection of 3 great circles; the first associated with orientations 1 & 2, the second with 2 & 3, and the third with 1 & 3. By solving all of the 3 orientation problems that include the two exact orientations we guarantee that all resulting fixed axes lie on the great circle associated with these two orientations. Moreover, in Eq. (3) we utilize the fixed axis that lies on the great circle associated with the two exact orientations and that is nearest the great circles associated with the guiding orientations. Hence, the solution dyad will guide the part exactly through the two prescribed orientations and nearest the guiding orientations for the selected moving axis.

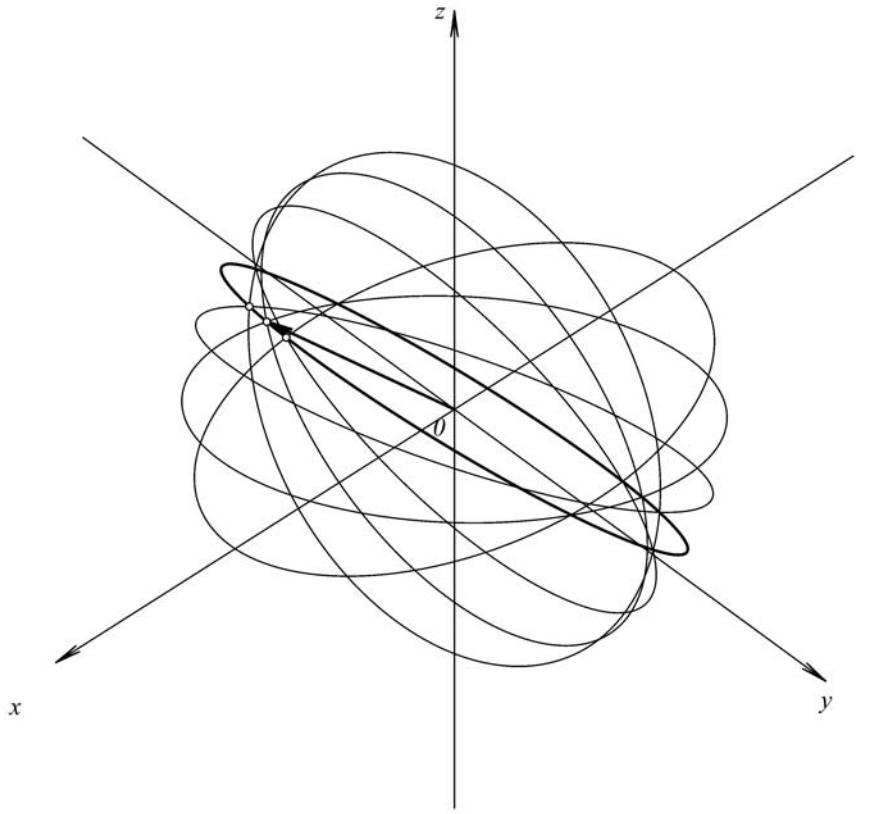


Fig. 3 The great circles associated with dyad #2.

3 Example

A common task in spatial assembly operations is the reorienting of a part by successive rotations of 90 (deg) about two orthogonal axes; the so called *90-90* problem. Here we employ the preceding methodology and design a POD to achieve the desired motion by synthesizing a spherical 4R mechanism for 5 orientations; 2 exact (the starting orientation and the final orientation after the 90-90 rotations) and 3 guiding orientations as defined in Table 1 where $[A] = [Rot_z(lng)][Rot_y(-lat)][Rot_x(rol)]$. In order to prescribe the size of the coupler link and to eliminate the need for any extension or attachment to connect the moving body to the coupler these moving axes were selected: $\mathbf{v}_1 = [1\ 1\ 1]^T$ and $\mathbf{v}_2 = [1\ 0\ 0]^T$ (see [4, 9]). The mixed synthesis algorithm yielded fixed axes: $\mathbf{u}_1 = [0.2745, -0.2745, 0.9216]^T$ and $\mathbf{u}_2 = [0.6780, -0.2839, 0.6780]^T$. The great circles that illustrate the application of the algorithm to determine $\mathbf{u}_1$ are shown in Figure 2 and those associated with $\mathbf{u}_2$ are shown in Figure 3. The great circle as-

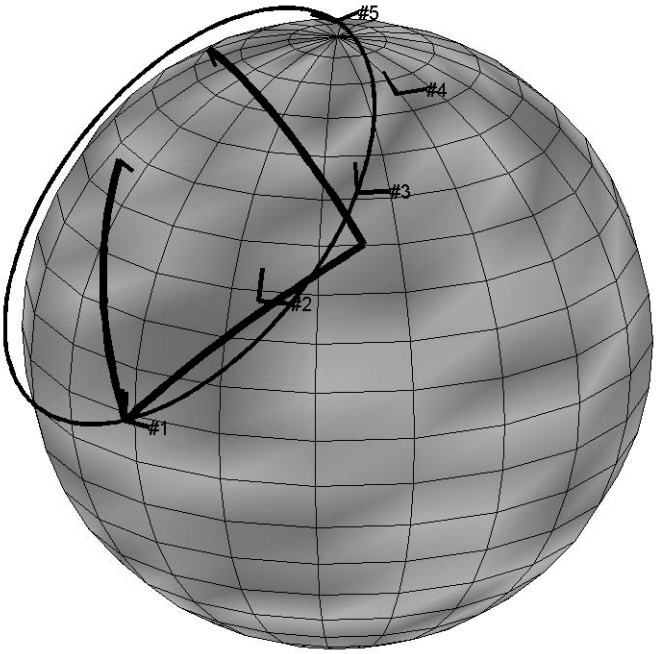


Fig. 4 The POD shown in orientation #1.

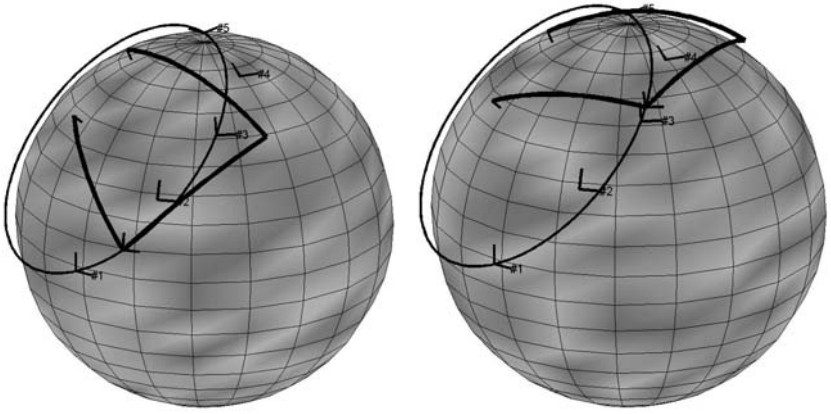


Fig. 5 Moving from orientation #1 to #2 (left) and from #3 to #4 (right).

sociated with the 2 exact orientations is thicker and the 3 fixed axes associated with orientations 1-2-3, 1-3-5, and 1-4-5 are indicated by ∘ symbols on the great circle. Recall that these axes are used in Eq. (3) to determine $\mathbf{u}_i$. The resulting POD is a Grashof double-crank spherical four-bar mechanism that does not suffer from circuit, branch, or order defects. Its link lengths are: $\alpha = 57.8512$, $\eta = 54.7321$,

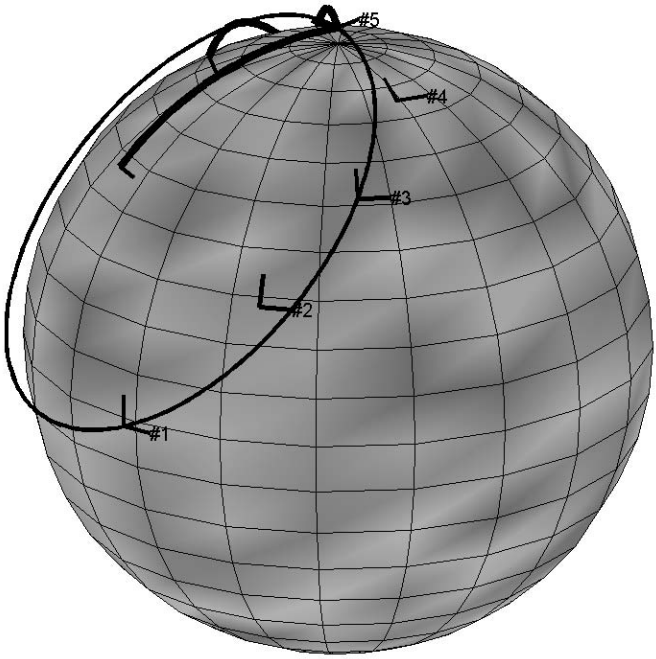


Fig. 6 The POD shown in orientation #5.

Table 1 Five prescribed orientations.

#	Longitude	Latitude	Roll	Motion Type
1	0.00	0.00	0.00	exact
2	25.00	25.00	0.00	approximate
3	45.00	45.00	0.00	approximate
3	65.00	65.00	0.00	approximate
5	90.00	90.00	0.00	exact

$\beta = 47.3688$, and $\gamma = 27.2574$ and the moving body is attached at 135 (deg) to the coupler at the driven moving axis. The solution POD is shown in Figures 4–6.

4 Conclusions

A novel dimensional synthesis technique for solving the mixed exact and approximate motion synthesis problem for spherical RR open and 4R closed kinematic chains has been presented. The methodology uses an analytic representation of the spherical RR dyad's rigid body constraint equation in combination with classical geometric constructions for exact motion synthesis to yield designs that exactly reach two

of the prescribed orientations while approximating the remaining guiding orientations. Such part orienting tasks are common in automated assembly systems. Here, we specifically address the design of spherical RR open and 4R closed chains since they serve as the motion generators for a class of PODs, or part orienting devices, that are being developed.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. #0422705. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

1. Al-Widyan, K. Angeles, J., The synthesis of robust spherical motion generators, in *Proc. of the 2004 ASME Design Engineering Technical Conferences*, Paper # DETC2004-57422, Salt Lake City, Utah (2004).
2. Brunnthaler, K., Schröcker, H.-P., Husty, M., Synthesis of spherical four-bar mechanisms using spherical kinematic mapping, in Lenarčič, J. and Roth, B. (Eds.), *Advances in Robot Kinematics: Mechanisms and Motion*, Springer, pp. 377–384 (2006).
3. Grimm, E., Murray, A., Turner, M., Software for the kinematic synthesis of coupler-driven spherical four-bar mechanisms, in *Proc. of the ASME Design Engineering Technical Conferences*, Paper # DETC2007-35720, Las Vegas, NV (2007).
4. Ketchel, J., Larochelle, P., Computer-aided manufacturing of spherical mechanisms, *Mechanism and Machine Theory* **42**(2), 131–146 (2007).
5. McCarthy, J.M., *Geometric Design of Linkages*, Springer-Verlag (2000).
6. Murray, A., Pierrot, F., Design of a high-speed spherical four-bar mechanism for use in a motion common in assembly processes, in *Proc. of the ASME Design Engineering Technical Conferences*, Paper # DETC2007-35354, Las Vegas, NV (2007).
7. Ravani, B., Roth, B., Motion synthesis using kinematic mappings, *ASME Journal of Mechanisms, Transmissions, and Automation in Design* **105**, 460–467 (1983).
8. Roth, B., The design of binary cranks with revolute, cylindric, and prismatic joints, *Journal of Mechanisms* **3**(2), 61–72 (1968).
9. Schuler, J., Ketchel, J., Larochelle, P., Computer-aided modeling and manufacturing of spherical mechanisms via a novel web tool, *ASME Journal of Computers and Information Science in Engineering* **7**(4), 339–346 (2007).
10. Tsai, L.W., Roth, B., Design of dyads with helical, cylindrical, spherical, revolute, and prismatic joints, *Mechanism and Machine Theory* **7**(1), 85–102 (1973).
11. Tse, D.M., Larochelle, P.M., Approximating spatial locations with spherical orientations for spherical mechanism design, *ASME Journal of Mechanical Design* **122**, 457–463 (2000).

Minimum Energy Manipulator Design

A. Rojas Salgado and Y. Ledezma Rubio

Departamento de Mecánica, Facultad de Ingeniería, UNAM, Mexico DF, Mexico, e-mail: arojas@servidor.unam.mx, yahve_ledezma@hotmail.com

Abstract. Research in the optimum design of a manipulator has taken different directions. One of those was to define the kinematic or dynamic parameters that determine the characteristics of the manipulator in order to justify the best design. In most of the studies that are under way, the possible solutions are restricted to one feasible region in which all of the geometrical and dynamic restrictions as well as the power input on the drives have to be met with. However, all these approaches lead to multiple solutions, since they are all within the feasible region, and it is as of yet unknown which of those solutions is best. This paper tries to give a solution through an energy approach where the optimum design in a multiple link system can be chosen, taking into account the trajectory and the structure of the links, be it a serial or a parallel link system.

Key words: design, manipulators, energy saving, parallel singularities.

1 Introduction

The definition of the optimum design of a mechanical system depends of the approach to it. The optimum design could be the most economic, the one that uses less infrastructure for its construction, the one that lasts longest in its use, or the one with the least weight. In order to be able to know which the best solution is in a specific case, it would depend on what it is, specifically, you are looking for. For example, in aerospace systems the highest concern would be weight and stress tolerance.

In the case of manipulator design there are the following approaches.

1.1 Serial Manipulators

A first approach would be to define a feasible region in which the optimum design of the manipulator has to be found. This region considers the geometrical restrictions intrinsic of the manipulator, through inverse kinematic of the model, and the restrictions of the angles that can be reached by each joint, taking into account the points

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 89–99.

of the trajectory to be executed. This approach does not lead to an exact solution; it only defines the feasible region of the design [9].

Other papers use the concept of the isotropic matrix within which the values of the best dynamic condition of a manipulator are found, either serial or parallel. This model involves dynamical values such as mass and inertial moment, and tries to improve control of the movement of the manipulator. Different optimum solutions are reached, since it is a region that is defined, not a point [8].

Other studies suggest that the design of manipulators should depend on the energy and the power input in the actuators that are necessary to complete the task. Here, the point of origin of the manipulator is modified, while the dimensions of the links are constant. In this case only key points of the trajectory are taken into account, not the whole trajectory [10].

Others take a different approach to the design problem of the RRR manipulator: through five points of a trajectory the concept of interval analysis is applied. They try to find the right dimensions of the links by means of the Denavit and Hartenberg parameters, using a numerical method that requires utmost computer resources. Here solutions are also reached within the feasible region, however, with no indication as to which one is the best [7]. The efficient use of applied energy to the design is another approach that can be found within the proposed models for special RRR manipulators [5], where tubular sections with thin walls aiming for less mass and sufficient strength, lower the energy demands of the manipulator.

1.2 Parallel Manipulators

Parallel manipulators require a higher complexity for their solution, since they do not only present serial singularities that can be identified by their geometric configuration, but also parallel singularities, such as the geometric points in which control over movement of the final link is lost.

One of the first steps would be to define the singularities for this specific model [8, 10], through which the conditions for losing control of the manipulator are identified. These singularities will later be helpful in defining the regions in which the movement can take place.

Another focus for solving parallel manipulators is the one described in [6], where kinematics and inversed dynamics are applied. Due to the complex equations generated and the arising difficulty in their manipulation, the ideas of Newton-Lagrange are not taken into account here. In order to find solutions for these systems, the recursive approach is preferred.

More recent studies of parallel manipulators [7, 9] are focused on a possible definition of regions through parallel and serial singular matrix, in order to find a solution for a defined trajectory. In these cases systems are used that leave one free variable, with which can be worked to avoid the singularities the system is exposed to.

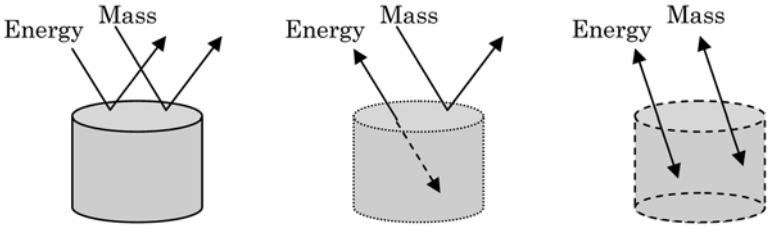


Fig. 1 Kinds of systems.

1.3 Optimization Approach

The optimization model that is presented in this paper is based on the mechanical energy of the model that is necessary in order to follow a defined trajectory. This trajectory cannot be changed, be it because of the specific work it is executing or because it has been optimized and it is considered to be the best trajectory. Examples would be painting or welding procedures, where trajectory cannot be changed.

In our example, the software Mathematica 6 is used to develop the corresponding equations, as well as in the search for the optimum point of the design.

2 Theoretical System Model

In order to define the system, concepts of classical thermodynamics will be used, including three different kinds of systems: insulated systems, closed systems, and open systems. As can be seen in Figure 1, the interactions between the system and its environment define the kind of system that has to be used. The boundaries of the system must be defined, for the interaction depends on them.

The property we are going to analyze is energy. Energy depends on the current conditions of the system, such as velocity and position of the center of mass. The equation of energy can then be expressed as follows:

$$\delta Q + \delta W^{\text{ext}} + \delta W^{\text{int}} = dE^{\text{ext}} + dE^{\text{int}}. \tag{1}$$

The interactions between the system and its environment we are interested in refer solely to one part of those energies. At this time it is not taken into account that there are heat flows within the movement of the mechanical system, nor changes in its internal energy, nor expansion work or deformation work. This leads us to the simplified equation:

$$\delta W^{\text{ext}} = dEP_{\text{Grav}} + dEP_{\text{Electr}} + dEP_{\text{Magnet}} + dEC_{\text{Lin}} + dEC_{\text{Rot}}. \tag{2}$$

Out of these energies there will only be taken into account those of mechanical origin, giving no consideration to the transformation of energy that exists within the

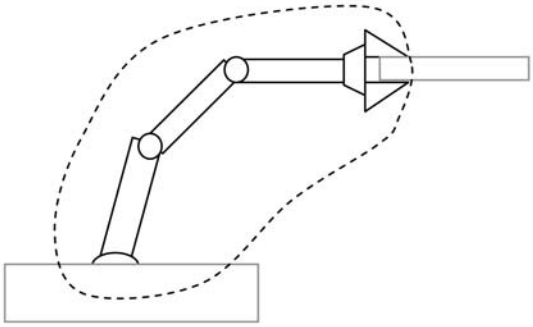


Fig. 2 System boundaries.

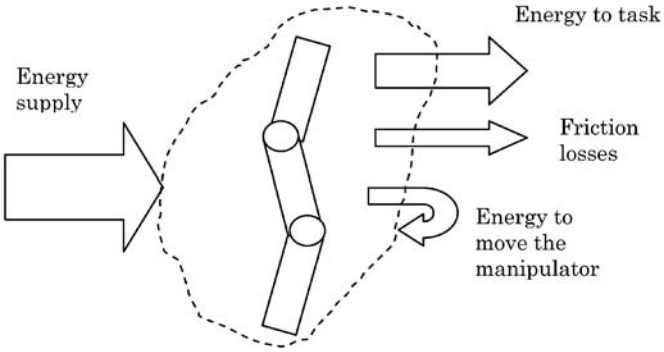


Fig. 3 Energy flow.

actuators. The model can be represented in the following schematic drawing. It is made up by an inertial base of the mechanical system which provides the necessary energy for the movement of the drivers, the system links, from the base to the effector link and the object to be moved or task to be done. The boundaries of the system are around the mechanism links, being closed boundaries allowing the flow of work and energy through them, however, not allowing mass exchange (Figure 2).

The next step consists in identifying the energy or work flows that have an effect on the system. The energy that enters from the inertial base to the manipulator is identified as the total amount of energy that enters the system. Next is the energy that flows from the manipulator to the task or object to be moved. This is also identified as loss of energy through heat by friction in the drivers and joints. However, there still exists part of the energy within the process, which is the one needed to move the manipulator links between the different positions. This is not a real flow across boundaries, because the energy is used up by the system (Figure 3).

For this model the following simplifications are used: the links are solid rigid, reason why the elastic effects and the expansion effects are not taken into account. On the other hand, the drivers involved won?t present losses through friction. These simplifications leave us with a model with only one entry, the provided energy to

the manipulator, and one exit, the energy that goes to the task to be executed. We are searching for the highest efficiency through least use of energy by the system links. The amount of energy required to execute the task is generally fixed, such as is the lifting of a weight or causing movement of different mechanisms. The remaining flow which causes the movement of the manipulator is the one that is to be optimized.

3 Method of Analysis

In order to obtain the optimum point the following steps have to be observed:

1. *Trajectory function*. An analytical function is needed that describes the task to be executed, be it with restrictions of position, velocity or acceleration required for certain points in order to obtain a continuum to evaluate.
2. *Work ratio and feasible region*. The work ratio is determined by the task to be executed, and the feasible region is determined by geometry and restrictions of the manipulator. With these restrictions the serial and parallel singularities have to be considered, since they define the feasible region in the manipulators. The mechanical system we analyze may have geometrical restrictions, assembly problems, and limitations in the angles that can be reached. These restrictions are to be expressed through the parameters of the design that is to be optimized. In this case, the analysis will be restricted by the geometric configurations and Jacobian matrix only, and the parameters of the design are the lengths of the links.
3. *Energy function*. A function is obtained by the trajectory and the dynamical properties of the model, like mass and inertia. This function includes all the potential and kinetic energy of the link system. The variables of this function are the design parameters, in this case, the lengths of the links.
4. *Search for the optimum design point*. Through the energy function of the system the minimum point of process energy is obtained. It will be within the feasible region. The search will be analytical or numerical. It depends on the complexity of the trajectory and the geometrical configuration of the mechanical system. The points with serial singularities will be automatically eliminated because they would need infinite energy to reach that position; however, the parallel singularities remain.

The complexity of the analysis is directly related to the complexity of the trajectory and the proposed mechanical model. It depends on inverse kinematics associated with trajectory and design parameters. Multiple configurations can be obtained, and all of those will be evaluated in order to determine which is the best one.

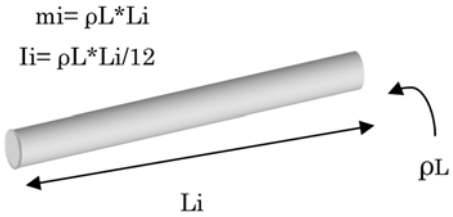


Fig. 4 Mass and inertia of link i.

3.1 Energy Equations

The resulting equations will have to show the mechanical energy of the model, depending on its design dimensions (Figure 4). The mass (m) depends on the length of the link (Li) and the density of the bar (ρL). On the other hand, the inertia of each link also depends on its length and shape, defined in this case as constant circular section links that represent a bar.

Here the kinematics will be functions of the lengths of the links of the system. Different configurations can be found as a solution to a given trajectory. For example, in the case of serial manipulators of two and three links, an upper configuration (A) and a lower configuration (B) arise, which would be the solutions for the inverse kinematics model proposed.

For each of link i the following energy equation applies:

$$ECline al_i = \frac{m_i v_i^2}{2}, \tag{3a}$$

$$EAngular = \frac{I_i \omega_i^2}{2}, \tag{3b}$$

$$EPot_i = m_i g h_i, \tag{3c}$$

$$E_{esl\,i} = ECLineal_{esl\,i} + ECAngular_{esl\,i} + EPot_{esl\,i}, \tag{3d}$$

$$Etot = \sum_{i=1}^{n} E_{esl\,i}, \tag{3e}$$

where v_i stands for the velocity of the center of mass i, ω_i is the angular velocity of the link i, and h_i is the distance to a point of reference with respect to the gravitational field. $Etot$ is given with respect to the design dimensions and the trajectory (Eq. (4)), and we can do a search of which is the point of less energy and establish it as the optimum.

$$Etot = [x(t), y(t), L1, L2, \ldots, Li]. \tag{4}$$

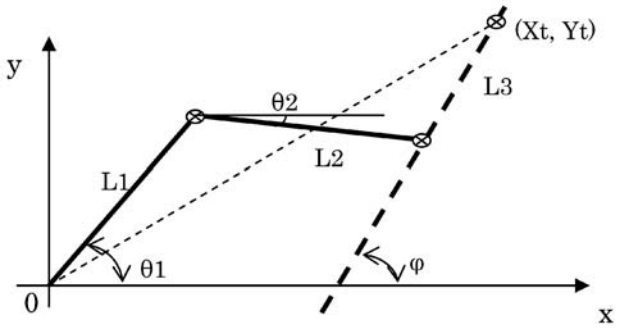


Fig. 5 Geometrical model of RRR serial manipulator.

4 Case Studies

This method can be applied to any kind of kinematic chain, be it open or closed, with one or more degrees of freedom. Two numerical simulations with respect to the behavior of two representative systems have been executed: one serial RRR manipulator, and one 3RRR parallel platform.

4.1 Serial Manipulator

With a serial manipulator of three degrees of rotational freedom in a plane movement, straight, circular, polynomic and trigonometrical trajectories have been described. The trajectories simulate any task to be executed, as can be the transfer of an object or a task of complex painting. The trajectory the manipulator has to follow is given by

$$x(t) = \cos(t), \tag{5a}$$

$$y(t) = \sin(t); \quad 0 \le t \le \frac{\pi}{2}, \tag{5b}$$

$$\varphi(t) = 2t. \tag{5c}$$

The inverse kinematics of the model leads to two possible solutions; an upper and a lower configuration. Figure 5 represents the upper configuration.

The equation for optimizing is

$$EGlobal = \sum_{t_0}^{t_f} Etot = f[x(t), y(t), \varphi(t), L1, L2, L3]. \tag{6}$$

The restrictions that are geometrically imposed are caused by the serial singularities that the manipulator may face. They are directly related to the work radius of the manipulator. For each of those cases two different configurations have been

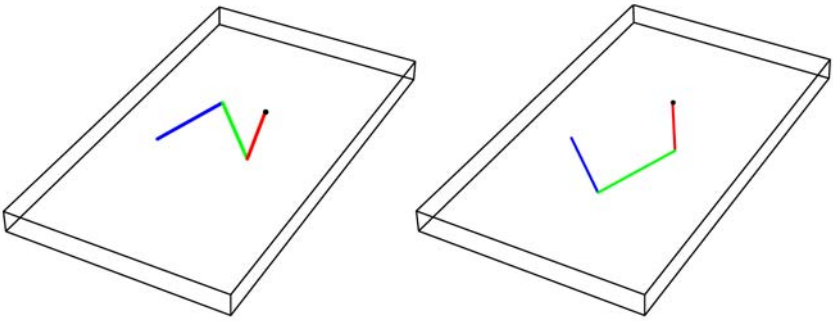


Fig. 6 Serial RRR manipulator solutions A and B.

encountered, each with the same lengths of links. The only difference between the two configurations is in potential energy (Figure 4).

$$L1 = 0.769679 \text{ [m]},$$

$$L2 = 0.835033 \text{ [m]},$$

$$L3 = 0.691782 \text{ [m]}.$$

With respect to global energy, there has to be said that the configuration with the arm in the lower position (B) has less energy than the configuration with the arm in upper position (A), as you can see at continuation.

$$EGlobal\ A = 1855.75 \text{ [J]},$$

$$EGlobal\ B = 1487.78 \text{ [J]}.$$

For each of those cases two different configurations have been encountered, each with the same lengths of links. The only difference between the two configurations is in potential energy, due to their position (Figure 6).

4.2 Parallel Manipulator

The parallel manipulator is composed of a triangular plate connected by three arms of two links with rotational articulations (3RRR). Here, the simulations consisted of giving position and orientation to the triangular plate with unknown dimensions, and to find out which is the configuration and length of the seven links of the manipulator that requires the least amount of energy. In order to simplify matters, the near and far links will be considered as of the same length, which reduces the variables to three lengths.

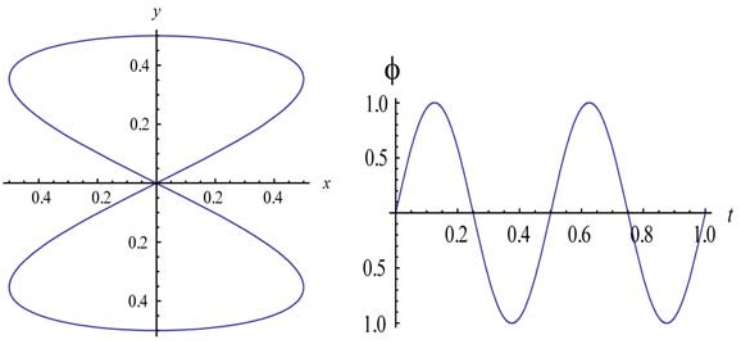


Fig. 7 Trajectory task of 3RRR manipulator.

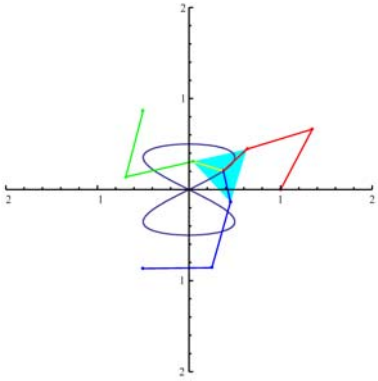


Fig. 8 Proposed model, configuration AAA.

The proposed trajectory is explained in Eq. (7) with the following graphic (Figure 7):

$$x(t) = 0.5\sin(4\pi t), \tag{7a}$$

$$y(t) = 0.5\cos(2\pi t), \tag{7b}$$

$$\varphi = \sin(4\pi t). \tag{7c}$$

The potential energy was not taken into account, positioning the plate and the arms in a plane movement perpendicular to the gravitational field. The results obtained favored those lengths that adjusted best to the movement within the feasible region, taking into account that the external singularities of this model are eliminated because of their need of infinite energy in order to reach those positions (Figure 8).

Here the definitions of the Jacobian parallel and serial matrix have been used in [1], resulting in a definition of the feasible region. The equation of energy remains as in Eq. (6), with infinite energy for the serial singularities. Therefore they have been eliminated in order to find the optimum point (Figure 9).

The solution for this trajectory is given by

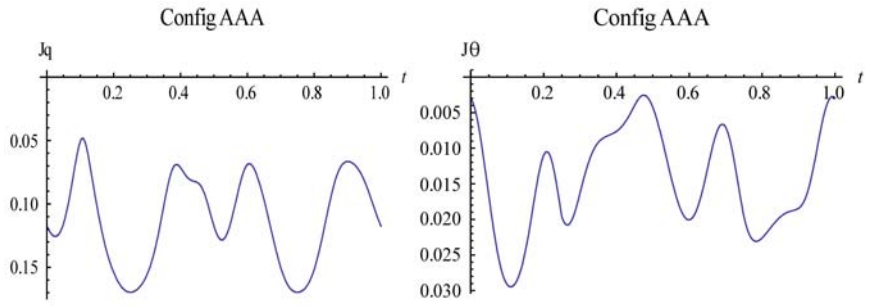


Fig. 9 Jacobian matrix, serial (Jq) and parallel ($J\theta$).

$$EGlobal = 2503.98 \text{ [J]},$$
$$L1 = 0.7775 \text{ [m]},$$
$$L2 = 0.7778 \text{ [m]},$$
$$L3 = 0.1500 \text{ [m]}.$$

5 Conclusions

The method of least energy demand offers an analysis that makes the lengths of the links dependent on the proposed trajectory. This way the point of the best design is obtained as the one that has the lowest energy consumption. The investigation is now directed toward design and proof manipulators with the following features:

1. Use of the least amount of energy for a given task.
2. Reconfiguration of the model in order to use the least amount of energy for different tasks.

The advantage of this radical method is that it leads quickly to a solution. For the example of the manipulator 3RRR you need merely 5 minutes with an Intel processor at 1.4 GHz. It is true, while the complexity of the system increases, it will be more difficult to find a solution, since it then will depend on the capacity of the optimization method used, as well as on the complexity of the model equations of proposed inverted kinematics, as it is the case with spatial manipulators.

Acknowledgement

This research work reported was made possible by Grant PAPIIT of DGEP, UNAM.

References

1. Alba, O.G., Pámanes, J.A., Wenger, P. (2007) Trajectory planning of a redundant parallel manipulator changing of working mode, in *12th IFToMM World Congress*, June 18–21.
2. Bonev, I.A., Gosselin, C.M. (2001) Singularity loci of planar parallel manipulators with revolute joints, in *2nd Workshop on Computational Kinematics*, May 20–22.
3. Bonev, I.A., Zlatanov, D., Gosselin, C.M. (2003) Singularity analysis of 3-DOF planar parallel mechanisms, *Transactions of the ASME Journal of Mechanical Design* **125**, 573–581.
4. Chablat, D., Wenger, P. (2006) Self motions of special 3RPR planar parallel robot, in *Advances in Robot Kinematics: Mechanisms and Motion*, J. Lenarčič, B. Roth (Eds.), pp. 221–228. Springer, Dordrecht.
5. Hirzinger, G., Sporer, N., Schedl, M., Butterfaß, J., Grebenstein, M. (2003) Torque-controlled lightweight arms and articulated hands: Do we reach technological limits now?, *The International Journal of Robotics Research* **23**(4-5), 331–340.
6. Khan, W.A., Krovi, V.N., Saha, S.K., Angeles, J. (2005) Recursive kinematics and inverse dynamics for a planar 3R parallel manipulator, *Journal of Dynamic Systems, Measurement, and Control, ASME* **127**, 529–536.
7. Lee, E., Mavrodis, C., Merlet, J.P. (2004) Five precision point synthesis of spatial RRR manipulators using interval analysis, *Transactions of the ASME Journal of Mechanical Design* **126**, 842–849.
8. Ma, O., Angeles, J. (1993) Optimum design of manipulators under dynamic isotropy conditions, in *IEEE International Conference on Robotics and Automation*, pp. 485–492.
9. Paredis, C.J.J., Koshla, P.K. (1993) Kinematic design of serial link manipulators from task specifications, *The International Journal of Robotics Research* **12**(3), 274–287.
10. Snyman, J.A., Berner, D.F. (1999) The design of a planar robotic manipulator from optimum performance of prescribed tasks, *Structural design* **18**, 95–106.

Synthesis and Analysis of a Constrained Spherical Parallel Manipulator

G.S. Soh[1] and J.M. McCarthy[2]

[1]*Robotics and Automation Laboratory, University of California, Irvine, CA 92697, USA, e-mail: gsoh@uci.edu*
[2]*Mechanical and Aerospace Engineering Department, University of California, Irvine, CA 92697, USA, e-mail: jmmccart@uci.edu*

Abstract. This paper presents a methodology for constraining a spherical parallel manipulator so that it guides a body through five task positions with one degree-of-freedom. A dimensional synthesis procedure is used to constrain links of a 3-RRR spherical parallel manipulator using two spherical RR chains to obtain a spherical ten-bar linkage. For our purposes, we begin with a spherical parallel manipulator that has a base and platform that are isosceles spherical triangles. Inverse kinematics analysis of the spherical manipulator provides a set of relative task positions that are used to formulate the synthesis equations for spherical RR chains. The primary challenge is the analysis of the resulting four loop spherical linkage in order to animate is movement.

Key words: kinematic synthesis, spherical parallel manipulator, spherical linkage synthesis, ten-bar spherical linkage.

1 Introduction

In this paper, we constrain a spherical parallel manipulator to obtain a ten-bar spherical linkage. This paper extends recent dimensional synthesis results for planar eight-bar linkages [7] to the design of a spherical linkage.

We begin with an arbitrarily specified 3-RRR spherical parallel manipulator and add two RR spherical chains to constrain its movement. Inverse kinematic analysis of this manipulator for five task positions of the end-effector yield five configurations for each link in the articulated system. This provides the data necessary to formulate dimensional synthesis equations for a spherical RR chain connecting any two links in the articulated system [6].

The result is a constrained spherical parallel manipulator that moves through five task positions with one degree-of-freedom.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 101–110.

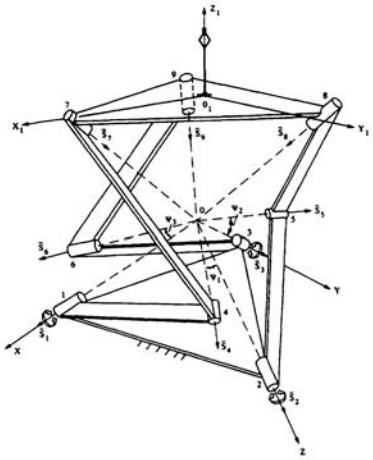


Fig. 1 The 3-RRR spherical parallel manipulator formed by an end-effector supported by three spherical 3R chains (figure used with the permission of R.I. Alizade).

2 Literature Review

This paper focuses constrains a 3-RRR spherical parallel manipulator, Figure 1, to obtain a desired movement. The forward and inverse kinematics of this manipulator was presented by Alizade et al. [1], and bulca et al. [3] analyzed its workspace.

Kong and Gosselin [5] classify the variety of structures available for spherical parallel manipulators, see also [4]. The focus of these papers are on ways that links and joints can be connected to obtain a "type" or topology for the system, and are examples of *type synthesis*. Our goal is to determine the dimensions of a given manipulator system, which is termed *dimensional synthesis*.

Once the spherical manipulator is constrained to a ten-bar linkage, it must be analyzed to determine its movement as a function of the input crank angle. We rely on the results of Wampler [9] to analyze this four loop spherical linkage.

3 Spherical Parallel Manipulators

A spherical parallel manipulator is a system of rigid bodies assembled so the end-effector moves about a fixed point in space. The 3-RRR spherical parallel manipulator is a three legged platform manipulator that is constructed so the joint axes of all of the revolute joints pass through this same fixed point. The result is that all of the links of the system undergo pure spatial rotation about this fixed point.

The kinematics equations of a spherical 3R robot equate the 3×3 rotation transformation $[D]$ between the end-effector and the base frame to the sequence of local coordinate transformations around the joint axes and along the links of the chain,

$$[D] = [G][Z(\theta_1)][X(\alpha_{12})][Z(\theta_2)][X(\alpha_{23})][Z(\theta_3)][H]. \quad (1)$$

The parameters θ_i define the movement at each joint and $\alpha_{i,j}$ define the angular length of the links. The transformation $[G]$ defines the position of the base of the chain relative to the fixed frame, and $[H]$ locates the task frame relative to the end-effector frame. The matrix $[D]$ defines the coordinate transformation from the world frame F to the task frame M.

4 The Spherical RR Constraint

A fundamental step of our spherical ten-bar synthesis methodology consists of sizing two spherical RR chains that constrains the three RRR spherical robot to one degree-of-freedom. We expand the RR synthesis equations, [6], to apply to this situation (see also [2, 8]).

Let $[B_{l,j}]$ be five position of the lth moving link, and $[B_{k,j}]$ be the five positions of the kth moving link measured in a world frame F, $j = 1, \ldots, 5$. Let $\mathbf{g}$ be the coordinates of the R-joint attached to the lth link measured in the link frame B_l. Similarly, let $\mathbf{w}$ be the coordinates of the other R-joint measured in the link frame B_k. The five positions of these points as the two moving bodies move between the task configurations are given by

$$\mathbf{G}^j = [B_{l,j}]\mathbf{g} \quad \text{and} \quad \mathbf{W}^j = [B_{k,j}]\mathbf{w} \quad (2)$$

Now, introduce the relative displacements

$$[R_{1j}] = [B_{l,j}][B_{l,1}]^{-1} \quad \text{and} \quad [S_{1j}] = [B_{k,j}][B_{k,1}]^{-1}, \quad (3)$$

so these equations become

$$\mathbf{G}^j = [R_{1j}]\mathbf{G}^1 \quad \text{and} \quad \mathbf{W}^j = [S_{1j}]\mathbf{W}^1 \quad (4)$$

where $[R_{11}] = [S_{11}] = [I]$ are the identity transformations.

The point $\mathbf{G}^j$ and $\mathbf{W}^j$ define the ends of a rigid angular link of length ρ, therefore we have the constraint equations

$$[R_{1j}]\mathbf{G}^1 \cdot [S_{1j}]\mathbf{W}^1 = \|\mathbf{G}^j\|\|\mathbf{W}^j\| \cos\rho \quad (5)$$

These five equations can be solved to determine the design parameters of the spherical RR constraint, $\mathbf{G}^1 = (u, v, w)$, $\mathbf{W}^1 = (x, y, z)$ and ρ. We will refer to these equations as the *synthesis equations* for the spherical RR link.

To solve the synthesis equations, it is convenient to introduce the displacements

$$[D_{1j}] = [R_{1j}]^T[S_{1j}], \quad (6)$$

so these equations become

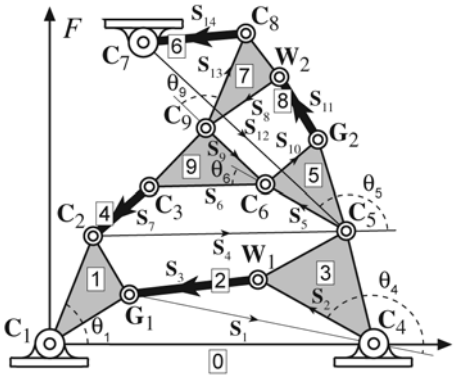


Fig. 2 This figure shows our conventions for the synthesis and analysis of the spherical parallel platform.

Table 1 Five task positions for the end-effector of the spherical platform in terms of longitude θ, latitude ϕ, and roll ψ.

Task	*Position* (θ, ϕ, ψ)
1	$(90°, -60°, 90°)$
2	$(77°, -35°, 82°)$
3	$(68°, -9°, 78°)$
4	$(65°, -1°, 75°)$
5	$(64°, 7°, 67°)$

$$\mathbf{G}^1 \cdot [D_{1j}]\mathbf{W}^1 = \|\mathbf{G}^j\|\|\mathbf{W}^j\| \cos\rho. \tag{7}$$

Subtract the first of these equations from the remaining to cancel the scalar terms $\|\mathbf{G}^j\|\|\mathbf{W}^j\| \cos\rho$, and the square terms in the variables u, v, w and x, y, z. The resulting four bilinear equations can be solved algebraically, or numerically using something equivalent to *Mathematica's* `Nsolve` function by setting $w = z = 1$ to obtain the desired pivots.

5 Synthesis of a Spherical Ten-Bar Linkage

To illustrate the proposed synthesis process, we attempt to constrain a spherical parallel manipulator to one degree of freedom by adding two spherical RR chains. Let the task positions in longitude, latitude and roll co-ordinates be as shown in Table 1. Select the angular links with dimensions such that $\alpha_{12} = \alpha_{23} = \alpha_{45} = \alpha_{56} = \alpha_{78} = \alpha_{89} = 60°$. Also, we assume that the transformation $[G]$ that defines the base of the chain relative to the fixed frame, and the transformation $[H]$ that locates the task frame relative to the end-effector frame for each of the 3R spherical chains $\mathbf{C}_1\mathbf{C}_2\mathbf{C}_3$, $\mathbf{C}_4\mathbf{C}_5\mathbf{C}_6$, and $\mathbf{C}_7\mathbf{C}_8\mathbf{C}_9$ are as defined in Table 2.

Table 2 The transformation $[G]$ and $[W]$ used to locate the base of the chain relative to the fixed frame and the task frame relative to the end-effector frame.

Spherical 3R Chain	$[G] = [Y(\theta)][X(-\phi)][Z(\psi)]$	$[H] = [Y(\theta)][X(-\phi)][Z(\psi)]$
$\mathbf{C}_1\mathbf{C}_2\mathbf{C}_3$	$(90°, -60°, 0°)$	$(-60°, 0°, 0°)$
$\mathbf{C}_4\mathbf{C}_5\mathbf{C}_6$	$(90°, 60°, 0°)$	$(0°, -60°, 0°)$
$\mathbf{C}_7\mathbf{C}_8\mathbf{C}_9$	$(90°, -180°, 0°)$	$(60°, 0°, 0°)$

Table 3 The spherical RR chain solutions obtained. The highlighted pivots were selected for the design.

S/N	$\mathbf{G}_1$	$\mathbf{W}_1$
1	$(-0.9456, -0.0093, 0.3250)$	$(-0.9138, -0.0213, 0.4057)$
2	$(-0.8793, -0.1323, 0.4575)$	$(-0.8989, -0.1716, 0.4032)$
3	$(\mathbf{-0.3041}, \mathbf{-0.6490}, \mathbf{0.6974})$	$(\mathbf{-0.7338}, \mathbf{-0.5137}, \mathbf{0.4445})$
4	$(0.5000, -0.8660, 0.0000)$	$(0.5000, 0.8660, 0.0000)$
5	$(0.35 + 1.27i, 3.44 - 0.62i, 0.56 + 3.05i)$	$(1.09 + 1.24i, 2.75 - 2.15i, 1.96 + 2.33i)$
6	$(0.35 - 1.27i, 3.44 + 0.62i, 0.56 - 3.05i)$	$(1.09 - 1.24i, 2.75 + 2.15i, 1.96 - 2.33i)$

S/N	$\mathbf{G}_2$	$\mathbf{W}_2$
1	$(0.54-29.18i, 20.60-2.00i, 20.96+2.72i)$	$(9.63-26.69i, 19.50+4.76i, 18.49+8.88i)$
2	$(0.54+29.18i, 20.60+2.00i, 20.96-2.72i)$	$(9.63+26.69i, 19.50-4.76i, 18.49-8.88i)$
3	$(0.11-0.01i, -0.71-0.01i, 0.70-0.00i)$	$(0.11-0.01i, -0.71-0.00i, 0.70-0.00i)$
4	$(0.11+0.01i, -0.71+0.01i, 0.70+0.00i)$	$(0.11+0.01i, -0.71+0.00i, 0.70+0.00i)$
5	$(\mathbf{0.1964}, \mathbf{-0.6483}, \mathbf{0.7356})$	$(\mathbf{0.2562}, \mathbf{-0.6482}, \mathbf{0.7170})$
6	$(-0.2500, -0.4330, 0.8660)$	$(0.5000, 0.8660, 0.0000)$

Once the various spherical platform dimensions are identified, the positions of its links for the various 3R chains, B_1, B_4, B_3, B_5, B_6, and B_7 can be determined by solving the inverse kinematics of the various spherical 3R chains. Therefore we can identify five positions $T_i^{B_1}$, and $T_i^{B_3}$, $i = 1, \ldots, 5$ for the design of a spherical RR chain denoted $\mathbf{G}_1\mathbf{W}_1$ in Figure 2. We compute the displacements $[D_{1j}] = ([B_{1,j}][B_{1,1}]^{-1})^T[B_{4,j}][B_{4,1}]^{-1}$, $i = 1, \ldots, 5$, to synthesize the spherical RR chain $\mathbf{G}_1\mathbf{W}_1$ using Eq. (7). Similarly, we identify five positions $T_i^{B_5}$, and $T_i^{B_7}$, $i = 1, \ldots, 5$ for the design of a spherical RR chain $\mathbf{G}_2\mathbf{W}_2$.

The five task positions listed in Table 1 yield three design candidate to constrain the spherical parallel manipulator, Table 3. Figure 3 shows the chosen design, and Figure 4 shows the constrained spherical parallel manipulator passing through each of the specified task positions.

6 Analysis of the Spherical Ten-Bar Linkage

The analysis of the spherical ten-bar linkage as shown in Figure 2 is equivalent to the displacement analysis of a single loop spherical triangle $\mathbf{G}_1\mathbf{C}_4\mathbf{W}_1$, and a three

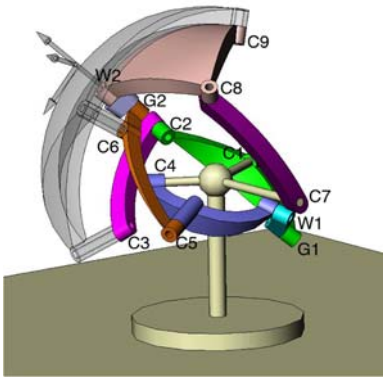


Fig. 3 The resulting constrained 3-RRR spherical parallel manipulator, which is a spherical ten-bar linkage.

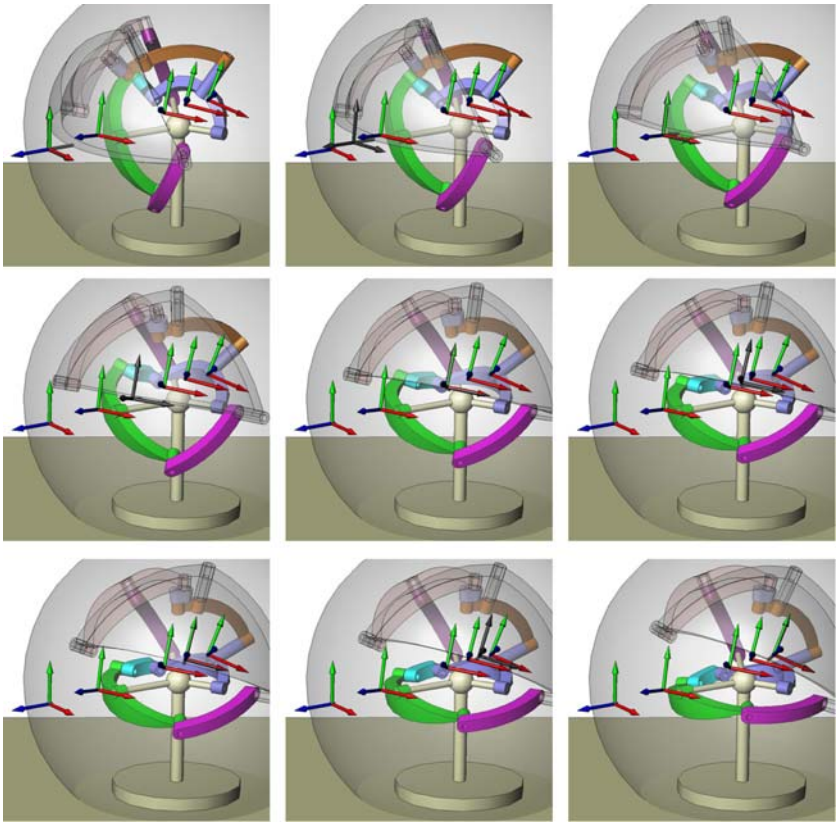


Fig. 4 A sequence of images showing the movement of the spherical ten-bar linkage reaching each of the five task positions.

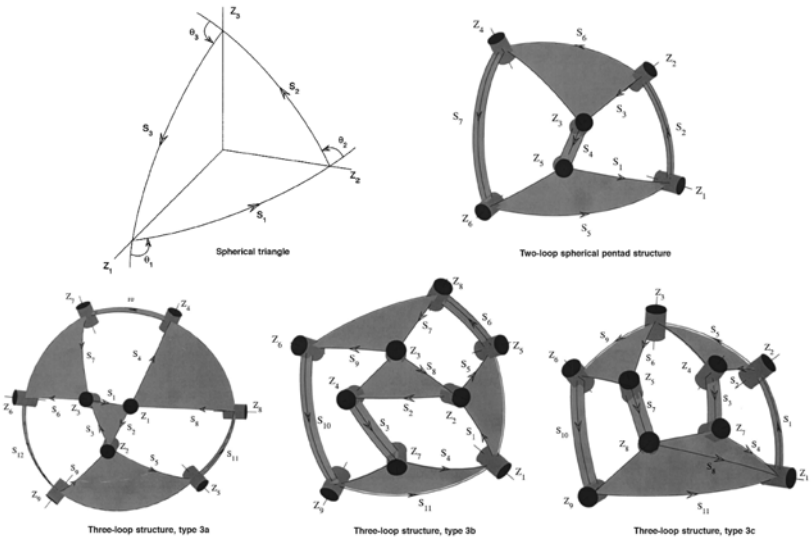


Fig. 5 The various basic spherical structures having three or fewer loops. In the first row, we have the single-loop spherical triangle, and the two-loop pentad structure. The second row consists of the three-loop type 3a, type 3b, and type 3c structures (figure used with the permission of C.W. Wampler).

loop spherical type 3b structure $\mathbf{C}_2\mathbf{C}_3\mathbf{C}_9\mathbf{C}_6\mathbf{C}_5\mathbf{G}_2\ \mathbf{W}_2\mathbf{C}_8\mathbf{C}_7$. See Figure 5 for the various spherical structures having three or fewer loops.

To see how we can decompose this ten-bar spherical linkage into a spherical triangle and type 3b structure, we label the links on the spherical linkage as $0, 1, \ldots, 9$ as shown in Figure 2. Given an input angle θ_1, we can merge links 0 and 1 into a composite rigid link. We use (01) to represent this composite link. Now, (01), 2, 3 forms a spherical triangle. We solve it to obtain two solutions. For each solution, we merge them into one big composite link (0123). Now (0123) and 4, 5, 6, 7, 8, 9 forms a spherical type 3b structure. We solve it to obtain 24 solutions, hence a total of 48 solutions for this ten-bar spherical topology.

6.1 Analyzing the Spherical Triangle

The loop equation for the spherical triangle $\mathbf{G}_1\mathbf{C}_4\mathbf{W}_1$ may be written as

$$Z_{G_1} S_1 Z_4 S_2 Z_{W_1} S_3 = I \tag{8}$$

where Z_i is a joint rotation and S_i is a side rotation and I is the identity rotation. Z_i is a rotation about the z-axis, and may be written in terms of a rotation angle θ_i as

$$Z_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 \\ \sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{9}$$

Let $\mathbf{z} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$, $\cos\theta_i = (1 - t_i^2)/(1 + t_i^2)$ and $\sin\theta_i = -2t_i/(1 + t_i^2)$. If we pre and post multiply $\mathbf{z}^T$ and $S_3^T\mathbf{z}$ respectively, we can simplify Eq. (8) into

$$\mathbf{z}^T S_1 \hat{Z}_4 S_2 \mathbf{z} = \mathbf{z}^T S_3^T \mathbf{z}(1 + t_4^2). \tag{10}$$

where $S_1 = R_y(2.37)$, $S_2 = R_y(2.52)$, and $S_3 = R_y(0.52)$ for $\theta_1 = 0.34$ rad, and

$$\hat{Z}_i = \begin{bmatrix} 1 - t_i^2 & -2t_i & 0 \\ 2t_i & 1 - t_i^2 & 0 \\ 0 & 0 & 1 + t_i^2 \end{bmatrix}. \tag{11}$$

We solve this quadratic equation for t_4.

6.2 Analyzing the Spherical 3b Structure

The loop equations for the three-loop spherical structure 3b $\mathbf{C}_2\mathbf{C}_3\mathbf{C}_9\mathbf{C}_6\mathbf{C}_5\mathbf{G}_2\mathbf{W}_2\mathbf{C}_8\mathbf{C}_7$ can be written as

$$\begin{aligned} &Z_2 S_4 Z_5 S_5 Z_6 S_6 Z_3 S_7 = I, \\ &Z_{W_2} S_8' Z_9 S_9' Z_6 S_{10} Z_{G_2} S_{11} = I, \\ &Z_7 S_{12}' Z_5 S_5' Z_6 S_9 Z_9 S_{13} Z_8 S_{14} = I, \end{aligned}$$

Using the simplification process as above, one obtains the following three equations in t_5, t_6, and t_9:

$$\begin{aligned} f_1 &: \mathbf{z}^T[S_4 \hat{Z}_5 S_5 \hat{Z}_6 S_6 - S_7^T(1 + t_5^2)(1 + t_6^2)]\mathbf{z} = 0, \\ f_2 &: \mathbf{z}^T[S_8' \hat{Z}_9 S_9' \hat{Z}_6 S_{10} - S_{11}^T(1 + t_6^2)(1 + t_9^2)]\mathbf{z} = 0, \\ f_3 &: \mathbf{z}^T[S_{12}' \hat{Z}_5 S_5' \hat{Z}_6 S_9 \hat{Z}_9 S_{13} - S_{14}^T(1 + t_5^2)(1 + t_6^2)(1 + t_9^2)]\mathbf{z} = 0 \end{aligned}$$

where $S_4 = R_y(2.52)$, $S_5 = R_y(1.05)$, $S_5' = R_y(1.05)R_z(-2.21)$, $S_6 = R_y(1.32)$, $S_7 = R_y(1.05)$, $S_8' = R_y(0.77)R_z(-0.99)$, $S_9 = R_y(1.32)$, $S_9' = R_y(1.32)R_z(1.20)$, $S_{10} = R_y(0.26)$, $S_{11} = R_y(0.06)$, $S_{12}' = R_y(1.62)R_z(-2.78)$, $S_{13} = R_y(1.05)$ and $S_{14} = R_y(1.05)$ for $\theta_1 = 0.34$ rad.

6.3 Elimination Procedure

To solve this system, first augment f_1 with $\{1, t_9, t_9^2, t_9^3\} \otimes \{1, t_5\} \otimes \{1, t_6\}$, f_2 with $\{1, t_5, t_5^2, t_5^3\} \otimes \{1, t_6\} \otimes \{1, t_9\}$, and f_3 with $\{1, t_5\} \otimes \{1, t_6\} \otimes \{1, t_9\}$ to get 40 equations in the 64 monomials $\mathbf{m} = \{1, t_5, t_5^2, t_5^3\} \otimes \{1, t_6, t_6^2, t_6^3\} \otimes \{1, t_9, t_9^2, t_9^3\}$. The excess of monomials over the equations is $64 - 40 = 24$, so we must append 24 identities to get a system of 64 polynomials in 64 monomials. The 24 identities are formulated such that they satisfy $t_9\mathbf{m}_1 - \mathbf{m}_2 = 0$, where $\mathbf{m}_1$ is the list of monomials in the set $[(\{1, t_5, t_5^2, t_5^3\} \otimes \{1, t_2\}) \bigcup \{t_6^2, t_5t_6^2, t_6^3, t_5t_6^3\}] \otimes \{1, t_9\}$, and $\mathbf{m}_2 = t_9 \otimes \mathbf{m}_1$.

Writing the equations in block matrix form and letting $t_9 = x$ we get

$$[\hat{K}(x)]\mathbf{m} = \begin{bmatrix} K\mathbf{m} \\ x\mathbf{m}_1 - \mathbf{m}_2 \end{bmatrix} = \begin{bmatrix} K_1 & K_2 & K_3 & K_4 \\ I_1x + C_1 & C_2 & 0 & 0 \\ 0 & I_2x & -I_2 & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \end{Bmatrix} = 0, \tag{14}$$

where $\mathbf{y}_1 = \mathbf{m}_1 \backslash \mathbf{y}_2, \mathbf{y}_2 = \mathbf{y}_3/x, \mathbf{y}_3 = \mathbf{m}_2 \backslash \mathbf{m}_1, \mathbf{y}_4 = \mathbf{m} \backslash (\mathbf{m}_1 \bigcup \mathbf{m}_2)$, with $A \backslash B$ defined as the set minus $A - (A \bigcap B)$, are the four partition sets of the monomials $\mathbf{m}$. I_1,I_2 are identity matrices and C_2 are sparse, having a single entry of -1 in each row of the pair. Now we reduce K_4 to upper triangular form, and premultiply by a matrix that annihilates $\mathbf{y}_3, \mathbf{y}_4$ to obtain,

$$\begin{bmatrix} 0 & 0 & I_1 & 0 \\ 0 & I_2 & 0 & \tilde{K}_{23} \end{bmatrix} \begin{bmatrix} \tilde{K}_{11} & \tilde{K}_{12} & \tilde{K}_{13} & U \\ \tilde{K}_{21} & \tilde{K}_{22} & \tilde{K}_{23} & 0 \\ I_1x + C_1 & C_2 & 0 & 0 \\ 0 & I_2x & -I_2 & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \end{Bmatrix} = 0. \tag{15}$$

Multiplying this out and dropping the trailing trivial columns we manage to reduce the system of 64 polynomial equations to a 24×24 generalize eigenvalue problem,

$$\begin{bmatrix} I_1x + C_1 & C_2 \\ \tilde{K}_{21} & \tilde{K}_{22} + \tilde{K}_{23}x \end{bmatrix} \begin{Bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{Bmatrix} = 0. \tag{16}$$

Each value of $x = t_9$ has an associated eigenvector $(\mathbf{y}_1, \mathbf{y}_2)^T$. The eigenvectors are up to scale corresponding to the monomial 1 in $\mathbf{m}_1$. Divide out the scale factor to retrieve the values of the remaining joint angles t_5, t_6.

7 Comments

Our design process yields a spherical ten-bar linkage that guides a rigid body through five arbitrary spatial orientations, however, it is known that a similar design process exists to obtain a spherical four-bar linkage that achieves the same task. While it seems obvious that fewer moving parts are preferred, we have found that it is easier to obtain effective designs for six, eight and now ten-bar linkages.

In our experience, the fewer number of dimensional parameters in a four-bar linkage limits the designer's options when flaws arise in the synthesis process, such as branch and order defects and partial crank rotatability. On the other hand, the increasing numbers of parameters in six, eight and ten-bar linkages provide increasing numbers of alternative configurations and dimensional options that make it relatively easy to find designs that avoid these flaws.

Furthermore, though there is almost no research in this area, it seems reasonable to expect that optimization techniques can use these extra dimensional parameters to obtain effective designs with constraints on fixed pivot locations, link lengths and mechanical advantage in order to enhance performance.

8 Conclusions

This paper presents a strategy to constrain a 3-RRR spherical parallel manipulator with two spherical RR chains to obtain a spherical ten-bar linkage. The resulting device performs a one degree-of-freedom movement through five arbitrary task positions. The analysis of this resulting four loop spherical linkage is presented using Wampler's method. An example design and analysis is presented.

Acknowledgement

The support of the National Science Foundation grant 0456213 is gratefully acknowledged.

References

1. Alizade, R.I., Tagiyev, N.R. and Duffy, J., A forward and reverse displacement analysis of an in-parallel spherical manipulator. *Mechanism and Machine Theory* **29**(1), 125–137 (1994).
2. Alizade, R.I. and Kilit, O., Analytical synthesis of function generating four-bar mechanism for five precision points. *Mechanism and Machine Theory* **40**(7), 863–878 (2005).
3. Bulca, F., Angeles, J. and Zsomber-Murray, P.J., On the workspace determination of spherical serial and platform mechanisms. *Mechanism and Machine Theory* **34**(3), 497–512 (1999).
4. Gallardo, J., Rodriguez, R., Caudillo, M. and Rico, J.M., A family of spherical parallel manipulators with two legs. *Mechanism and Machine Theory* **43**(2), 201–216 (2007).
5. Kong, X. and Gosselin, C.M., Type synthesis of 3-dof spherical parallel manipulators based on screw theory. *ASME Journal of Mechanical Design* **126**(1), 101–108 (2004).
6. McCarthy, J.M., *Geometric Design of Linkages*. Springer-Verlag, New York (2000).
7. Soh, G.S. and McCarthy, J.M., Synthesis of mechanically constrained planar 2-RRR planar parallel robots. In *Proceedings 12th IFToMM World Congress*, Besancon, France (2007).
8. Suh, C.H. and Radcliffe, C.W., *Kinematics and Mechanisms Design*. John Wiley and Sons, New York (1978).
9. Wampler, C.W., Displacement analysis of spherical mechanisms having three or fewer loops. *ASME Journal of Mechanical Design* **126**(1), 93–100 (2004).

A Nonholonomic 3-DOF Parallel Robot

Patricia Ben-Horin[1] and Federico Thomas[2]

[1]*Technion – Israel Institute of Technology, Technion City, Haifa 32000, Israel, e-mail: patbh@tx.technion.ac.il*
[2]*Institut de Robòtica i Informàtica Industrial (CSIC-UPC), 08028 Barcelona, Spain, e-mail: fthomas@iri.upc.edu*

Abstract. This paper addresses the problem of devising a parallel robot that can achieve a general position and orientation for its moving platform using three motors only. At first glance, this might seem impossible but, by using nonholonomic joints, three motors can manoeuver to approximate any six-degree-of-freedom motion for the platform. We analyze the case of a 3-UPU parallel robot in which each universal joint attached to the base is substituted by a spherical joint whose motion is constrained by a disk that can freely roll in contact with the sphere without slipping. The proposed robot involves fewer links and actuators than standard six-degree-of-freedom robots offering a large workspace free from leg interferences.

Key words: parallel robot, nonholonomic joint, Lie bracket, underactuated robot.

1 Introduction

Among all possible spatial three degree-of-freedom parallel robots, those based on three UPU legs [15] have drawn the attention of many researchers because of its fascinating properties. It consists of a fixed base and a moving platform connected by three serial chains, with each chain having a universal-prismatic-universal joint arranged in sequence as shown in Figure 1 (left). The universal joints are passive, only the three prismatic joints are actuated. The platform has three degrees of freedom (DOF) with respect to the base that, in general, involve coupled translations and rotations but, by properly arranging the universal joints, the moving platform can undergo pure translational [3, 13] or spherical [4, 8] motions.

In this paper, each universal joint attached to the base is substituted by a passive joint based on a sphere whose motion is constrained, as shown in Figure 1 (right), by a disk that can freely roll in contact with the sphere without slipping. This no-slip constraint is a *nonholonomic constraint*, a constraint in the velocity. The kinematics of this element is equivalent to that of a unicycle on a sphere [5]. Since the disk can only roll without slipping laterally with respect to the sphere, the sphere can instantaneously only spin about $\mathbf{x}_i$ and $\mathbf{y}_i$ with respect to the base. Thus, from the instantaneous kinematics point of view, this joint is equivalent to a universal joint.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 111–118.

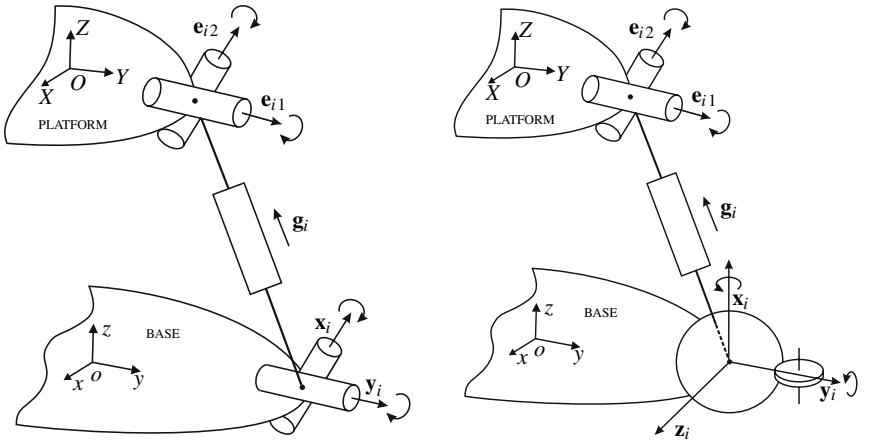


Fig. 1 A UPU leg (left); and the result of substituting the universal joint attached to the base by the proposed nonholonomic joint (right).

The only difference being that $\mathbf{y}_i$ remains fixed with respect to the base, independently of the orientation of the leg. Hence, the instantaneous kinematics and statics of the resulting mechanism can be carried out using the same techniques applied to the standard 3-UPU platform [7].

There are many examples of nonholonomic systems that involve wheels rolling in contact without slipping on planes or spheres. These kind of mechanical systems have received attention because they can access a configuration space with a higher dimension than the number of its actuators. However, due to the nonintegrability of the velocity constraints, nonholonomic systems cannot follow an arbitrary path in the configuration space. Therefore, it is an important problem to find a feasible trajectory between any two configurations that satisfies the nonholonomic constraints [10]. In this paper, we concentrate ourselves on the rather simpler problem of deciding if the proposed robot can manoeuvre in any direction from a given configuration. This is achieved by using the concept of Lie bracket. Given two vector fields, the Lie bracket tell us if infinitesimal motions along these vector fields can be used to locally generate a motion in a direction not contained in the subspace linearly spanned by them.

To the best of our knowledge, the use of nonholonomic mechanical elements as joints in Robotics has received little attention in the past. Two related previous works are the two-motor actuated robot wrist described in [14], and the continuous variable transmission described in [11].

This paper is structured as follows. Section 2 deals with the statics and the instantaneous kinematics of the proposed robot. Section 3 discuses the mathematical tools needed to decide if the proposed robot can manoeuvre in any direction from an arbitrary configuration. Section 4 summarizes all possible singularities for the proposed design and, finally, Section 5 offers the conclusions.

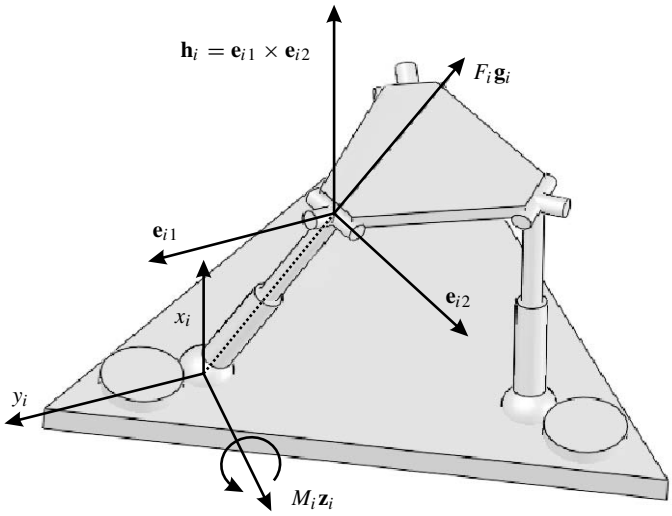


Fig. 2 The proposed robot and the relevant vectors associated with its ith leg, $i = 1, 2, 3$, used in the the presented statics analysis.

2 Kinetostatics of the Proposed Robot

In this section, we analyze the statics of a single leg, then the statics of the whole robot and, finally, its instantaneous kinematics.

Figure 2 shows the proposed robot. In our design the universal joints attached to the platform are arranged so that the unit vectors $\mathbf{e}_{i2}$ and $\mathbf{e}_{i1}$ can be expressed as:

$$\mathbf{e}_{i2} = \mathbf{R}(\mathbf{n} \times \mathbf{r}_i), \quad \mathbf{e}_{i1} = \mathbf{e}_{i2} \times \mathbf{g}_i,$$

where $\mathbf{n}$ is the normal to the platform, $\mathbf{R}$ is the rotation matrix from the platform to the world reference frame, and $\mathbf{r}_i$ is the unit vector from the platform reference frame to the universal joint center. As a consequence, the vector orthogonal to both vectors can be expressed as:

$$\mathbf{h}_i = \mathbf{e}_{i1} \times \mathbf{e}_{i2} = ((\mathbf{R}(\mathbf{n} \times \mathbf{r}_i)) \times \mathbf{g}_i) \times \mathbf{R}(\mathbf{n} \times \mathbf{r}_i).$$

As explained in the previous section, the nonholonomic joint attached to the base is constrained to only rotate instantaneously about axes $\mathbf{x}_i$ and $\mathbf{y}_i$, thus a torque between the base and the leg can only be transmitted about axis $\mathbf{z}_i = \mathbf{x}_i \times \mathbf{y}_i$. Likewise, a torque between the leg and the platform can only be transmitted about axis $\mathbf{h}_i = \mathbf{e}_{i1} \times \mathbf{e}_{i2}$. Then, the torque $M_i\mathbf{z}_i$ on the base is transmitted through the leg to the platform by projecting it on axis $\mathbf{h}_i$. In other words, the transmitted torque is $(M_i\mathbf{z}_i \cdot \mathbf{h}_i)\mathbf{h}_i$. Moreover, a force directed along the leg, say $F_i\mathbf{g}_i$, and acting at the universal joint center, generates the torque $\mathbf{R}\mathbf{r}_i \times F_i\mathbf{g}_i$ on the platform.

Now, let $\mathbf{F}_e$ and $\mathbf{M}_e$ be a external force and a external torque, respectively, acting on the platform. Then, the static equilibrium equations of the platform can be written as:

$$\mathbf{F}_e = \sum_{i=1}^{3} F_i \mathbf{g}_i$$

$$\mathbf{M}_e = \sum_{i=1}^{3} (M_i \mathbf{z}_i \cdot \mathbf{h}_i)\mathbf{h}_i + \sum_{i=1}^{3} \mathbf{Rr}_i \times F_i \mathbf{g}_i$$

which can be rewritten in matrix form as:

$$\begin{pmatrix} \mathbf{g}_1 & \mathbf{g}_2 & \mathbf{g}_3 & \cdots \\ \mathbf{Rr}_1 \times \mathbf{g}_1 & \mathbf{Rr}_2 \times \mathbf{g}_2 & \mathbf{Rr}_3 \times \mathbf{g}_3 & \cdots \\ \mathbf{0}_{3\times 1} & \mathbf{0}_{3\times 1} & \mathbf{0}_{3\times 1} & \cdots \\ (\mathbf{z}_1 \cdot \mathbf{h}_1)\mathbf{h}_1 & (\mathbf{z}_2 \cdot \mathbf{h}_2)\mathbf{h}_2 & (\mathbf{z}_3 \cdot \mathbf{h}_3)\mathbf{h}_3 & \cdots \end{pmatrix} \begin{pmatrix} F_1 \\ F_2 \\ F_3 \\ M_1 \\ M_2 \\ M_3 \end{pmatrix} = \begin{pmatrix} \mathbf{F}_e \\ \mathbf{M}_e \end{pmatrix} \tag{1}$$

where the right-hand side of the above equation is known as the *wrench* applied on the platform.

Finally, to obtain the linear mapping from the velocities in the prismatic actuators $(\dot{q}_1, \dot{q}_2, \dot{q}_3)$ to the platform linear and angular velocities $(\dot{x}, \dot{y}, \dot{z}, \dot{\theta}_x, \dot{\theta}_y, \dot{\theta}_z)$, that is, the platform *twist*, one can rely on the fact that the twist and the wrench are reciprocal to each other [1]. This permits to conclude that

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{pmatrix} = \begin{pmatrix} \mathbf{g}_1^T & (\mathbf{Rr}_1 \times \mathbf{g}_1)^T \\ \mathbf{g}_1^T & (\mathbf{Rr}_1 \times \mathbf{g}_1)^T \\ \mathbf{g}_1^T & (\mathbf{Rr}_1 \times \mathbf{g}_1)^T \\ \mathbf{0}_{1\times 3} & (\mathbf{z}_1 \cdot \mathbf{h}_1)\mathbf{h}_1^T \\ \mathbf{0}_{1\times 3} & (\mathbf{z}_2 \cdot \mathbf{h}_2)\mathbf{h}_2^T \\ \mathbf{0}_{1\times 3} & (\mathbf{z}_3 \cdot \mathbf{h}_3)\mathbf{h}_3^T \end{pmatrix}^{-1} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 \end{pmatrix} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix}. \tag{2}$$

Note that the components of vectors $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$ are not derivatives of any generalized coordinates of the moving platform as there is no representation of the platform orientation whose derivatives correspond to angular velocities [9]. Nevertheless, since

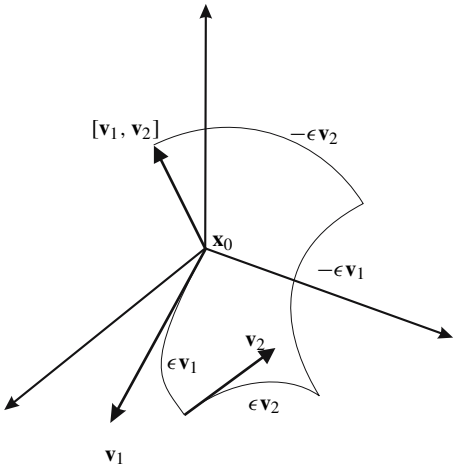


Fig. 3 Geometric interpretation of the Lie bracket of the vector fields $\mathbf{v}_1$ and $\mathbf{v}_2$.

$$\mathbf{v}_i(0) = \left.\frac{\partial \mathbf{x}}{\partial q_i}\right|_{\mathbf{x}=0}, \quad \text{with } \mathbf{x} = (x, y, z, \theta_x, \theta_y, \theta_z),$$

when analyzing the behavior of the proposed robot at a given configuration, say $\mathbf{x}_0$, we can always change the world reference frame so that $\mathbf{x}_0 = \mathbf{0}$. Under this circumstance, the matrix $(\mathbf{v}_1\mathbf{v}_2\mathbf{v}_3)$ can be seen as an analytic Jacobian.

3 Locally Reachable Configurations

The configuration of the platform can be modelled as a point in $\mathbb{R}^3 \times SO(3)$ which is locally diffeomorphic to $\mathbb{R}^6$, thus, equipped with a proper set of local coordinates, we are going to treat the configuration space of the platform locally as $\mathbb{R}^6$.

As described in the previous section, the instantaneous motion of the proposed robot can be described in terms of three vector fields – $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$ – associated with the three actuated prismatic joints. Linear combinations of these vector fields define a linear subspace in $\mathbb{R}^6$, denoted by Span($\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$), describing all possible instantaneous motions of the platform. We would like to know the reachable set of configurations following these vector fields. While this is generally difficult globally, it is possible to learn something about the reachable set locally [2].

For two of our vector fields, say $\mathbf{v}_1$ and $\mathbf{v}_2$, consider the state reached from $\mathbf{x}_0$ by first following $\mathbf{v}_1$ for a small time ϵ, then following $\mathbf{v}_2$ for time ϵ, then following $-\mathbf{v}_1$ for time ϵ, then following $-\mathbf{v}_2$ for time ϵ. Then, it can be shown that see [12, p. 323]:

$$\lim_{\epsilon \to 0} \mathbf{x}(4\epsilon) = \mathbf{x}_0 + \epsilon^2 \left(\frac{\partial \mathbf{v}_2}{\partial \mathbf{x}} \mathbf{v}_1(\mathbf{x}_0) - \frac{\partial \mathbf{v}_1}{\partial \mathbf{x}} \mathbf{v}_2(\mathbf{x}_0) \right), \tag{3}$$

where the partial derivatives are evaluated at $\mathbf{x}_0$. The term multiplying ϵ^2 represents the net motion of the system which is technically known as the *Lie bracket* of $\mathbf{v}_1$ and $\mathbf{v}_2$ and denoted by $[\mathbf{v}_1, \mathbf{v}_2]$.

The Lie bracket $[\mathbf{v}_1, \mathbf{v}_2]$ defines a new vector field, and if it is not contained in $\text{Span}(\mathbf{v}_1, \mathbf{v}_2)$, then it represents a new motion direction that can be followed. Locally generating motion in this direction is slower than following the vector field $\mathbf{v}_1$ and $\mathbf{v}_2$ directly, as the net motion is only $O(\epsilon^3)$ for time $O(\epsilon)$.

Since $[\mathbf{v}_1, \mathbf{v}_2]$ is a vector field, we can calculate its Lie bracket with another vector field, say $\mathbf{v}_3$. Since the resulting net motion would be $O(\epsilon^9)$ for time $O(\epsilon)$, these motions are not considered here for practical reasons. Hence, we limit our analysis to Lie products of degree 1, that is, $[\mathbf{v}_1, \mathbf{v}_2]$, $[\mathbf{v}_2, \mathbf{v}_3]$, and $[\mathbf{v}_3, \mathbf{v}_1]$. Then, if the dimension of the space $\text{Span}(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, [\mathbf{v}_1, \mathbf{v}_2], [\mathbf{v}_2, \mathbf{v}_3], [\mathbf{v}_3, \mathbf{v}_1])$ is six at a given configuration $\mathbf{x}_0$, the robot can locally manoeuver in any direction.

4 Singularities

Note that vector fields $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$ are undefined if the matrix in Eq. (2) cannot be inverted, i.e., if it is rank deficient. The configurations where this happens correspond to *singularities* of the robot. We can distinguish the following three cases:

(a) $\mathbf{g}_1$, $\mathbf{g}_2$ and $\mathbf{g}_3$ lie on a plane, or they are parallel to each other.
(b) $\mathbf{h}_1$, $\mathbf{h}_2$ and $\mathbf{h}_3$ lie on a plane, or any two of them are parallel.
(c) $\mathbf{h}_i \perp \mathbf{z}_i$, for $i = 1, 2, 3$. In this case, no torque can be transmitted from the base to the platform, so that the platform gains an instantaneous degree of freedom. This corresponds to an *uncertain singularity*, i.e. a singularity with increased instantaneous mobility.

While the first case corresponds to a usual parallel-chain singularity, the other two correspond to singularities classified in [17] as *constraint singularities*. These singularities are not detected when directly computing the input-output velocity equations without performing a complete statics analysis, as it has been done here.

Assuming that $\mathbf{v}_1$, $\mathbf{v}_2$, and $\mathbf{v}_3$ are well-defined, i.e., the robot is in a configuration where none of the above conditions is satisfied, *nonholonomic singularities* have also to be considered in our case. They are defined as those configurations in which the robot cannot manoeuvre in any direction of the configuration space. They can be characterized algebraically as those configurations satisfying

$$\left| \mathbf{v}_1(\mathbf{x})\ \mathbf{v}_2(\mathbf{x})\ \mathbf{v}_3(\mathbf{x})\ [\mathbf{v}_1(\mathbf{x}), \mathbf{v}_2(\mathbf{x})]\ [\mathbf{v}_2(\mathbf{x}), \mathbf{v}_3(\mathbf{x})]\ [\mathbf{v}_3(\mathbf{x}), \mathbf{v}_1(\mathbf{x})] \right| = 0.$$

The explicit computation of the above expression requires a computer algebra system. It remains to be seen how these singularities are distributed in the working space of the robot.

5 Conclusions

A car-like robot cannot move sideways but, by manoeuvring, it can reach any place and orientation on a plane. In this paper, by introducing nonholonomic joints, we have extended this behavior to parallel robots. This permits reducing the number of actuators, and hence the complexity of the robot, at the cost of motion speed.

We have analyzed the case of a 3-UPU parallel robot in which each universal joint attached to the base has been substituted by a nonholonomic joint whose instantaneous kinematics is equivalent to that of a universal joint. Many other architectures can be devised based on this substitution.

Acknowledgements

This work has been partially supported by the Spanish Ministry of Education and Science through contract DPI2007-60858, and by the Research Network on Advanced Production Techniques of the Autonomous Government of Catalonia (Xartap).

References

1. Bruyninckx, H. and De Schutter, J. (1998), Unified Kinetostatics for serial, parallel and mobile robots, in *Proceedings of ARK'98*, Strobl, Austria.
2. Choset, H., Lynch, K.M., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L.E. and Thrun, S. (2005), *Principles of Robot Motion: Theory, Algorithms, and Implementations*, MIT Press, Boston.
3. Di Gregorio, R. and Parenti-Castelli, V. (2002), Mobility analysis of the 3-UPU parallel mechanism assembled for pure translational motion, *ASME Journal of Mechanical Design* **124**, 259–264.
4. Di Gregorio, R. (2004), Statics and singularity loci of the 3-UPU wrist, *IEEE Trans. on Robotics* **30**(4), 630–635.
5. Hennessey, M.P. (2006), Visualizing the motion of a unicycle on a sphere, *Int. J. of Modelling and Simulation* **26**(1), 69–79.
6. Huang, Z., Li, S.H. and Zuo, R.G. (2004), Feasible instantaneous motions and kinematic characteristics of a special 3-DOF 3-UPU parallel manipulator, *Mechanism and Machine Theory* **39**, 957–970.
7. Joshi, S.A. and Tsai, L-W. (2002), Jacobian analysis of limited-DOF parallel manipulators, *Trans. of the ASME* **124**, 254–258.
8. Karouia, M. and Hervé, J.M. (2000), A three-dof tripod for generating spherical rotation, in *Proceedings of ARK'00*, Portorož, Slovenia.
9. Kim, D., Chung, W. and Youm, Y. (2000), Analytic Jacobian of in-parallel manipulators, in *Proceedings of IEEE ICRA'00*, San Fancisco, USA.
10. Li, Z. and Canny, J.F. (Eds.) (1993), *Nonholonomic Motion Planning*, Kluwer Academic Publishers, Boston.
11. Moore, C.A. (1997), Continuously variable transmission for serial link cobot architectures, Master Thesis, Northwestern University.

12. Murray, R.M., Li, Z. and Sastry, S.S. (1994), *A Mathematical Introduction to Robotic Manipulation*, CRC Press, Boca Raton.
13. Parenti-Castelli, V., Di Gregorio, R. and Rubani, F. (2000), Workspace and optimal design of a pure translation parallel manipulator, *Meccanica* **35**, 203–214.
14. Stammers, C.W. (1993), Operation of a two-motor robot wrist to achive three-dimensional manoeuvres with minimum total rotation, *Proc. Inst. Mech. Eng., C* **207**(1), 33–39.
15. Tsai, L-W. (1996), Kinematics of a three-DOF platform with three extensible limbs, in *Proceedings of ARK'96*, Portorož, Slovenia.
16. Tsai, L-W. and Joshi, S.A. (2000), Kinematics and optimization of a spatial 3-UPU parallel manipulator, *ASME Journal of Mechanical Design* **122**, 439–446.
17. Zlatanov, D., Bonev, I.A. and Gosselin, C.M. (2002), Constraint singularities of parallel mechanisms, in *Proceedings of the IEEE ICRA*, Washington, DC, USA.

4-DOF Parallel Architecture for Laparoscopic Surgery

Mohammad Aamir Khan, Matteo Zoppi and Rezia Molfino

DIMEC, University of Genova, Via Opera Pia 15A, 16145 Genova, Italy,
e-mail: {aamirkhan, zoppi, molfino}@dimec.unige.it

Abstract. This paper analyzes a new laparoscopic surgical setup with two surgical robots. Requirements and constraints are outlined and the architecture selected for the surgical robots is presented and discussed. Analytical models of the inverse and forward kinematics are provided together with a detailed analysis of mobility and constraints. The velocity kinematics is addressed and singularity analysis outlined.

Key words: parallel mechanisms, robotic surgery, minimally invasive surgery.

1 Introduction and Surgical Requirements

In minimally invasive surgery (MIS) surgeon manipulates tissues using tools inserted through small incisions. It has become standard for various surgical and diagnostic procedures owing to minimal trauma caused to healthy tissues resulting in less post-operative pain and shorter hospitalization. Drawbacks occurs are in the form of drastically reduced haptic feedback, motion reversal effect and hand tremor. Use of telerobotics restores the tactile cues and intuitive dexterity of the surgeon and improves accuracy through force feedback, filtering of hand tremor and motion scaling [1, 4]. In spite of extensive research, very few systems are fully commercially available and are expensive and bulky, for instance Da Vinci System.

Hence the intention to design a low cost and portable system for laparoscopic surgical and diagnostic procedures.

In a standard laparoscopic surgery generally 4 or more key holes are required [5]: one for scope, two for tools manipulated by surgeon and one for surgical tool handled by assisting nurse. In diagnostic and minor surgical procedures (gynecological) two or even one hole(s) could be sufficient [4]. In an ideal setup, tools should be placed at equal azimuth angles along a semi circular line (about 160 to 180 mm long) centered on the projection of the target organ and $\pi/3$ elevation angle [6]. The inter-port distance can be as close as 50 mm, but it is generally around 140 mm [3]. Angle between tools should be $\pi/3$. Workspace required to reach the full extent of the abdomen cavity, laterally and longitudinally, is $\pi/3$ and $\pi/2$ cone angle respect-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 119–126.

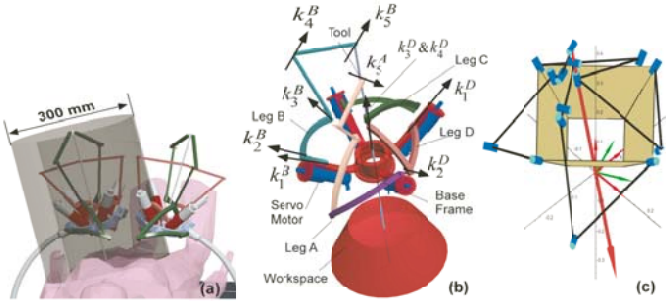


Fig. 1 Original (a) and new architecture (b) and (c).

ively [7]. Also surgeons spend 95% of the time in a conical workspace with vertex angle of $\pi/3$. The extrusion of 100 mm is adequate.

On the bases of these requirements, a 4 degree of freedom (3 rotational and 1 translational) purely parallel architecture was selected [10] with 4 legs of the same architecture.

2 Review of the Original Surgical System

A preliminary 3D virtual mock-up of the architecture in [10] was constructed, based on drive components and link design selected through force requirements. Two such virtual mock-ups were assembled to realize a surgical configuration with two surgical robots and the possible collision of links was investigated for movements of the robots in their entire work spaces. Due to the force requirements each robot has a large footprint and in the surgical configuration with two units, it is very hard to find a design avoiding collisions. As shown in Figure 1a, even with a design specifically developed to avoid collisions, at inter-trocar distance of more than 300 mm there are configurations in which arms are still inevitably interfering with each other.

The conclusion is that the architecture in [10] appears to be feasible for diagnostic setups where only one robot is used, and a modified architecture may result in a preferable setup for surgical applications requiring two robots.

Various modifications of the original architecture were attempted to preserve the advantageous characteristics for surgical application. The result is new architecture shown in Figures 1a and 1b.

3 Modified Architecture

The new architecture proposed is purely parallel with 4 serial leg chains. Two of the legs (labeled $L = A, B$) have architecture $(\underline{R}RR)R|R)$ and each comprise of a

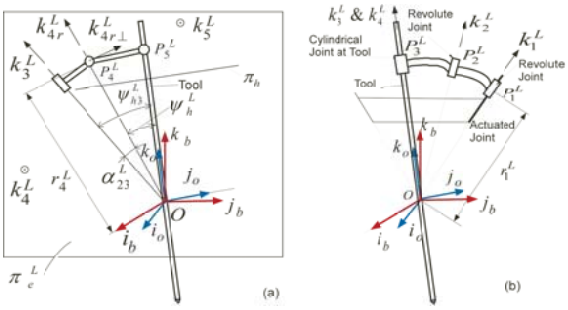


Fig. 2 (a) Heave sub-chain for legs A and B, (b) schematics of the legs C and D.

spherical sub-mechanism formed by the revolute joints $\boldsymbol{\xi}_1^L, \boldsymbol{\xi}_2^L, \boldsymbol{\xi}_3^L$ (with directions $\mathbf{k}_1^L, \mathbf{k}_2^L, \mathbf{k}_3^L$) intersecting at point O and a heave sub-mechanism with two parallel revolute joints $\boldsymbol{\xi}_4^L, \boldsymbol{\xi}_5^L$ (with direction $\mathbf{k}_4^L \parallel \mathbf{k}_5^L$), $\boldsymbol{\xi}_4^L \perp \boldsymbol{\xi}_3^L$. Other two legs (labeled $L = C, D$) with architecture $(\underline{R}RR)|P)$ comprise of a spherical chain $(\boldsymbol{\xi}_1^L, \boldsymbol{\xi}_2^L, \boldsymbol{\xi}_3^L)$ connected to the end effector by a prismatic joint $\boldsymbol{\xi}_4^L$ with direction parallel to $\boldsymbol{\xi}_3^L$. $\boldsymbol{\xi}_4^C$ and $\boldsymbol{\xi}_4^D$ are parallel and orthogonal to $\boldsymbol{\xi}_5^A$ and $\boldsymbol{\xi}_5^B$. We consider geometries with $\boldsymbol{\xi}_5^A \perp \boldsymbol{\xi}_5^B$ because there seem to be no advantage in different orientations to obtain the desired symmetric workspace.

Compared to the original mechanism in [10], in the new architecture the orientation of the tool is commanded by the spherical legs C and D without heave sub-mechanism which in the original mechanism created most of the collision problems. Legs A and B maintain the heave sub-mechanism required to command tool extrusion and torsion.

In the surgical setup with two robots, the work-spaces of the robots are oriented in a way that for any feasible torsion angle the legs A and B of both robots are maintained away from each other while shorter and easier to design legs C and D move in the space between the robots without collisions.

The following sections present the main steps undertaken to check the feasibility of the new architecture for surgical application and elements of its design.

3.1 Geometry and Configuration Parameters

Consider Figure 2 (with details of the geometry and configuration parameters). A fixed base coordinate frame $O\mathbf{i}_b\mathbf{j}_b\mathbf{k}_b$ and a rotating frame $O\mathbf{i}_o\mathbf{j}_o\mathbf{k}_o$ are introduced. The heave plane π_e^L for $L = A, B$ is through O and orthogonal to $\boldsymbol{\xi}_5^L$. Points P_4^L and P_5^L are at the intersection of $\boldsymbol{\xi}_4^L$ and $\boldsymbol{\xi}_5^L$ with π_e^L, respectively. r_4^L is the distance between P_4^L and O. π_h^L is the plane orthogonal to $\mathbf{k}_o$ through $\boldsymbol{\xi}_5^L$, h^L is the distance between π_h^A and π_h^L ($h^A = 0$). The extrusion of the end effector is measured by the distance h between π_h^A and O. The tilt angle α_1^L and azimuth angle β_1^L place the

Table 1 Geometry parameters (dimensionless referred to l_{45}^L).

leg	l_{45}^L/l_{45}^L	h^L/l_{45}^L	r_4^L/l_{45}^L	α_{34}^L	α_1^L	β_1^L	α_{12}^L	α_{23}^L
A	1	–0.125	1.875	0.09774	0.69813	0	1.01229	1.01229
B	1.375	0.75	2.125	0.09774	0.69813	0.69813	1.01229	1.01229
C	–	–	–	–	0.69813	3.14159	1.01229	1.01229
D	–	–	–	–	0.69813	4.71239	1.01229	1.01229

base joints $\boldsymbol{\xi}_1^L$, $L = A, B, C, D$, in $O\mathbf{i}_b\mathbf{j}_b\mathbf{k}_b$. α_{ij}^L is the angle between $\boldsymbol{\xi}_i^L$ and $\boldsymbol{\xi}_j^L$, where $i = 1, 2$ and $j = i + 1$.

A set of values shown in Table 1 for the geometry parameters has been heuristically selected from kinematics and work-space analysis. These values have been used as reference to investigate the suitability of the new mechanism in surgical setups with two robots.

To orientate $O\mathbf{i}_o\mathbf{j}_o\mathbf{k}_o$ w.r.t. $O\mathbf{i}_b\mathbf{j}_b\mathbf{k}_b$ the tilt-and-torsion (T&T) representation mentioned in [2] (angles $\{\theta, \phi, \psi\}$) is used for the more realistic depiction of constant torsion movements of the surgical tool compared to other representations e.g. Euler angles.

3.2 Mobility and Constraint Analysis

The mechanism is purely parallel. The constraint applied by each leg to the end-effector is analyzed and the overall end-effector constraint and mobility is discussed.

In any leg L, the twist systems spanned by all joint screws and by the passive joint screws are labeled, $\mathcal{T}_L$ and $\mathcal{P}_L$, respectively. The corresponding reciprocal wrench systems are $\mathcal{W}_L = \mathcal{T}_L^{\perp}$ (wrenches, the leg can transmit to the base with all joints free to move) and $\mathcal{V}_L = \mathcal{P}_L^{\perp}$ (wrenches, transmitted by the leg with the actuated joint locked).

Legs A, B and C, D are analyzed separately distinguishing for each of them different subsets, $\boldsymbol{\mathcal{C}}_i^L$, of the feasible leg configuration space, $\boldsymbol{\mathcal{C}}^L$, with different relative location of the passive joints and consequently different types of constraint systems. The classes for legs A, B are also mentioned in [12].

Consider first the legs A, B. The leg postures $\boldsymbol{q}^L$ are described geometrically referring to the planes π_{23}^L through $\boldsymbol{\xi}_2^L, \boldsymbol{\xi}_3^L$, π_{45}^L defined by $\boldsymbol{\xi}_4^L, \boldsymbol{\xi}_5^L$ and π_0 through O parallel to $\boldsymbol{\xi}_5^A, \boldsymbol{\xi}_5^B$. For each class, $\boldsymbol{\mathcal{C}}_i^L$, a subclass, $\boldsymbol{\mathcal{C}}_{i1}^L$, is defined by the additional condition $\mathbf{k}_1^L \parallel \pi_{23}^L$.

$\boldsymbol{\mathcal{C}}_1^L = \{\boldsymbol{q}^L \mid \mathbf{k}_4^L \nparallel \pi_{23}^L \text{ and } O \notin \pi_{45}^L\}$. $\mathcal{V}_L = \text{Span}\,(\boldsymbol{\varphi}_0^L, \boldsymbol{\varphi}^L)$, $\boldsymbol{\varphi}_0^L$ force at O with direction $\mathbf{k}_5^L$, $\boldsymbol{\varphi}^L$ force along $\pi_{23}^L \cap \pi_{45}^L$. If $\boldsymbol{q}^L \in \boldsymbol{\mathcal{C}}_{11}^L$, $\mathcal{W}_L = \mathcal{V}_L$ else $\mathcal{W}_L = \text{Span}\,(\boldsymbol{\varphi}_0^L)$.

$\boldsymbol{\mathcal{C}}_2^L = \{\boldsymbol{q}^L \mid \mathbf{k}_4^L \parallel \pi_{23}^L \text{ and } O \notin \pi_{45}^L\}$. $\mathcal{V}_L = \text{Span}\,(\boldsymbol{\varphi}_0^L, \boldsymbol{\mu}^L)$, $\boldsymbol{\mu}^L$ pure moment orthogonal to π_{23}^L). If $\boldsymbol{q}^L \in \boldsymbol{\mathcal{C}}_{21}^L$, $\mathcal{W}_L = \mathcal{V}_L$ else $\mathcal{W}_L = \text{Span}\,(\boldsymbol{\varphi}_0^L)$.

$\mathcal{C}_3^L = \{\boldsymbol{q}^L \mid \mathbf{k}_4^L \nparallel \pi_{23}^L \text{ and } O \in \pi_{45}^L\}$. $\mathcal{V}_L = \text{Span}\,(\boldsymbol{\varphi}_0^L, \boldsymbol{\varphi}_\perp^L)$, $\boldsymbol{\varphi}_\perp^L$ pure force through O on π_{45}^L orthogonal to $\mathbf{k}_4^L$. $\mathcal{W}_L = \mathcal{V}_L$ everywhere in $\mathcal{C}_3^L$.

$\mathcal{C}_4^L = \{\boldsymbol{q}^L \mid \mathbf{k}_4^L \parallel \pi_{23}^L \text{ and } O \in \pi_{45}^L\}$. $\mathcal{V}_L = \text{Span}\,(\boldsymbol{\varphi}_0^L, \boldsymbol{\mu}^L, \boldsymbol{\varphi}_\perp^L)$. If $\boldsymbol{q}^L \in \mathcal{C}_{41}^L$, $\mathcal{W}_L = \mathcal{V}_L$ else $\mathcal{W}_L = \text{Span}\,(\boldsymbol{\varphi}_0^L, \boldsymbol{\varphi}_\perp^L)$.

In any nonsingular posture (linearly independent leg joint screws) $\text{Span}\,(\boldsymbol{\varphi}_0^L) = \mathcal{W}_L \subsetneq \mathcal{V}_L$, $\dim \mathcal{V}_L = 2$. In any leg singularity $\text{Span}\,(\boldsymbol{\varphi}_0^L) \subsetneq \mathcal{W}_L \subset \mathcal{V}_L$.

Consider now legs C, D. We have only one class with subclass:

$\mathcal{C}_1^L = \{\boldsymbol{q}^L \mid \mathbf{k}_4^L \nparallel \pi_{23}^L \text{ and } O \notin \pi_{45}^L\}$. $\mathcal{V}_L = \text{Span}\,(\boldsymbol{\varphi}_{01}^L, \boldsymbol{\varphi}_{02}^L, \boldsymbol{\mu}^L)$, $\boldsymbol{\varphi}_{01}^L$ and $\boldsymbol{\varphi}_{02}^L$ forces at O orthogonal to $\mathbf{k}_4^L$, $\boldsymbol{\mu}^L$ moment orthogonal to π_{23}^L. If $\boldsymbol{q}^L \in \mathcal{C}_{11}^L$, $\mathcal{W}_L = \mathcal{V}_L$ else $\mathcal{W}_L = \text{Span}\,(\boldsymbol{\varphi}_{01}^L, \boldsymbol{\varphi}_{02}^L)$.

For leg C, D in any nonsingular posture (with linearly independent leg joint screws) $\text{Span}\,(\boldsymbol{\varphi}_{01}^L, \boldsymbol{\varphi}_{02}^L) = \mathcal{W}_L \subsetneq \mathcal{V}_L$, $\dim \mathcal{V}_L = 3$. In any leg singularity $\text{Span}\,(\boldsymbol{\varphi}_{01}^L, \boldsymbol{\varphi}_{02}^L) \subsetneq \mathcal{W}_L \subset \mathcal{V}_L$.

The $\mathcal{W}_L$ and $\mathcal{V}_L$ of the single legs combine in the end-effector spaces of *structural constraints*, $\mathcal{W} = \sum_L \mathcal{W}_L$, and *actuated constraints*, $\mathcal{V} = \sum_L \mathcal{V}_L$. If no leg is singular, $\mathcal{W} = \text{Span}\,(\boldsymbol{\varphi}_0^A, \boldsymbol{\varphi}_0^B, \boldsymbol{\varphi}_{01}^C, \boldsymbol{\varphi}_{02}^C, \boldsymbol{\varphi}_{01}^D, \boldsymbol{\varphi}_{02}^D)$. Because $\boldsymbol{\xi}_4^C \parallel \boldsymbol{\xi}_4^D \perp \pi_0$, $\mathcal{W} = \mathcal{W}_0 = \text{Span}\,(\boldsymbol{\varphi}_x, \boldsymbol{\varphi}_y)$, with $\boldsymbol{\varphi}_x$ and $\boldsymbol{\varphi}_y$ two forces at O lying in π_0 with directions $\mathbf{i}_o$, $\mathbf{j}_o$. $\dim \mathcal{W}_0 = 2$ and the mechanism has 3 rotational and 1 translational freedoms. $\dim \mathcal{V} = 6$ and the mechanism is correctly actuated.

Mechanism configurations with at least one $\boldsymbol{q}^L \in \mathcal{C}_3^L \cup \mathcal{C}_4^L$, $L = A, B$, belong to the extrusion boundary of the 4-dimensional workspace of the mechanism. The remainder of the boundary contains configurations with $\boldsymbol{q}^L \in \mathcal{C}_{11}^L \cup \mathcal{C}_{21}^L$, $L = A, B$, or $\boldsymbol{q}^L \in \mathcal{C}_{11}^L$, $L = C, D$. All boundary configurations are at least IO-type (Impossible Output) singular [8]. All mechanism configurations with at least one singular leg are singular.

Configurations with $\boldsymbol{q}^L \in \mathcal{C}_1^L \cup \mathcal{C}_2^L$, $L = A, B$, and $\boldsymbol{q}^L \in \mathcal{C}_1^L$, $L = C, D$, belong to the interior of the workspace. They can be singular although no leg is singular. The following discussion of position and velocity kinematics is mainly addressed to these mechanism configurations. In conclusion, we discuss the localization of RO-type singularities in the interior boundary and show that with the set of geometry parameters selected the desired workspace is singularity-free. There are no constraint singularities because $\dim \mathcal{W} < 3$ [9].

3.3 Inverse Position Kinematics

The base joint angles are computed for given tool pose $({}^B\mathbf{R}_o, h)$.

In each leg $\mathbf{k}_1^L = [\text{c}\beta_1^L\text{c}\alpha_1^L \ \text{s}\beta_1^L\text{c}\alpha_1^L \ -\text{s}\alpha_1^L]^T$. $\mathbf{k}_o$ is along the axis of the tool. For legs $L = C, D$, $\mathbf{k}_3^L = {}^B\mathbf{R}_o\mathbf{k}_o$, while $\mathbf{k}_2^L$ can be computed as linear combination of $\mathbf{k}_1^L, \mathbf{k}_3^L, \mathbf{k}_1^L \times \mathbf{k}_3^L$:

$$\mathbf{k}_2^L = \left(1 - (\mathbf{k}_1^L \cdot \mathbf{k}_3^L)^2\right)^{-1} [\mathbf{k}_1^L \ \mathbf{k}_3^L \ \mathbf{k}_1^L \times \mathbf{k}_3^L][t_1^L \ t_2^L \ t_3^L]^T \tag{1}$$

with $t_1^L = c_{12}^L - c_{23}^L(\mathbf{k}_1^L \cdot \mathbf{k}_3^L)$, $t_2^L = c_{23}^L - c_{12}^L(\mathbf{k}_1^L \cdot \mathbf{k}_3^L)$, $c_{ij}^L = \mathrm{c}\alpha_{ij}^L$, $s_{ij}^L = \mathrm{s}\alpha_{ij}^L$, $t_3^L = \delta^L\sqrt{(|s_{12}s_{23}| + e_{AB}^L)(|s_{12}s_{23}| - e_{AB}^L)}$, $e_{AB}^L = \mathbf{k}_1^L \cdot \mathbf{k}_3^L - c_{12}^L c_{23}^L$. Knowing the direction of $\mathbf{k}_2^L$ w.r.t. $\mathbf{k}_1^L$ the actuation angles can be finally computed.

For legs $L = A, B$, the angle ψ_h^L is derived as:

$$\mathrm{c}\psi_h^L = \left((h + h^L)^2 + {r_4^L}^2 - l_{45}^L\right)/(2h^L r_4^L) \tag{2}$$

while $\mathbf{k}_5^A = \mathbf{k}_4^A = \mathbf{i}_o$, $\mathbf{k}_5^B = \mathbf{k}_4^B = \mathbf{j}_o$. Through vector projection we obtain: $\mathbf{k}_{4r}^L = -\mathrm{s}\psi_h^L\,\mathbf{j}_o + \mathrm{c}\psi_h^L\,\mathbf{k}_o$ and $\mathbf{k}_3^L = -\mathrm{s}(\alpha_{34}^L + \psi_h^L)\,\mathbf{j}_o + \mathrm{c}(\alpha_{34}^L + \psi_h^L)\,\mathbf{k}_o$

Vector $\mathbf{k}_2^L$ is obtained as in Eq. (1). $P_5^L = (h + h^L)\mathbf{k}_o$, $P_4^L = r_4^L\mathbf{k}_{4r}^L$.

A graphical mock-up was constructed based on inverse kinematics in Maple to verify the mathematical model.

3.4 Forward Position Kinematics

The pose $({}^B\mathbf{R}_o, h)$ of the end-effector is computed from given actuator angles. The $\mathbf{k}_2^L$ are available from the actuated joint angles. Finding ${}^B\mathbf{R}_o$ requires determination of angles $\{\theta, \phi, \psi\}$. From Figure 2b, $\mathbf{k}_3^C = \mathbf{k}_3^D = \mathbf{k}_o$. $\mathbf{k}_o$ can be obtained by linear combination of $\mathbf{k}_2^C$, $\mathbf{k}_2^D$, $\mathbf{k}_2^C \times \mathbf{k}_2^D$ as outlined in Section 3.3. Angles ϕ and θ come from $\mathbf{k}_b \cdot \mathbf{k}_o = cos\phi$, $\mathbf{k}_b \times \mathbf{k}_o = \mathbf{j}'$, $\mathbf{j}_b \cdot \mathbf{j}' = \cos\theta$. Here $\mathbf{j}'$ is the unit vector along the T&T tilt axis.

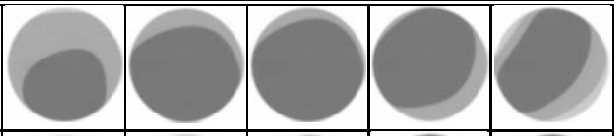
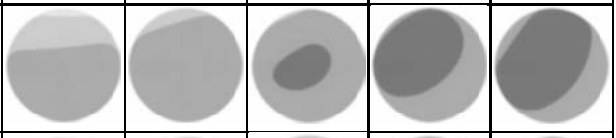
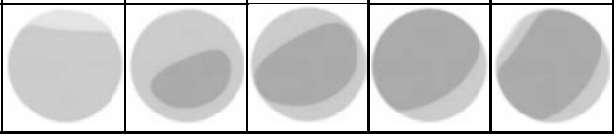
When legs A, B have same geometry, $\psi_3^A = \psi_3^B$ and $h, k_3^A, \psi_3^A, k_3^B$ are computed solving a system of 7 equations: $\mathbf{k}_o \cdot \mathbf{k}_3^L = \cos\psi_3^A$, $\mathbf{k}_2^L \cdot \mathbf{k}_3^L = \cos\alpha_{23}^A$ $(L = A, B)$ $\mathbf{k}_3^A \cdot \mathbf{k}_3^B = \cos^2\psi_3^A$ plus the unit vector equations for $\mathbf{k}_3^A$ and $\mathbf{k}_3^B$. $\psi_h = \psi_3^A - \alpha_{23}^A$. h is calculated from Eq. (2). Then $\mathbf{i}_o = \mathbf{k}_3^A \times \mathbf{k}_o$. Finally ψ is obtained from $\mathbf{j}_o \cdot \mathbf{j}_b = \cos\psi$, $\mathbf{j}_o = \mathbf{k}_o \times \mathbf{i}_o$.

3.5 Velocity and Singularity Analysis

We use the rotating reference frame $O\mathbf{i}_o\mathbf{j}_o\mathbf{k}_o$ for ease, and apply the method detailed in [11]. The input-output velocity equation is of the form: $[\,\mathbf{Z}_a\ \mathbf{Z}_c\,]^T\boldsymbol{\xi} = [\,\boldsymbol{\Lambda}\dot{\boldsymbol{\theta}}_a\ \mathbf{0}_2\,]^T$, where, out of singularities, $\mathbf{Z}_a = [\tilde{\boldsymbol{\varphi}}^A\,\tilde{\boldsymbol{\varphi}}^B\,\tilde{\boldsymbol{\mu}}^C\,\tilde{\boldsymbol{\mu}}^D]^T$, $\mathbf{Z}_c = [\tilde{\boldsymbol{\varphi}}_x\,\tilde{\boldsymbol{\varphi}}_y]^T$ (with the symbol $\sim$ on a wrench denoting switching of force and moment components), $\boldsymbol{\Lambda} = \mathrm{diag}\{\boldsymbol{\varphi}^A \circ \boldsymbol{\xi}_1^A, \boldsymbol{\varphi}^B \circ \boldsymbol{\xi}_1^B, \boldsymbol{\mu}^C \circ \boldsymbol{\xi}_1^C, \boldsymbol{\mu}^D \circ \boldsymbol{\xi}_1^D\}$.

The different entries of $\mathbf{Z}_a$ and $\boldsymbol{\Lambda}$ can be calculated using the geometry of the mechanism. In $O\mathbf{i}_o\mathbf{j}_o\mathbf{k}_o$, the $\mathbf{i}_o$ and $\mathbf{j}_o$ components of the translation velocity are always zero due to $\mathcal{W}_0$ and we can consider the simplified end-effector twist $\bar{\boldsymbol{\xi}}$ with the only three rotational and $\mathbf{k}_o$ translational components. The final form of the simplified velocity equations is:

Table 2 Conformal conditioning maps (the smaller the darker) for given extrusion and torsion – no singularities are present.

Extrusion h^A	$\tau = \pi/4$	$\tau = 5\pi/12$	$\tau = \pi/2$	$\tau = 2\pi/3$	$\tau = 11\pi/12$
(1.75) l_{45}^A					
(2.125) l_{45}^A					
(2.5) l_{45}^A					

$$\left[\begin{array}{c|c} (\mathbf{k}_2^A \times \mathbf{k}_3^A)^T & \frac{k_2^A k_3^A k_4^A}{-r_{45}^A} \\ (\mathbf{k}_2^B \times \mathbf{k}_3^B)^T & \frac{k_2^B k_3^B k_4^B}{-r_{45}^B} \\ (\mathbf{k}_2^C \times \mathbf{k}_3^C)^T & 0 \\ (\mathbf{k}_2^D \times \mathbf{k}_3^D)^T & 0 \end{array}\right] \bar{\boldsymbol{\xi}} = \begin{bmatrix} \mathbf{k}_1^A \mathbf{k}_2^A \mathbf{k}_3^A & 0 & 0 & 0 \\ 0 & \mathbf{k}_1^B \mathbf{k}_2^B \mathbf{k}_3^B & 0 & 0 \\ 0 & 0 & \mathbf{k}_1^C \mathbf{k}_2^C \mathbf{k}_3^C & 0 \\ 0 & 0 & 0 & \mathbf{k}_1^D \mathbf{k}_2^D \mathbf{k}_3^D \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^A \\ \dot{\theta}_1^B \\ \dot{\theta}_1^C \\ \dot{\theta}_1^D \end{bmatrix} \tag{3}$$

where r_{45}^L in leg $L = A, B$ is the distance between $\pi_{45}^L \cap \pi_0$ and O.

Extensive iterations of singularity analysis were performed before getting the geometry parameters listed in Table 1. The singularity free workspace obtained is a truncated cone with half angle $13\pi/90$ and extrusion range 90 mm (if l_{45}^A is taken as 120 mm). The torsional range of the mechanism is $2\pi/3$, which is sufficient for many surgical tools and procedures, though, it is targeted to extend the capacities of the mechanism by further refinements. The analysis checked for RO-type singularities and was performed through C++ code based on conditioning of $\mathbf{Z}$ and $\mathbf{\Lambda}$ matrices in Eq. (3). Color maps were generated to highlight the surfaces of singularity in the workspace. Results for some workspace sections (at constant h and torsion) for the torsional and extrusion range are shown in Table 2. White colors represent singular configurations while colors from light gray to dark gray show conditioning of the matrices. As the entries of matrix $\mathbf{Z}$ are non-homogeneous, the conditioning only determines the presence or absence of singularities. Since the analysis performed is at discrete locations it is not proven that no singularity exists in the workspace between the examined configurations but the lattice of configurations examined is very dense.

4 Conclusion

This paper presents a parallel architecture selected to develop a slave manipulator for laparoscopic surgery setups employing two manipulators. The architecture proposed overcomes some of the drawbacks of a previous version from which it is derived. All favorable characteristics of the original mechanism are maintained. Kinematics models, velocity patterns and singularities are discussed.

References

1. Asao, T., Kuwano, H., Mochiki, E. (2004), Laparoscopic surgery update for gastrointestinal malignancy, *Journal of Gastroenterology* **39**, 309–318.
2. Bonev, I.A., Zlatanov, D., Gosselin, C.M. (2002), Advantage of the modified Euler angles in the design and control of PKMs, in *Proceedings International Conference PKS02*, Chemnitz, Germany.
3. Cavusoglu, M.C., Villanueva, I., Tendick, F., (2001), Workspace analysis of robotic manipulator for a teleoperated suturing task, in *Proceedings of IEEE/RSJ IROS*, USA.
4. Darzi, A., Mackay, S. (2006), Recent advances in minimal access surgery, *BMJ* **324**, 31–34.
5. Faraz, A., Payandeh, S. (2000), *Engineering Approaches to Mechanical and Robotic Design for Minimally Invasive Surgeries*, Kluwer Academic Publishers, Dordrecht.
6. Ferzli, G.S., Fingerhut, A. (2004), Trocar placement for laparoscopic abdominal procedures: A simple standardized method, *J. of American College of Surgeons* **198**(1), 163–173.
7. Lum, M.J.H., Rosen, J., Hannaford, B., Sinanan, M.N. (2006), Optimization of a spherical mechanism for a minimally invasive surgical robot: Theoretical and experimental approaches, *IEEE Trans. on Biomed. Engrg.* **53**(7), 1140–1445.
8. Zlatanov, D., Benhabib, B., Fenton, R.G., (1995), A unifying framework for classification and interpretation of mechanism singularities, *ASME J. of Mech. Design* **117**, 566–572.
9. Zlatanov, D., Bonev, I., Gosselin, C., (2002), Constraint Singularities of parallel mechanisms, in *Proceedings IEEE International Conference on Robotics and Automation ICRA02*, Washington, USA, pp. 496–502.
10. Zoppi, M., Zlatanov, D., Gosselin, C.M. (2005), Analytical kinematics models and special geometries of a class of 4-DOF parallel mechanisms, *IEEE TRO* **21**(6), 1046–1055.
11. Zoppi, M., Zlatanov, D., Molfino, R. (2006), On the vel. analysis of interconnected chains mechanism, *Int. J. MMT* **41**(11), 1346–1358.
12. Zoppi, M. (2004), High dynamics parallel mechanisms: Contribution to force transmission and singularity analysis, PhD Thesis, DIMEC, University of Genoa, Italy.

Flapping-Wing Mechanism for a Bird-Sized UAVs: Design, Modeling and Control

Ch. Grand[1,2], P. Martinelli[2], J.-B. Mouret[1] and S. Doncieux[1]

[1]*ISIR, Université Pierre et Marie Curie-Paris 6, 4 Place Jussieu, 75252 Paris Cedex, France, e-mail: grand@isir.fr*
[2]*IUT Cachan, Université Paris Sud-Paris 11, 94230 Cachan, France*

Abstract. Birds daily execute complex maneuvers out of reach of current UAVs of comparable size. These capabalities are at least partly linked to the efficient flapping kinematics. This article describes the flapping wing mechanism developed within the ROBUR project to create a bird-sized UAV relying on such advanced kinematics.

Key words: UAV, flapping-wing, parallel mechanism, modeling, control.

1 Introduction

The capabilities of Mini-UAVs have drastically increased thanks to recent advances in terms of energy storage, effector power and electronic miniaturization, but they still remain far below the maneuverability and energy efficiency exhibited by birds and bats. An European kestrel for instance can stay above a point, fly forward at varying speeds, glide or soar to save energy, while demonstrating maneuverability capacities that far exceed those of the most efficient acrobatic aircraft. We have chosen to study flapping-flight for bird-sized UAVs, i.e., with a wing-span ranging from 0.5 to 1 meter and a weight of 500 g.

This article describes current research efforts targeted at designing a flapping-wing platform within the ROBUR project [1]. Whereas current flapping-flight artifacts show a maneuverability similar to that of fixed-wing engines, our overall objective is to design an aircraft whose capabilities will more closely resemble that of a kestrel, or at least of a pigeon. To this end, the wing kinematics have to be carefully controlled, for instance through the implementation of a neural network-based closed-loop control. As a consequence, the mechanical instantiation of our artificial bird must be able to move the wings on a wide range of periodical and non periodical trajectories, while remaining as energy-efficient as possible. This makes the corresponding wing-beat mechanism different from many mechanisms found in the literature [2–5], because they generate periodical or quasi-sinusoidal movements only. The design of such an improved device is especially challenging since, to prop-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 127–134.

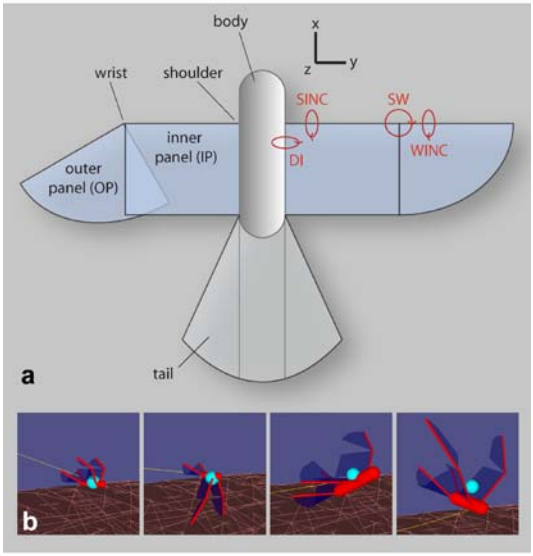


Fig. 1 The morphology of a simulated UAV.

erly size its actuators and mechanical parts, we must know the order of magnitude of torques, angle ranges and dimensions, which strongly depend on the wing-beat kinematics and the morphology of the artificial bird. The optimal kinematics, as well as the required degrees of freedom and the required power to fly at a given speed, being still open scientific questions, we want to address them while designing such mechanism.

This article deals with the design, the modeling and the control of the flapping-wing mechanism developed within the ROBUR project. In a first part, we briefly describe the results obtained on the optimal design of bird morphology and kinematics. Based on these results the wing beating mechanism that we have designed is then presented. In a second and a third part, the system modeling and its control are described.

2 Mechanical Design

In a first step, the morphology and kinematics the most adapted to several flying speeds have been sought using an evolutionary algorithm. Figure 1a shows the wing panels and their DOFs (dihedral (DI), sweep (SW), shoulder incidence (SINC) and wrist incidence (WINC)). In Figure 1b possible morphologies corresponding to boundary values of wing area (0.1–0.4 m^2) and wing aspect ratio (4.5–10) are illustrated.

Table 1 Angular ranges (in degrees) for the three DOFs (wing folding is not represented) for Pareto-optimal individuals.

Speed (m/s)	Dihedral	Shoulder incidence	Wrist incidence
6–8	15–50	0–30	10–50
10–12	25–45	0–15	8–15
16–20	30–65	0–5	1–10

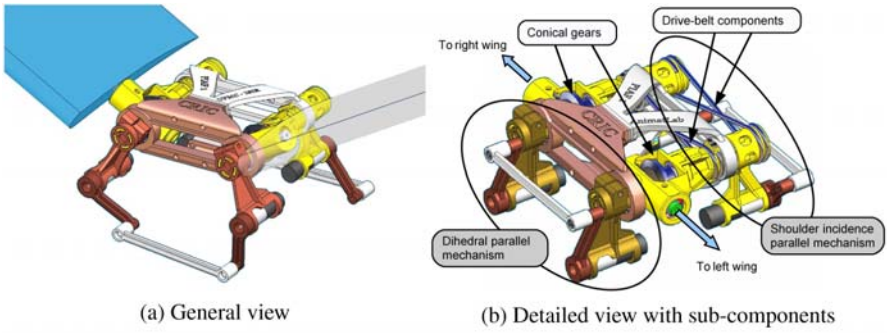


Fig. 2 Wing-beating mechanism.

The results of this optimization give us the first insights of the typical mechanical parameters required for a basic horizontal flight of a 0.5 kg UAV at different speed [6]. Optimization lead to a minimum energy consumption of 20–50 W/kg for a 10–12 m/s speed, with wing-beat frequencies between 3 to 5 Hz. Table 1 describes the obtained angle ranges for the different DOFs, for Pareto-optimal kinematics. This means that the UAV should at least allow the DOFs to move in these ranges for a basic flight along a straight line.

Capitalizing on these results, we designed a wing-beating mechanism that allows a wide range of dihedral and twist variations, with a high energetic and mechanical efficiency. To reach such capabilities, an innovative mechanism was developed as shown in Figure 2.

In this mechanism, the wings motion is produced by four position-controlled brushless motors (each motor is about 30 W power and 100 g weight). These four motors are associated two by two, constituting a parallel mecanism that uses two connected rod-crank mechanisms. Thus, the wings can be moved to follow an arbitrary trajectory and the power required to execute the quasi-sinusoidal movements is especially low. Each pair of motors is used to control: (1) the dihedral (DI) motion and (2) the shoulder incidence (SINC) motion.

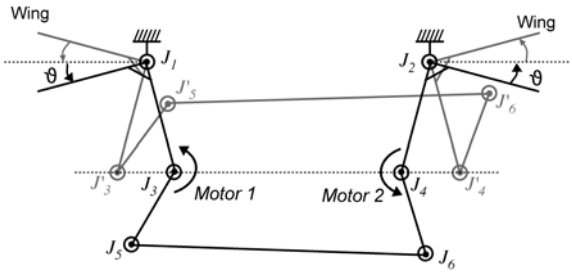


Fig. 3 Kinematic schema of the parallel mechanism.

3 Kinematical Modelling

The wing-beating system is composed of two identical mechanisms. This elementary part is a parallel system actuated by two motors. The first one, located in the front plane, is directely used to control the dihedral motion of the wings and the second one, located on the rear plane, controls the shoulder incidence motion through two conical gears. Each of these elementary systems is made symmetrical by using a drive-belt component (see Figure 2). Thus, in this first prototype, the flapping motion is identical for left and right wings (both for dihedral and twist motion).

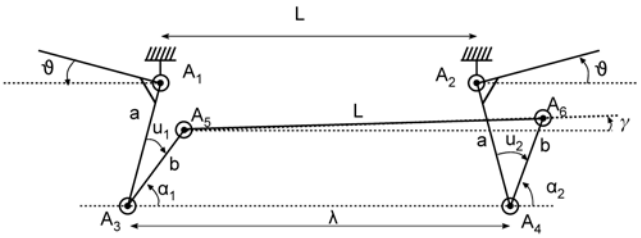
The elementary parallel mechanism is composed of five rods connected through six revolute joints. Figure 3 shows the kinematic schema of this parallel mechanism. Joint J_1 and J_2 are coupled by the drive-belt, joint J_3 and J_4 are actuated by two motors, and the last two joints (J_5, J_6) remain free. The mobility index of the mechanism is given by the classical Grubler equation:

$$m = \sum_{i=1}^{n} f_i - 3(n - b)$$

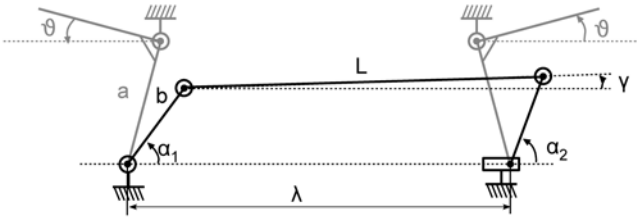
where b is the number of moving bodies, n the number of joints, f_i the number of degrees of freedom of each joint i. In this case, the mobility index is $m = 3$ (5 bodies, 6 revolute joints), but the overall system mobility is reduced to $m = 2$ when the symmetry condition from the drive-belt is considered. As these two degrees of freedom are actuated by two motors, the mechanism motion is completely constrained.

For analysis purpose, the system can be simplified by the kinematically equivalent system depicted in Figure 4b. Because of the symmetrical motion on joint J_1 and J_2 (angle θ), the motion of points A_3 and A_4 is constrained to an horizontal axis (A_i denotes the center of joint J_i). Thus, we can consider only the 3 rods mechanism and express the kinematic relation between λ and the input angles α_1 and α_2. The solution is obtained by solving the kinematical closure-form equations:

$$\begin{cases} b\cos\alpha_1 + L\cos\gamma - b\cos\alpha_2 = \lambda \\ b\sin\alpha_1 + L\sin\gamma - b\sin\alpha_2 = 0 \end{cases} \quad (1)$$



(a) Kinematical parameters



(b) Simplified schema

Fig. 4 Detailed kinematic schema.

Eliminating γ from these equations gives the following expression:

$$L^2 = \big(\lambda + b(\cos\alpha_1 - \cos\alpha_2)\big)^2 + b^2(\sin\alpha_1 - \sin\alpha_2)^2 \tag{2}$$

Then, λ can be determined from this second order equation. When the solution existence condition is satisfied ($L > \sqrt{2}b$), we obtain the following unique solution:

$$\lambda = \sqrt{L^2 - b^2(\sin\alpha_1 - \sin\alpha_2)^2} + b(\cos\alpha_1 - \cos\alpha_2) \tag{3}$$

And considering the schema in Figure 4a, the wing flapping angle θ is given as a function of λ:

$$\theta = \sin^{-1}\frac{L - \lambda}{2a} \tag{4}$$

The reader should notice that the parameters α_1 and α_2 are not directly the control inputs. But, if we consider $\mathbf{u} = [u_1\ u_2]^{\mathrm{t}}$ the input vector composed of the two motor angles (corresponding to the joint angles of J_3 and J_4), there can be computed as function of the parameters α_1, α_2 and the kinematical configuration of the mechanism characterized by the angle θ:

$$\begin{cases} u_1 = \left(\alpha_1 - \frac{\pi}{2}\right) - \theta \\ u_2 = \left(\alpha_2 - \frac{\pi}{2}\right) + \theta \end{cases}$$

However for the mathematical description of the kinematical model, it is more efficient to consider the input parameters (α_1, α_2) instead of the motor angles (u_1, u_2).

The system presents two control inputs α_1 and α_2 for one state parameter θ. So, the mechanism is over-actuated and we need to determine the compatible angles. Thus, let us introduce a new set of input variables α and φ, respectively the mean input angle and the half-phase angle:

$$\begin{cases} \alpha = \frac{1}{2}(\alpha_2 + \alpha_1) \\ \varphi = \frac{1}{2}(\alpha_2 - \alpha_1) \end{cases} \quad \text{and} \quad \begin{cases} \alpha_1 = \alpha - \varphi \\ \alpha_2 = \alpha + \varphi \end{cases}$$

Equation (3) becomes:

$$\lambda = \sqrt{L^2 - 4b^2 \cos^2\alpha \, \sin^2\varphi} + 2b \, \sin\alpha \, \sin\varphi \tag{5}$$

And the relations between the motor angular positions (u_1 and u_2) and the new variables become:

$$\begin{cases} u_1 = \left(\alpha - \frac{\pi}{2}\right) - (\theta + \varphi) \\ u_2 = \left(\alpha - \frac{\pi}{2}\right) + (\theta + \varphi) \end{cases} \quad \text{and} \quad \begin{cases} \alpha = \frac{1}{2}(u_2 + u_1) + \frac{\pi}{2} \\ \varphi + \theta = \frac{1}{2}(u_2 - u_1) \end{cases}$$

4 Motion Control of the Flapping Angle

In this part, we demonstrate that the previous choice of variables α and φ leads to a simplified control model of the wings-flapping motion. It allows to define quasi-sinusoidal motion given by the desired flapping frequency and amplitude.

Equation (5) can be differentiated with respect to α:

$$\frac{d\lambda}{d\alpha} = 2b \cos\alpha \, \sin\varphi \left[1 + \frac{2b \sin\alpha \, \sin\varphi}{\sqrt{\frac{L^2}{4b^2} - \cos^2\alpha \, \sin^2\varphi}} \right] \tag{6}$$

By considering the schema depicted in Figure 3, one can conclude that the wing angle θ is at its extremums when λ is also at its extremums. In order to find the extremum positions $\alpha_{\min}$ and $\alpha_{\max}$, the relation $d\lambda/d\alpha = 0$ must be solved. This leads to the following solutions:

$$\begin{cases} \cos\alpha = 0 \\ \sin\varphi = \pm\dfrac{L}{2b} \end{cases}$$

Thus, for any values of φ there are only two extremum positions at each periode that are solutions of $\cos\alpha = 0$. When introducing this result in Equations (5) and (4), we can find the maximum values for λ and consequently for θ:

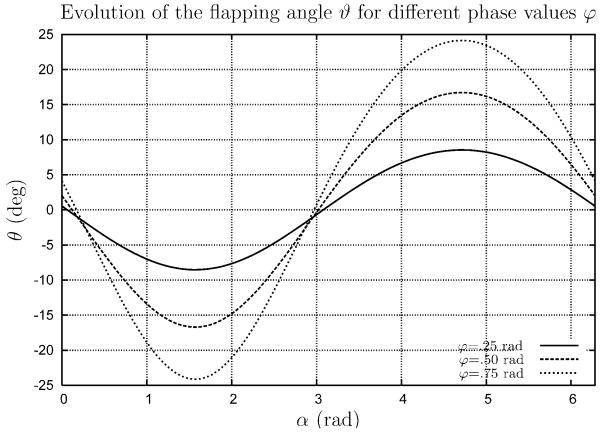


Fig. 5 Exemple of θ trajectory obtained with kinematic based control.

$$\begin{cases} \lambda_{\max} = L \pm 2\,b\,\sin\varphi \\ a\sin\theta_{\max} = b\sin\varphi \end{cases} \tag{7}$$

Figure 5 shows some trajectories θ obtained for different phase angles φ. The motion is quasi-sinusoidal, its amplitude depends on φ and its frequency can be modulated thanks to the velocity term $\dot{\alpha}$. So, if the desired quasi-sinusoidal trajectory is specified by its frequency f_θ and its amplitude $\theta_{\max}$, the velocity $\dot{\alpha}$ and the phase are determined as follows:

$$\begin{cases} \dot{\alpha} = 2\pi f_\theta \\ \varphi = \sin^{-1}\left(\dfrac{a}{b}\sin\theta_{\max}\right) \end{cases} \tag{8}$$

In order to extent this simple quasi-sinusoidal control to an arbitrary flapping motion, we need to investigate the differential kinematic model. The flapping velocity is given by $\dot{\theta}$ that can be expressed as a function of the input velocities $(\dot{\alpha},\ \dot{\varphi})$:

$$\dot{\theta} = \frac{d\theta}{d\lambda}\left(\frac{d\lambda}{d\alpha}\dot{\alpha} + \frac{d\lambda}{d\varphi}\dot{\varphi}\right) = J_\alpha\dot{\alpha} + J_\varphi\dot{\varphi} \tag{9}$$

The corresponding Jacobian terms are determined from Equations (4) and (6):

$$\begin{cases} J_\alpha = \dfrac{d\theta}{d\lambda}\dfrac{d\lambda}{d\alpha} = -\dfrac{b\cos\alpha\,\sin\varphi}{a\sqrt{1-\left(\frac{L-\lambda}{2a}\right)^2}}\left[1 + \dfrac{2b\sin\alpha\,\sin\varphi}{\sqrt{\frac{L^2}{4b^2} - cos^2\alpha\,\sin^2\varphi}}\right] \\ J_\varphi = \dfrac{d\theta}{d\lambda}\dfrac{d\lambda}{d\varphi} = -\dfrac{b\sin\alpha\,\cos\varphi}{a\sqrt{1-\left(\frac{L-\lambda}{2a}\right)^2}}\left[1 - \dfrac{2b\cos\alpha\,\cot\alpha\sin\varphi}{\sqrt{\frac{L^2}{4b^2} - cos^2\alpha\,\sin^2\varphi}}\right] \end{cases} \tag{10}$$

Then, the control principle is to compute the instantanous velocities terms $(\dot{\alpha},\ \dot{\varphi})$ as a function of the desired flapping velocity $\dot{\theta}^c$. This desired velocity becomes the control input used to follow a desired trajectory $\theta(t)$ with a classical feedforward controller:

$$\dot{\theta}^c = K_1(\theta - \theta^m) + \dot{\theta} \tag{11}$$

where θ^m is the measured flapping angle, K_1 is a positive gain and $\dot{\theta}$ is the feedforward velocity computed from the trajectory $\theta(t)$. Then, the control law becomes:

$$\begin{cases} \dot{\varphi} = K_2(\varphi^d - \varphi^m) \\ \dot{\alpha} = \dfrac{1}{J_\alpha}(\dot{\theta}^c - J_\varphi \dot{\varphi}) \end{cases} \tag{12}$$

where φ^d is the phase angle determined from the maximum absolute values of the flapping angle trajectory $\theta(t)$ on a given time horizon.

5 Conclusion

These results constitute the first stones of a long work towards a fully maneuverable flapping-wing UAV. Kinematical and morphological data from an evolutionary optimization process have been used to properly dimension a flapping-mechanism able to move the wing dihedral and incidence to follow arbitrary kinematics. The kinematic model of this innovative parallel mechanism has been detailed in this paper. Based on this model, a simple control law for quasi-sinusoidal motions has been developed. Its extension to a velocity model based controller that is able to follow various cyclic trajectories has been proposed.

References

1. Doncieux, S. et al. (2006), Building an artificial bird: Goals and accomplishments of the ROBUR Project, *European Micro Aerial Vehicles (EMAV)*.
2. Pornsin-Sirirak, T.N., Tai, Y.C., Ho, C.M. and Keennon, M. (2001), Microbat: A palm-sized electrically powered ornithopter, in *Proceedings of NASA/JPL Workshop on Biomorphic Robotics*, Pasadena, CA, August 14–17.
3. Vest, M.S. and Katz, J. (1999), Aerodynamic study of a flapping-wing micro-UAV, in *Proceedings 37th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, January 11–14.
4. Raney, D.L. and Slominski, E.C. (2004), Mechanization and control concepts for biologically inspired micro air vehicles, *Journal of Aircraft* **41**(6), 1257–1265.
5. Hunt, R., Hornby, G.S. and Lohn, J.D. (2004), Toward evolved flight, in *Proceedings of the 2005 Conference on Genetic and Evolutionary Computation*, pp. 957–964.
6. de Margerie, E., Mouret, J.-B., Doncieux, S. and Meyer, J.-A. (2007), Artificial evolution of the morphology and kinematics in a flapping-wing mini UAV. *Bioinspir. Biomim.* **2**, 65–82.

METHODS IN KINEMATICS

Extended Jacobian Inverse Kinematics and Approximation of Distributions

Mariusz Janiak and Krzysztof Tchoń

Institute of Computer Engineering, Control and Robotics,
Wrocław University of Technology, ul. Janiszewskiego 11/17,
50-372 Wrocław, Poland, e-mail: {mariusz.janiak, krzysztof.tchon}@pwr.wroc.pl

Abstract. The paper addresses the synthesis problem of repeatable Jacobian inverse kinematics algorithms for robotic manipulators. For the kinematics of redundancy 1 this synthesis is accomplished by defining an extended Jacobian inverse that in certain sense approximates the Jacobian pseudo-inverse. The approximation problem is formulated in differential geometric terms, and solved using the existing results on the approximation of a prescribed codistribution by an integrable codistribution. As an illustration, extended Jacobian inverses are derived for the normal form kinematics of a stationary manipulator and a mobile robot.

Key words: robot kinematics, redundancy, extended Jacobian inverse, Jacobian pseudo-inverse, distribution, approximation.

1 Introduction

Suppose that the map $k : R^n \longrightarrow R^m, \quad y = k(q)$, where $m < n$, denotes a coordinate representation of the forward kinematics of a robotic manipulator. Given a task space point y_d, the inverse kinematic problem consists in determining a joint position q_d such that $k(q_d) = y_d$. This problem is usually solved by means of the continuation method [4], in the following way. We choose an initial configuration q_0, take a smooth curve $q(\theta), \theta \in R$, such that $q(0) = q_0$, and define a task space error $e(\theta) = k(q(\theta)) - y_d$ along this curve. The curve should render a convergence of the error to 0; we let the convergence be exponential with a rate $\gamma > 0$, i.e. $d\,e(\theta)/d\,\theta = -\gamma e(\theta)$. A differentiation of the error $e(\theta)$ leads to the differential equation

$$J(q)\frac{d\,q}{d\,\theta} = -\gamma(k(q) - y_d)$$

for the curve $q(\theta)$, referred to as the Ważewski–Davidenko equation [2, 11]. From this equation, for every right inverse $J^{\#}(q)$ of the Jacobian $J(q)$, we obtain an inverse kinematics algorithm in the form of a dynamic system

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 137–146.

$$\frac{d\,q}{d\,\theta} = -\gamma\, J^{\#}(q)(k(q) - y_d), \tag{1}$$

whose trajectory in the limit yields the solution, $q_d = \lim_{\theta \to +\infty} q(\theta)$ of the inverse kinematic problem.

Geometrically, the columns of the matrix $J^{\#}(q)$ may be treated as vector fields spanning a distribution

$$\mathcal{D} = \mathrm{span}_{C^\infty(R^n)}\{J_1^{\#}(q), \ldots, J_m^{\#}(q)\} \tag{2}$$

associated with (1). Among the most often used right inverses there is the Jacobian pseudo-inverse (the Moore–Penrose inverse of the Jacobian), defined as $J^{\#P}(q) = J^T(q)(J(q)J^T(q))^{-1}$, and the extended Jacobian inverse $J^{\#E}(q) = \bar{J}(q)^{-1}|_{m \text{ first columns}}$, that, after introducing an augmenting kinematics map $h : R^n \to R^{n-m}$, results from the extended Jacobian

$$\bar{J}(q) = \left[J^T(q) \left(\frac{\partial h(q)}{\partial q} \right)^T \right]^T.$$

It is well known that the Jacobian pseudo-inverse algorithm has outstanding (quadratic) convergence, whereas the extended Jacobian algorithm is repeatable. Let us remind that an inverse kinematics algorithm is called repeatable, if every closed path in the task space is transformed by the algorithm into a closed path in the joint space. Repeatable control strategies become most advantageous when the manipulator carries out cyclic sequences of tasks. It is well known that the Jacobian pseudo-inverse inverse kinematics algorithm is not repeatable [3]. Necessary and sufficient conditions for repeatability of inverse kinematics algorithms for stationary manipulators have been derived by Shamir and Yomdin [8], and generalized to mobile manipulators by Tchoń [9]. These conditions require integrability of the associated distribution. By design, the extended Jacobian inverse kinematics algorithm is repeatable.

An idea of making a fusion of convergence and repeatability inherent in these two inverse kinematics algorithms comes from [5, 6], and consists in defining a repeatable (e.g. extended Jacobian) inverse that optimally approximates, in the sense of the minimization of an error functional, the Jacobian pseudo-inverse. Apparently, the optimality conditions for this approximation problem look rather intractable, moreover, topologically such an approximation presents a sort of ill-posed problem, for the set of repeatable inverses is very small (of infinite codimension).

The objective of this paper is to set forth new a synthesis procedure of the extended Jacobian inverse kinematics algorithms that approximate the Jacobian pseudo-inverse. Instead of the variational approach, we take up a differential geometric formulation of the approximation problem, adopting as a guideline the method of approximation of codistributions expounded in [7]. A codistribution is a geometric object dual to the distribution, spanned by differential one-forms that annihilate the distribution. Given a codistribution, the method provides an integrable codistribution that agrees with the original codistribution on certain submanifolds

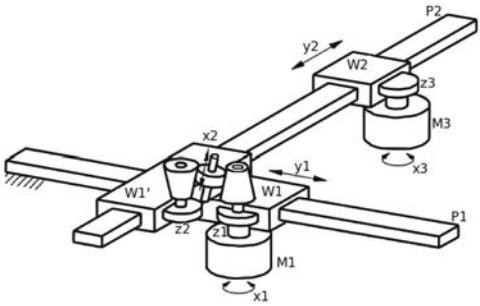


Fig. 1 3DOF manipulator.

and along a specific vector field associated with a homotopy map. The resulting codistribution is spanned by differentials of certain functions that in the robotic context compose into an augmenting kinematics map. For simplicity, we shall present the main idea of this method restricting to the kinematics of redundancy 1. Both the stationary and the mobile robot kinematics will be considered.

The composition of this paper is the following. In Section 2 we apply the codistribution approximation method to the kinematics of a 3 degree-of-freedom (DOF) manipulator. Section 3 will show corresponding results for a chained-form system. Section 4 presents computer simulations. The paper concludes with Section 5.

2 Stationary Manipulator

In order to present the main ingredients of the differential geometric approximation procedure, we shall use the kinematics of a 3DOF stationary manipulator studied in [10] and shown in Figure 1.

As can be seen, the manipulator has 3 joint variables (x_1, x_2, x_3) and 2 tasks coordinates (y_1, y_2) describing the Cartesian position of the car $W2$ with respect to the inertial coordinate frame fixed to the base. The joint variable x_1 is moved directly by the motor $M1$, similarly x_3 is driven by $M3$. The position of $W2$ along the runner $P2$ depends on x_3, and also on the revolution angle of the toothed wheel $z2$. This angle is coupled with the revolution angle of the toothed wheel $z1$ through a transmission gear whose gear ratio is adjusted by the joint variable x_2 moved by a motor $M2$ (not shown in the figure). A computation yields the kinematics $y_1 = c_1 x_1$ and $y_2 = f_2(x_2)x_1 + c_3 x_3$ for constants c_1 , c_3 and a nonlinear function $f_2(x_2)$. After a change of coordinates $q_1 = c_1 x_1$, $q_2 = c_3 x_3$ and $q_3 = \frac{1}{c_1} f_2(x_2)$ these kinematics may be given the normal form $k(q) = (q_1, q_2 + q_1 q_3)$. Accordingly, the Jacobian becomes

$$J(q) = \begin{bmatrix} 1 & 0 & 0 \\ q_3 & 1 & q_1 \end{bmatrix}.$$

By definition, the distribution (2) associated with the Jacobian pseudo-inverse $J^{\#P}(q) = J^T(q)(J(q)J^T(q))^{-1}$, is generated by the columns of $J^T(q)$. Equivalently, this distribution may also be represented as

$$\mathcal{D}_P = \mathrm{span}_{C^\infty(R^3)}\{X_1(q), X_2(q)\}, \tag{3}$$

where $X_1(q) = (1, 0, 0)^T$, $X_2(q) = (0, 1, q_1)^T$.

In accordance with Sluis et al. [7], we begin the approximation procedure with picking a one-form $\omega(q) = (0, -q_1, 1)$ annihilating (3). Then, we introduce a foliation of R^3 whose leaves are vertical lines parallel to the unit vector $e_3 = (0, 0, 1)^T$, i.e. $E_\alpha = \{\alpha_1\} \times \{\alpha_2\} \times R$, where $\alpha_i \in R$. By means of the map $\Phi_t(q) = tq + (1-t)q_3 e_3 = (tq_1, tq_2, q_3)^T$, $t \in [0, 1]$, we establish a homotopy between R^3 and the basic leaf $E_0 = \{0\} \times \{0\} \times R$. A differentiation of this homotopy yields a vector field

$$X(q) = \frac{d}{dt}\Big|_{t=1} \Phi_t(q) = (q_1, q_2, 0)^T.$$

Now, our objective is to define a function $h(q)$ such that its differential agrees with $\omega(q)$ along the vector field $X(q)$, and on the leaves of the foliation, i.e.

$$dh(q)X(q) = \omega(q)X(q) \quad \text{and, for every } \alpha = (\alpha_1, \alpha_2), \quad dh|_{E_\alpha} = \omega|_{E_\alpha}. \tag{4}$$

To this aim, let us compute $F(q) = \omega(q)X(q) = -q_1 q_2$, and divide the joint space coordinates $(q_1, q_2, q_3) \in R^3$ into $z = q_3$ along the leaves, and $y_1 = q_1$, $y_2 = q_2$ in the horizontal plane. For (y, z) close to E_0 the conditions (4) result in a partial differential equation

$$\frac{\partial h \circ \Phi_s(y, z)}{\partial s} - \frac{F \circ \Phi_s(y, z)}{s} \frac{\partial h \circ \Phi_s(y, z)}{\partial z} = 0,$$

parametrized by y, whose characteristic equation takes the form

$$\frac{dz}{ds} = -\frac{F(\Phi_s(y, z))}{s} = s y_1 y_2. \tag{5}$$

The solution of (5) with initial condition $z(0) = z_0$ is $z(s) = z_0 + \frac{1}{2} y_1 y_2 s^2$. After taking $s = 1$ and setting $z = z(1)$, we compute $z_0(y, z) = z - \frac{1}{2} y_1 y_2$, and obtain the augmenting kinematics function $h(q) = q_3 - \frac{1}{2} q_1 q_2$. In conclusion, the resulting extended Jacobian inverse

$$J^{\#E1}(q) = \frac{1}{1 + \frac{1}{2}q_1^2} \begin{bmatrix} 1 + \frac{1}{2}q_1^2 & 0 \\ -q_3 - \frac{1}{2}q_1 q_2 & 1 \\ \frac{1}{2}q_2 - \frac{1}{2}q_1 q_3 & \frac{1}{2}q_1 \end{bmatrix}. \tag{6}$$

Alternatively, a similar derivation for the one-form $\omega(q) = (0, 1, -q_1^{-1})$, $q_1 > 0$, the foliation $E_{a,\alpha} = \{a + \alpha_1\} \times R \times \{\alpha_3\}$, where $a > 0$, $\alpha_1 > -a$ and $\alpha_3 \in R$ of the half-space $(0, +\infty) \times R^2$ by horizontal lines such that the basic leaf $E_{a,0} = \{a\} \times R \times \{0\}$, and for the homotopy $\Phi_t(q) = tq + (1-t)(a, q_2, 0)^T$ leads to the characteristic equation

$$\frac{d\,z}{d\,s} = \frac{y_2}{sy_1 + a}.$$

Having solved this equation, we discover the augmenting function

$$h(q) = q_2 - \frac{q_3}{q_1 - a} \ln\left(\frac{q_1}{a}\right), \quad q_1 > 0,$$

and obtain the extended Jacobian inverse

$$J^{\#E2}(q) = \begin{bmatrix} 1 & 0 \\ q_3 \frac{-2q_1 \ln(\frac{q_1}{a}) + a \ln(\frac{q_1}{a}) + q_1 - a}{(-q_1+a)(-\ln(\frac{q_1}{a}) - q_1^2 + q_1 a)} & \frac{\ln(\frac{q_1}{a})}{\ln(\frac{q_1}{a}) + q_1^2 - q_1 a} \\ q_3 \frac{-q_1^3 + 2q_1^2 a - q_1 a^2 + q_1 \ln(\frac{q_1}{a}) - q_1 + a}{q_1(-q_1+a)(-\ln(\frac{q_1}{a}) - q_1^2 + q_1 a)} & \frac{-q_1+a}{-\ln(\frac{q_1}{a}) - q_1^2 + q_1 a} \end{bmatrix}. \tag{7}$$

Results of computer simulations illustrating the performance of extended Jacobian inverses (6) and (7) compared with the Jacobian pseudo-inverse will be shown in Section 4.

3 Chained-Form System

The codistribution approximation approach can also be applied to mobile robots and mobile manipulators. As an example, we shall consider the chained-form system

$$\dot{q}_1 = u_1 \;\; \dot{q}_2 = u_2, \;\; \dot{q}_3 = q_2 u_1, \tag{8}$$

representing a feedback equivalent of the differential drive-type mobile platform. The kinematics of this mobile robot are defined as the end-point map of the control system (8), see [1]. Let $q(t) = \varphi_{q_0,t}(u(\cdot))$ denote the state trajectory of this system steered by a control function $u(\cdot)$. Then, given a control time horizon $T > 0$, the kinematics are a map from control functions into the state space of (8), identified with $q(T)$. Because the domain of the kinematics is an infinite dimensional functional space, for computations we need to resort to a finite dimensional expansion of control functions. In this way the kinematics become a map between finite dimensional spaces. For the chained-form system we have adopted the controls

$$u_1(t) = \lambda_1 + \lambda_2 \sin t, \quad u_2(t) = \lambda_3 + \lambda_4 \sin t.$$

A substitution into (8) and setting $q_0 = 0$, $T = 2\pi$ results in the kinematics

$$q(T) = K(\lambda) = \left(T\lambda_1, T\lambda_3, \frac{1}{2}T^2\lambda_1\lambda_3 - T\lambda_2\lambda_3 + T\lambda_1\lambda_4\right)^T,$$

and the Jacobian

$$J(\lambda) = \begin{bmatrix} T & 0 & 0 & 0 \\ 0 & 0 & T & 0 \\ \frac{1}{2}T^2\lambda_3 + T\lambda_4 & -T\lambda_3 & \frac{1}{2}T^2\lambda_1 - T\lambda_2 & T\lambda_1 \end{bmatrix}.$$

The derivation of the approximation will be done for the annihilating one-form

$$\omega(\lambda) = \left(0, 1, 0, \frac{\lambda_3}{\lambda_1}\right), \quad \lambda_1 > 0,$$

and the foliation $E_{a,\alpha} = \{\alpha_1 + a\} \times R \times \{\alpha_3\} \times \{\alpha_4\}$ with parameter $a > 0$, where $\alpha_1 > -a$, $\alpha_3, \alpha_4 \in R$, so that the basic leaf $E_{a,0} = \{a\} \times R \times \{0\} \times \{0\}$. Now, we choose the homotopy $\Phi_t(\lambda) = t\lambda + (1-t)(a, \lambda_2, 0, 0)^T$, and derive the characteristic equation

$$\frac{d\,z}{d\,s} = -\frac{s y_2 y_3}{s y_1 + a},$$

whose solution yields the augmenting function

$$h(\lambda) = \lambda_2 + \frac{\lambda_3\lambda_4}{(\lambda_1 - a)^2}\left(\lambda_1 - a - a\ln\left(\frac{\lambda_1}{a}\right)\right),$$

defined for $\lambda_1 > 0$. In consequence, the extended Jacobian inverse

$$J^{\#E}(\lambda) = \begin{bmatrix} 1 & 0 & 0 \\ J_{21}^{\#E} & J_{22}^{\#E} & J_{23}^{\#E} \\ 0 & 1 & 0 \\ J_{41}^{\#E} & J_{42}^{\#E} & J_{43}^{\#E} \end{bmatrix}, \tag{9}$$

where

$$J_{21}^{\#E} = -\frac{\lambda_3}{2(-\lambda_1 + a)(\lambda_3^2\lambda_1 - \lambda_3^2 a - \lambda_3^2 a(\ln a + \ln\lambda_1) + \lambda_1^3 - 2\lambda_1^2 a + \lambda_1 a^2)} \times$$
$$\big(\lambda_3 T\lambda_1^2 - 2\lambda_3 T\lambda_1 a + \lambda_3 T a^2 - \lambda_3 T\lambda_1 a(\ln a + \ln\lambda_1) + \lambda_3 T a^2(\ln a + \ln\lambda_1) +$$
$$4\lambda_4\lambda_1^2 - 4\lambda_4\lambda_1 a - 6\lambda_4\lambda_1 a(\ln a + \ln\lambda_1) + 2\lambda_4 a^2(\ln a + \ln\lambda_1)\big),$$
$$J_{22}^{\#E} = -\frac{(-\lambda_1 + a + a(\ln a + \ln\lambda_1))(\lambda_3 T\lambda_1 - 2\lambda_3\lambda_2 - 2\lambda_4\lambda_1)}{2(\lambda_3^2\lambda_1 - \lambda_3^2 a - \lambda_3^2 a(\ln a\ + \ln\lambda_1) + \lambda_1^3 - 2\lambda_1^2 a + \lambda_1 a^2)},$$
$$J_{23}^{\#E} = \lambda_3\frac{-\lambda_1 + a + a(\ln a + \ln\lambda_1)}{(\lambda_3^2\lambda_1 - \lambda_3^2 a - \lambda_3^2 a(\ln a + \ln\lambda_1) + \lambda_1^3 - 2\lambda_1^2 a + \lambda_1 a^2)},$$

$$J_{41}^{\#E} = -\frac{1}{2(-\lambda_1 + a)\lambda_1(\lambda_3^2\lambda_1 - \lambda_3^2 a - \lambda_3^2 a(\ln a + \ln \lambda_1) + \lambda_1^3 - 2\lambda_1^2 a + \lambda_1 a^2)} \times$$
$$\big(- \lambda_3 T\lambda_1^4 + 3\lambda_3 T\lambda_1^3 a - 3\lambda_3 T\lambda_1^2 a^2 + \lambda_3 T\lambda_1 a^3 - 2\lambda_4\lambda_1^4 + 6\lambda_4\lambda_1^3 a - 6\lambda_4\lambda_1^2 a^2 +$$
$$2\lambda_4\lambda_1 a^3 + 2\lambda_3^2\lambda_4\lambda_1^2 - 4\lambda_3^2\lambda_4\lambda_1 a(\ln a + \ln \lambda_1) - 2\lambda_3^2\lambda_4 a^2\big),$$

$$J_{42}^{\#E} = -\frac{1}{(\lambda_3^2\lambda_1 - \lambda_3^2 a - \lambda_3^2 a(\ln a + \ln \lambda_1) + \lambda_1^3 - 2\lambda_1^2 a + \lambda_1 a^2)} \times$$
$$\big(2\lambda_3\lambda_4\lambda_1 - 2\lambda_3\lambda_4 a - 2\lambda_3\lambda_4 a(\ln a + \ln \lambda_1) + \lambda_1^3 T - 2\lambda_1^2 T a + \lambda_1 T a^2 -$$
$$2\lambda_2\lambda_1^2 + 4\lambda_2\lambda_1 a - 2\lambda_2 a^2\big),$$

$$J_{43}^{\#E} = \frac{(-\lambda_1 + a)^2}{\lambda_3^2\lambda_1 - \lambda_3^2 a - \lambda_3^2 a(\ln a + \ln \lambda_1) + \lambda_1^3 - 2\lambda_1^2 a + \lambda_1 a^2}.$$

The results of computer simulations of the extended Jacobian inverse vs. the Jacobian pseudo-inverse will be presented in the next section.

4 Computer Simulations

Computer simulations of the inverse kinematics algorithms defined by $J^{\#P}$, $J^{\#E1}$ and $J^{\#E2}$ for the stationary manipulator were run for $y_d = (0.9, 2)$ and $q_0 = (0.1, 0, 0)$. Additionally, in the extended Jacobian inverse $J^{\#E2}$ we used $a = 5$. The performance of these algorithms has been portrayed in the following figures. Figures 2, 4 and 6 display the convergence in the joint space of solution of the dynamic system (1) for the Jacobian pseudo-inverse and for the extended Jacobian inverses (6) and (7). Figures 3, 5 and 7 display the corresponding paths in the task space.

The computations of the inverse kinematics for the chained-form system have been carried out for $T = 2\pi$, $q_0 = (0, 0, 0)$, the desirable task space point $y_d = (5, 5, -2)$, and the initial controls $\lambda_0 = (0.5, 0.2, 0.5, 0.2)$, using the inverses $J^{\#P}$ and $J^{\#E}$. In the extended Jacobian inverse we set $a = 100$. The performance of the inverse kinematics algorithms is visualized in the figures. Specifically, Figures 8–11 display the convergence of the Jacobian pseudo-inverse and of the extended Jacobian inverse in the control space. Figures 12 and 13 demonstrate the paths of the chained-form system in the (q_1, q_2)-plane.

5 Conclusions

We have addressed the synthesis problem of repeatable inverse kinematics algorithms for robotic manipulators. Special attention has been paid to the approximation of the Jacobian pseudo-inverse by an extended Jacobian inverse. The problem

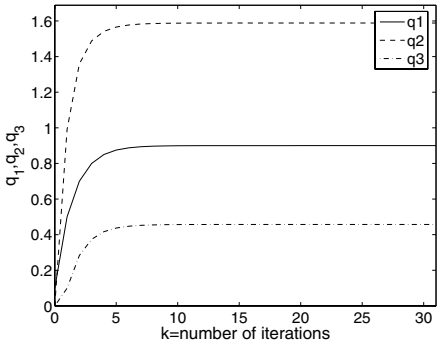


Fig. 2 Joint space trajectories for $J^{\#P}$.

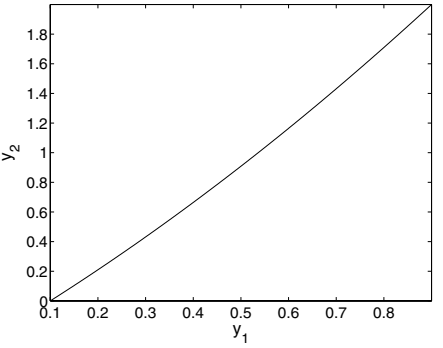


Fig. 3 End effector path for $J^{\#P}$.

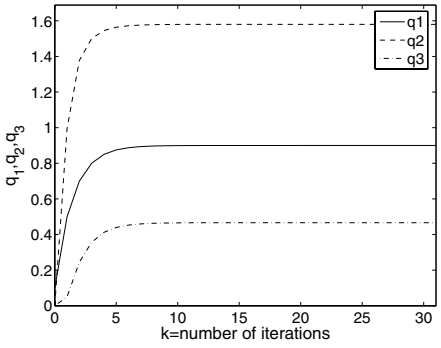


Fig. 4 Joint space trajectories for $J^{\#E1}$.

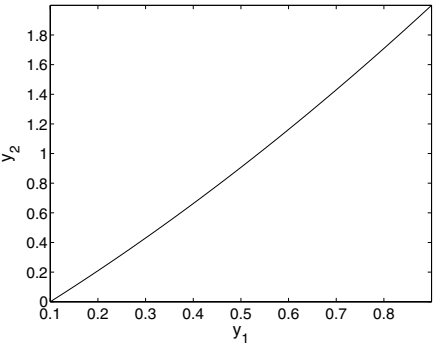


Fig. 5 End effector path for $J^{\#E1}$.

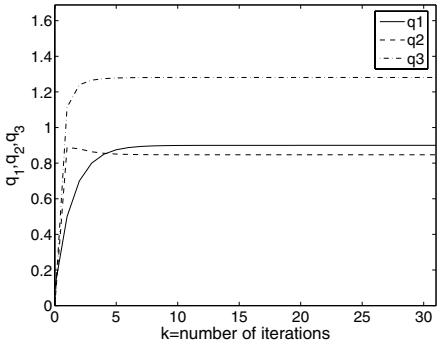


Fig. 6 Joint space trajectories for $J^{\#E2}$.

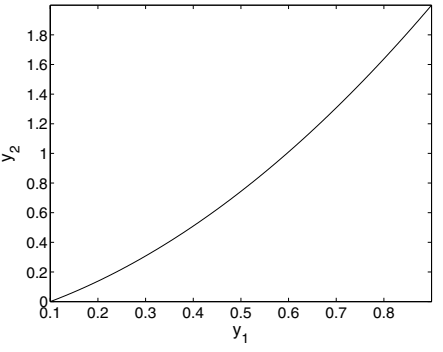


Fig. 7 End effector path for $J^{\#E2}$.

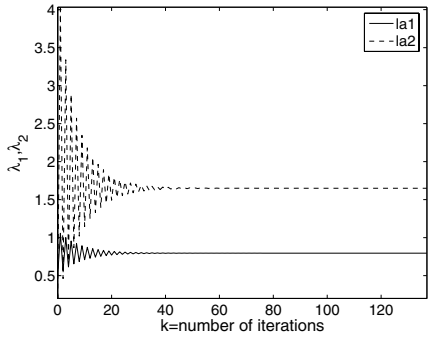


Fig. 8 Controls λ_1, λ_2 for $J^{\#P}$.

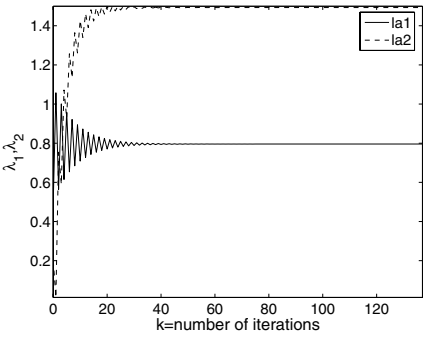


Fig. 9 Controls λ_1, λ_2 for $J^{\#E}$.

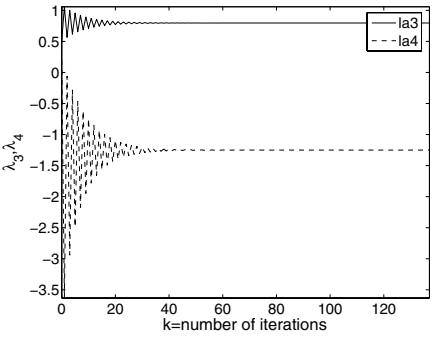


Fig. 10 Controls λ_3, λ_4 for $J^{\#P}$.

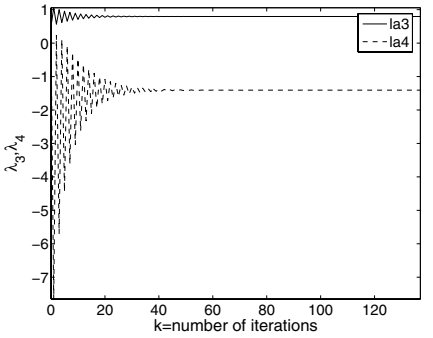


Fig. 11 Controls λ_3, λ_4 for $J^{\#E}$.

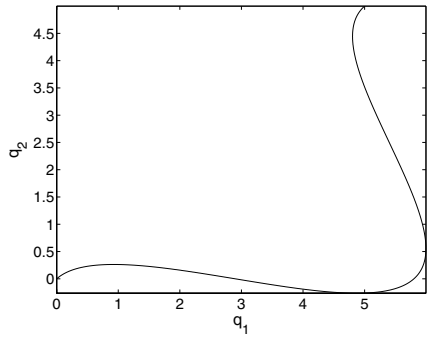


Fig. 12 Task space path of $J^{\#P}$.

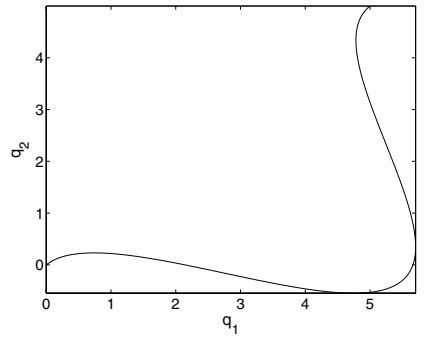


Fig. 13 Task space path of $J^{\#E}$.

has been formulated in differential geometric terms as the approximation problem of a codistribution associated with the pseudo-inverse by an integrable codistribution. The approximation guarantees that the approximating and the approximated codistributions will agree on leaves of a certain foliation of the configuration space and along a characteristic direction. These requirements translate into a linear partial differential equation whose characteristic equation provides the augmenting kinematics map defining the extended Jacobian inverse. This approach has been illustrated with the derivation of extended Jacobian inverse kinematics algorithms for a stationary manipulator and a chained-form system.

Acknowledgments

This research was supported in part by the Foundation for Polish Science, and in part by a statutory grant.

References

1. Chitour, Y., and Sussmann, H. J. (1998), Motion planning using the continuation method, In: *Essays on Mathematical Robotics*, Springer-Verlag, New York, pp. 91–125.
2. Davidenko, D. F. (1953), On a new method of numerically integrating a system of nonlinear equations, *Dokl. Akad. Nauk SSSR*, vol. 88, pp. 601–603.
3. Klein, C. A., and Huang, C. H. (1983), Review of pseudo-inverse control for use with kinematically redundant manipulators, *IEEE Trans. Syst., Man, Cybern*., vol. 13, no 3, pp. 245–250.
4. Richter, S. L., and DeCarlo, R. A. (1983), Continuation methods: Theory and applications, *IEEE Trans. Circuits Syst.*, vol. 30, pp. 347–352.
5. Roberts, R. G., and Maciejewski, A. A. (1992), Nearest optimal repeatable control strategies for kinematically redundant manipulators, *IEEE Trans. Robot. Autom.*, vol. 8, no. 3, pp. 327–337.
6. Roberts, R. G., and Maciejewski, A. A. (1993), Repeatable generalized inverse control strategies for kinematically redundant manipulators, *IEEE Trans. Autom. Control,* vol. 38, no. 5, pp. 689–699.
7. Sluis, W. M., Banaszuk, A., Hauser, J., and Murray, R. M. (1996), A homotopy algorithm for approximating geometric distributions by integrable systems, *Syst. Control Lett.,* vol. 27, pp. 285-291.
8. Shamir, T., and Yomdin, Y. (1988), Repeatability of redundant manipulators: Mathematical solution of the problem, *IEEE Trans. Automat. Contr.*, vol. 33, no. 11, pp. 1001–1009.
9. Tchoń, K. (2002), Repeatability of inverse kinematics algorithms for mobile manipulators, *IEEE Trans. Automat. Contr*., vol. 47, no. 8, pp. 1376–1380.
10. Tchoń, K., Janiak, M., Mianowski, K., and Małek, Ł. (2006), Optimal, repeatable inverse kinematics algorithm for robotic manipulators, In: *Progress in Robotics: Control, Perception and Communication*, Warsaw, WKL Publisher, pp. 141–150 (in Polish).
11. Ważewski, T. (1947), Sur l'évaluation du domaine d'existence des fonctions implicites réelles ou complexes, *Ann. Soc. Pol. Math.*, vol. 20, pp. 81–120.

A Screw Syzygy with Applications to Robot Singularity Computation

J.M. Selig[1] and Peter Donelan[2]

[1]*Faculty of Business, Computing and Information Management, London South Bank University, London SE1 0AA, U.K. e-mail: seligjm@lsbu.ac.uk*
[2]*School of Mathematics, Statistics and Computer Science, Victoria University of Wellington, Wellington 6140, New Zealand, e-mail: peter.donelan@vuw.ac.nz*

Abstract. A syzygy is a relation between invariants. In this paper a syzygy is presented between invariants of sequences of six screws under the action of the Euclidean group. This relation is useful in simplifying the computation of the determinant of a robot Jacobian and hence can be used to investigate the singularities of robot manipulators.

Key words: Jacobians, determinants, singularities.

1 Introduction

Methods for determining singularities of serial and parallel manipulators generally involve a combination of exact or symbolic algebraic manipulation, to obtain as simple a form for the Jacobian or its determinant as possible, together with algorithms from linear algebra and numerical analysis to obtain good approximate solutions. The extent to which the former methods are effective depends on properties of the Euclidean geometry that describes the motion of the rigid components of manipulators and on the specific architectural features of a given manipulator, such as partitioning or symmetry.

In this paper we derive a new relation or syzygy among the Euclidean invariants of sequences of screws that leads to a simplification in the calculation of the Jacobian determinant for 6-dof manipulators. This is illustrated by some specific applications.

2 Isometries and Invariants

The motion of a rigid link, in particular the end-effector of a robot manipulator or platform relative to its home configuration, is described mathematically by a distance and orientation-preserving transformation of 3-space: a proper Euclidean iso-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 147–154.

metry. The collection of these isometries forms a Lie group, denoted $SE(3)$. For a given choice of coordinate system such an isometry is represented by a rotation about the origin together with a translation and so $SE(3)$ is a product of the rotation group $SO(3)$ and the translation group $\mathbb{R}^3$, though the composition of two Euclidean isometries is not component-wise or direct but 'semi-direct'. There is a 6×6 matrix representation of an isometry:

$$H = \begin{pmatrix} R & 0 \\ TR & R \end{pmatrix}, \tag{1}$$

where R is a 3×3 rotation matrix, T a 3×3 skew-symmetric 'translation' matrix and the composition of two isometries is given by matrix multiplication. For details see for example Selig (2005).

If we chose a different initial coordinate system in 3-space then there would be a different but related matrix representation of a given isometry. Specifically, if the change of coordinates itself is represented by $A \in SE(3)$ then H transforms to $H' = AHA^{-1}$; H and H' are said to be congruent.

Considering just rotations for the moment, an infinitesimal rotation at the home configuration is a skew-symmetric matrix

$$\Omega = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix} \tag{2}$$

and these form a 3-dimensional space. Once again an orthogonal change of coordinates A (fixing the origin) results in a transformation of Ω to $\Omega' = A\Omega A^{-1}$. This is known as the *adjoint action* of the rotation group $SO(3)$. This action is exactly the same as the effect of transforming the 3-vector $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)^T$ by the orthogonal transformation A. In particular, invariants of the adjoint action of $SO(3)$ are simply invariants of its ordinary action on $\mathbb{R}^3$. Hence, writing Q for the 3×3 identity matrix, the quadratic form $\boldsymbol{\omega}^T Q \boldsymbol{\omega}$ is an invariant of the adjoint action, while $\boldsymbol{\omega}_1^T Q \boldsymbol{\omega}_2$ is an invariant of pairs of infinitesimal rotations. In the theory of Lie groups, the quadratic form $\boldsymbol{\omega}^T Q \boldsymbol{\omega}$ is called the *Killing form* of the adjoint action.

Generalising to $SE(3)$, the infinitesimal isometries now form a 6-dimensional space. Its elements are combinations of infinitesimal rotations Ω (2) and infinitesimal translations $\mathbf{v}$. The infinitesimal isometries are *twists*, though frequently referred to as *screws* which, properly, are the collections of all non-zero multiples of a given (non-zero) twist. However, we use the common term 'screws' here for the 6-vectors $\mathbf{s} = (\boldsymbol{\omega}^T, \mathbf{v}^T)^T$. The adjoint action of $SE(3)$ on its Lie algebra is given by the matrix multiplication:

$$\begin{pmatrix} R & 0 \\ TR & R \end{pmatrix} \begin{pmatrix} \boldsymbol{\omega} \\ \mathbf{v} \end{pmatrix}. \tag{3}$$

The Killing form is given by $\mathbf{s}_1^T Q_\infty \mathbf{s}_2$ where Q_∞ is the 6×6 symmetric matrix given below. It is degenerate, having only rank 3. There is a second, non-degenerate, invariant bilinear form for the adjoint action of $SE(3)$, namely the *reciprocal product*

$\mathbf{s}_1^T Q_0 \mathbf{s}_2$:

$$Q_0 = \left(\begin{array}{c|c} 0 & I_3 \\ \hline I_3 & 0 \end{array}\right), \qquad Q_\infty = \left(\begin{array}{c|c} 2I_3 & 0 \\ \hline 0 & 0 \end{array}\right).$$

Unlike the Killing form Q for $SO(3)$ however the reciprocal product, regarded as a quadratic form, the *Klein form*, is not positive definite. The ratio of the two quadratic forms is the *pitch* of a screw:

$$p(\mathbf{s}) = \mathbf{s}^T Q_0 \mathbf{s} / \mathbf{s}^T Q_\infty \mathbf{s}.$$

The set of screws for which $\mathbf{s}^T Q_\infty \mathbf{s} \neq 0$ but $p(\mathbf{s}) = \mathbf{s}^T Q_0 \mathbf{s} = 0$ can be identified with the set of lines in 3-space, the corresponding line being the axis of the screw. The coordinates of $\boldsymbol{\omega}, \mathbf{v}$ are the Plücker coordinates of the line. The quadric hypersurface $\mathbf{s}^T Q_0 \mathbf{s} = 0$ (projectively, Klein's quadric) includes the subspace $\mathbf{s}^T Q_\infty \mathbf{s} = 0$ of screws of pitch infinity, i.e. infinitesimal translations. Pairs of screws for which $\mathbf{s}_1^T Q_0 \mathbf{s}_2 = 0$ are called *reciprocal*. From the manipulator point of view, screws of pitch zero and infinity correspond to the motions associated with revolute and prismatic joints respectively.

3 The Syzygy

As motivation, we begin by looking at a syzygy among invariants of the group of rotations $SO(3)$. Weyl (1946, theorem 2.9.A) asserts that the invariants of the standard representation of $SO(3)$ (i.e. its action on vectors in $\mathbb{R}^3$) are generated by the scalar product of pairs of vectors and the scalar triple product of triples of vectors.

There is one form of syzygy between these invariants – that is to say, they are not *algebraically* independent. Consider three arbitrary vectors $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3 \in \mathbb{R}^3$ and write them as columns of a matrix 3×3 matrix

$$M = \left(\mathbf{a}_1 \middle| \mathbf{a}_2 \middle| \mathbf{a}_3\right).$$

Then $\det M = \mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3)$ is a basic invariant. We have

$$M^T M = \begin{pmatrix} \mathbf{a}_1 \cdot \mathbf{a}_1 & \mathbf{a}_1 \cdot \mathbf{a}_2 & \mathbf{a}_1 \cdot \mathbf{a}_3 \\ \mathbf{a}_2 \cdot \mathbf{a}_1 & \mathbf{a}_2 \cdot \mathbf{a}_2 & \mathbf{a}_2 \cdot \mathbf{a}_3 \\ \mathbf{a}_3 \cdot \mathbf{a}_1 & \mathbf{a}_3 \cdot \mathbf{a}_2 & \mathbf{a}_3 \cdot \mathbf{a}_3 \end{pmatrix} \tag{4}$$

and the matrix entries are themselves invariants. In the terminology of Section 2, the left-hand side could be written as $M^T Q M$. Taking the determinant of both sides of (4) gives the syzygy

$$\det(M)^2 = \det\left(\mathbf{a}_i \cdot \mathbf{a}_j\right),$$

representing a polynomial identity between the basic invariants, of degree 6 in the coordinates of the vectors. Given the identification of the standard representation

with the adjoint representation for $SO(3)$, we may also think of these as invariants and a syzygy among invariants of the adjoint action.

By analogy we now exhibit an $SE(3)$ syzygy. Begin with six screws $\mathbf{s}_1, \ldots, \mathbf{s}_6$. The reciprocal product $\mathbf{s}_i^T Q_0 \mathbf{s}_j$ of any pair is certainly invariant. Combine the screws into the 6×6 matrix

$$J = \left(\mathbf{s}_1 \middle| \mathbf{s}_2 \middle| \mathbf{s}_3 \middle| \mathbf{s}_4 \middle| \mathbf{s}_5 \middle| \mathbf{s}_6 \right).$$

The adjoint action (3) of $SE(3)$ on the individual screws gives rise to an action on J and since $\det R = 1$, the determinant of J is also an invariant. Now form the product

$$J^T Q_0 J = \begin{pmatrix} \mathbf{s}_1^T Q_0 \mathbf{s}_1 & \mathbf{s}_1^T Q_0 \mathbf{s}_2 & \mathbf{s}_1^T Q_0 \mathbf{s}_3 & \mathbf{s}_1^T Q_0 \mathbf{s}_4 & \mathbf{s}_1^T Q_0 \mathbf{s}_5 & \mathbf{s}_1^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_2^T Q_0 \mathbf{s}_1 & \mathbf{s}_2^T Q_0 \mathbf{s}_2 & \mathbf{s}_2^T Q_0 \mathbf{s}_3 & \mathbf{s}_2^T Q_0 \mathbf{s}_4 & \mathbf{s}_2^T Q_0 \mathbf{s}_5 & \mathbf{s}_2^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_3^T Q_0 \mathbf{s}_1 & \mathbf{s}_3^T Q_0 \mathbf{s}_2 & \mathbf{s}_3^T Q_0 \mathbf{s}_3 & \mathbf{s}_3^T Q_0 \mathbf{s}_4 & \mathbf{s}_3^T Q_0 \mathbf{s}_5 & \mathbf{s}_3^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_4^T Q_0 \mathbf{s}_1 & \mathbf{s}_4^T Q_0 \mathbf{s}_2 & \mathbf{s}_4^T Q_0 \mathbf{s}_3 & \mathbf{s}_4^T Q_0 \mathbf{s}_4 & \mathbf{s}_4^T Q_0 \mathbf{s}_5 & \mathbf{s}_4^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_5^T Q_0 \mathbf{s}_1 & \mathbf{s}_5^T Q_0 \mathbf{s}_2 & \mathbf{s}_5^T Q_0 \mathbf{s}_3 & \mathbf{s}_5^T Q_0 \mathbf{s}_4 & \mathbf{s}_5^T Q_0 \mathbf{s}_5 & \mathbf{s}_5^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_6^T Q_0 \mathbf{s}_1 & \mathbf{s}_6^T Q_0 \mathbf{s}_2 & \mathbf{s}_6^T Q_0 \mathbf{s}_3 & \mathbf{s}_6^T Q_0 \mathbf{s}_4 & \mathbf{s}_6^T Q_0 \mathbf{s}_5 & \mathbf{s}_6^T Q_0 \mathbf{s}_6 \end{pmatrix}. \quad (5)$$

Each entry in the matrix is invariant, so taking determinants of each side of (5) gives a syzygy of degree 12 in the screw coordinates:

$$\det(J)^2 = -\det\left(\mathbf{s}_i^T Q_0 \mathbf{s}_j\right), \quad (6)$$

the negative sign coming from the fact that $\det(Q_0) = -1$.

4 Applications

The key point is that the syzygy (6) enables us to find the determinant of a manipulator Jacobian, at least up to sign, by finding the determinant of a *symmetric* matrix. In particular, singularity detection, that is checking whether $\det J = 0$, is feasible. In general this leads to faster computation. Moreover in many particular cases the matrix in (5) has a nice form, reflecting the intrinsic geometry of the manipulator.

4.1 Fast Computation

In general the determinant of a positive definite symmetric matrix can be found using Cholesky decomposition, this algorithm is known to be about twice as fast as LU decomposition which would be the appropriate method for a non-symmetric matrix, see Press et al. (1992, section 2.9). Unfortunately the matrix of reciprocal products is not necessarily positive definite so another algorithm must be used, for

example LDLT decomposition (Bunch and Parlett, 1971), which is still twice as fast as LU decomposition.

A simple count of the number of arithmetic operations involved in a symbolic algebraic expression of the determinant, using Maple say, gives 3600 multiplications for a general 6×6 determinant but only 2252 for a symmetric matrix. Moreover, if the joints are revolute or prismatic, that is if the screws satisfy $\mathbf{s}_i^T Q_0 \mathbf{s}_i = 0$, $i = 1, \ldots, 6$, then the diagonals in the matrix in (5) are all zero and the number of multiplications is reduced to 765. Similarly, additions are also reduced. Of course, the matrix of reciprocal products has to be determined first, but given the simplicity of Q_0, that involves only six multiplications per entry so 126 in all, by symmetry.

4.2 Wrist-Partitioned Serial Manipulators

In many cases, the matrix of reciprocal products has a special form, further simplifying calculation of the Jacobian determinant and hence robot singularities. Consider the singularities of a wrist-partitioned serial 6R robot (see, for example, Stanišić and Engelbert, 1988; Hayes et al., 2002). Not only are the computations accomplished easily using the syzygy, but the approach yields a simple proof of the types of singularity that can occur. Notice that it is not even necessary to set up a particular coordinate frame.

Since the last three joints of the robot comprise a wrist, the joint axes all meet at the wrist centre. This means that the last three screws are mutually reciprocal and hence the bottom right-hand corner of the matrix of reciprocal products is zero. Since the top right and bottom left corners of the matrix are the transpose of each other the expansion of the determinant leads to

$$\det(J) = \pm \det \begin{pmatrix} \mathbf{s}_1^T Q_0 \mathbf{s}_4 & \mathbf{s}_1^T Q_0 \mathbf{s}_5 & \mathbf{s}_1^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_2^T Q_0 \mathbf{s}_4 & \mathbf{s}_2^T Q_0 \mathbf{s}_5 & \mathbf{s}_2^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_3^T Q_0 \mathbf{s}_4 & \mathbf{s}_3^T Q_0 \mathbf{s}_5 & \mathbf{s}_3^T Q_0 \mathbf{s}_6 \end{pmatrix}.$$

If we are only interested in singularities the sign ambiguity is not important. Setting this 3×3 determinant to zero means that the columns (or rows) of the matrix must be linearly dependent. Column dependence can be expressed by the equations

$$\begin{aligned} \lambda \mathbf{s}_1^T Q_0 \mathbf{s}_4 + \mu \mathbf{s}_1^T Q_0 \mathbf{s}_5 + \nu \mathbf{s}_1^T Q_0 \mathbf{s}_6 &= 0, \\ \lambda \mathbf{s}_2^T Q_0 \mathbf{s}_4 + \mu \mathbf{s}_2^T Q_0 \mathbf{s}_5 + \nu \mathbf{s}_2^T Q_0 \mathbf{s}_6 &= 0, \\ \lambda \mathbf{s}_3^T Q_0 \mathbf{s}_4 + \mu \mathbf{s}_3^T Q_0 \mathbf{s}_5 + \nu \mathbf{s}_3^T Q_0 \mathbf{s}_6 &= 0, \end{aligned} \tag{7}$$

for some non-zero constants λ, μ and ν. These equations can be solved in two ways. First, if the last three joints are linearly dependent, that is

$$\lambda \mathbf{s}_4 + \mu \mathbf{s}_5 + \nu \mathbf{s}_6 = \mathbf{0}$$

for some λ, μ, ν, then clearly (7) is satisfied. This occurs if and only the three screw axes $\mathbf{s}_4$, $\mathbf{s}_5$ and $\mathbf{s}_6$ are coplanar and constitutes a *wrist singularity*. Likewise, if the rows of the matrix are linearly dependent then for some α, β, γ not all zero

$$\begin{aligned}
\alpha\mathbf{s}_1^T Q_0\mathbf{s}_4 + \beta\mathbf{s}_2^T Q_0\mathbf{s}_4 + \gamma\mathbf{s}_3^T Q_0\mathbf{s}_4 &= 0,\\
\alpha\mathbf{s}_1^T Q_0\mathbf{s}_5 + \beta\mathbf{s}_2^T Q_0\mathbf{s}_5 + \gamma\mathbf{s}_3^T Q_0\mathbf{s}_5 &= 0,\\
\alpha\mathbf{s}_1^T Q_0\mathbf{s}_6 + \beta\mathbf{s}_2^T Q_0\mathbf{s}_6 + \gamma\mathbf{s}_3^T Q_0\mathbf{s}_6 &= 0,
\end{aligned} \tag{8}$$

and the same reasoning leads to a *shoulder singularity* of the first three joints when they fail to span a 3-dimensional subspace. Whether or not such a singularity can occur depends on the screw systems determined by the first three joints and so, in turn, on the design parameters of the shoulder.

The second kind of solution is most easily derived from Equations (8) which are satisfied if and only if there is a screw $\alpha\mathbf{s}_1 + \beta\mathbf{s}_2 + \gamma\mathbf{s}_3$ reciprocal to each of $\mathbf{s}_4$, $\mathbf{s}_5$ and $\mathbf{s}_6$, and hence to all screws in the screw system $\mathcal{S}$ spanned by the last three joints. Since their axes intersect in the wrist centre, $\mathcal{S}$ is a type IIA ($p = 0$) 3-system, which is self-reciprocal (Gibson and Hunt, 1990). Hence the screw $\alpha\mathbf{s}_1 + \beta\mathbf{s}_2 + \gamma\mathbf{s}_3$ lies in $\mathcal{S}$ so it must also be a line through the wrist centre. In general, the $\mathbf{s}_1$, $\mathbf{s}_2$ and $\mathbf{s}_3$ form a 3-system corresponding to a projective 2-plane which will intersect the Klein quadric in a conic. Each point on the conic represents a line in the screw system and it is well known that such a set of lines form the regulus of a hyperboloid. If any of these lines passes through the wrist centre then the configuration of the robot is singular. In some cases the only possibilities are when the wrist centre lies on the first or second joint axis.

However, there are some more complex situations. Consider the PUMA manipulator where the first and second joint axes meet and the second and third joint axes are parallel. In this case, the conic in the screw system degenerates into a pair of lines. One of these corresponds to the lines in 3-space parallel to the second and third joint axes, the other to the set of lines meeting both the first and second joint axes. The PUMA then has singularities if its wrist centre lies in either the plane containing the first and second joints or the plane containing the second and third joints.

4.3 Serial Manipulators with Self-Reciprocal Subassemblies

In the previous example an important point was the fact that a subassembly of three consecutive joints form a self-reciprocal screw system. Such systems are necessarily 3-systems and must consist only of lines. The fact that the system is self-reciprocal ensures that the matrix of reciprocal products contains a diagonal 3×3 block of zeros. Besides the IIA ($p = 0$) 3-systems met in the example above, there are two other such systems: IIC ($p = 0$) systems and the IID system. These three screw systems are also Lie subalgebras; this is important because it means that the screw

system is invariant under motions of the manipulator and hence the the matrix of reciprocal products will contain a block of zeros in every position of the manipulator. There is another 3-system which is a subalgebra, the IIC ($p \neq 0$) system, but it is not self-reciprocal.

The IID system contains only screws of infinite pitch, that is infinitesimal translations. Cartesian-type and gantry robots contain subassemblies of prismatic joints which generate this screw system. However their singularities are already easy to analyse and we do not pursue this case.

The IIC ($p = 0$) systems can be generated by three parallel lines. Hence a robot containing three consecutive parallel revolute joints satisfies this requirement. Such designs are rare in commercially available robots at the moment. Examples are the original Cincinnati Milacron T3 and the Telequipment MA2000 in which joints 2, 3 and 4 are parallel. Expanding the determinant of the matrix of reciprocal products gives

$$\det(J) = \pm \det \begin{pmatrix} \mathbf{s}_1^T Q_0 \mathbf{s}_2 & \mathbf{s}_5^T Q_0 \mathbf{s}_2 & \mathbf{s}_6^T Q_0 \mathbf{s}_2 \\ \mathbf{s}_1^T Q_0 \mathbf{s}_3 & \mathbf{s}_5^T Q_0 \mathbf{s}_3 & \mathbf{s}_6^T Q_0 \mathbf{s}_3 \\ \mathbf{s}_1^T Q_0 \mathbf{s}_4 & \mathbf{s}_5^T Q_0 \mathbf{s}_4 & \mathbf{s}_6^T Q_0 \mathbf{s}_4 \end{pmatrix}.$$

By similar arguments to Section 4.2 we have three types of singularity. First, when $\mathbf{s}_2, \mathbf{s}_3$ and $\mathbf{s}_4$ are linearly dependent, the three parallel joints must be coplanar and the robot has an *elbow singularity*. Second, it may be possible for the other three joints $\mathbf{s}_1$, $\mathbf{s}_5$ and $\mathbf{s}_6$ to be linearly dependent, though this will depend on the disposition of these joints. Thirdly, the robot will be singular if a screw from the 3-system determined by $\mathbf{s}_1$, $\mathbf{s}_5$ and $\mathbf{s}_6$ lies in the screw system determined by the three parallel joint axes. For example this can happen in the T3 design if the last joint is parallel to joints 2, 3 and 4.

The IIC ($p = 0$) systems can also be generated by a revolute joint and two prismatic joints or two parallel revolute joints and a prismatic joint. These do not seem to correspond to subassemblies of any design of robot that has been proposed.

4.4 Gough–Stewart Platforms

For Gough–Stewart platforms it is well known that the rows of the Jacobian matrix can be found from the lines joining the passive spherical joints (Merlet, 2000). Indeed if $\mathbf{s}_i$ is the unit line joining the passive spherical joints on the ith leg then the corresponding row in the Jacobian matrix will be $\mathbf{s}_i^T Q_0$. Writing K for this Jacobian and using $Q_0^2 = I$, the syzygy gives

$$-\det(K)^2 = \det(\mathbf{s}_i^T Q_0 \mathbf{s}_j).$$

Since the screws $\mathbf{s}_i$ are lines, the diagonal of the matrix of reciprocal products will consist of zeros. In the 3–3 design pairs of legs meet at the passive spherical joints alternately on the base and the platform. In this way there are just three passive

spherical joints on each. In this case consecutive leg axes are reciprocal, so that the matrix of reciprocal products has the form

$$K Q_0 K^T = \begin{pmatrix} 0 & 0 & \mathbf{s}_1^T Q_0 \mathbf{s}_3 & \mathbf{s}_1^T Q_0 \mathbf{s}_4 & \mathbf{s}_1^T Q_0 \mathbf{s}_5 & 0 \\ 0 & 0 & 0 & \mathbf{s}_2^T Q_0 \mathbf{s}_4 & \mathbf{s}_2^T Q_0 \mathbf{s}_5 & \mathbf{s}_2^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_3^T Q_0 \mathbf{s}_1 & 0 & 0 & 0 & \mathbf{s}_3^T Q_0 \mathbf{s}_5 & \mathbf{s}_3^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_4^T Q_0 \mathbf{s}_1 & \mathbf{s}_4^T Q_0 \mathbf{s}_2 & 0 & 0 & 0 & \mathbf{s}_4^T Q_0 \mathbf{s}_6 \\ \mathbf{s}_5^T Q_0 \mathbf{s}_1 & \mathbf{s}_5^T Q_0 \mathbf{s}_2 & \mathbf{s}_5^T Q_0 \mathbf{s}_3 & 0 & 0 & 0 \\ 0 & \mathbf{s}_6^T Q_0 \mathbf{s}_2 & \mathbf{s}_6^T Q_0 \mathbf{s}_3 & \mathbf{s}_6^T Q_0 \mathbf{s}_4 & 0 & 0 \end{pmatrix}.$$

The zero-structure of this matrix renders the determinant relatively easy to compute.

5 Conclusions

We have presented a new syzygy among invariants of sets of screws. Using a couple of examples we have shown how this relation enables easy determination and analysis of robot manipulator singularities.

References

Bunch, J.R. and Parlett, B.N. (1971) Direct methods for solving symmetric indefinite systems of linear equations. *SIAM J. Numerical Analysis* **8**(4), 639–655.

Gibson, C.G. and Hunt, K.H. (1990) Geometry of screw systems I & II. *Mechanism and Machine Theory* **25**(1), 1–27.

Hayes, M., Husty, M. and Zsombor-Murray, P.J. (2002) Singular configurations of wrist-partitioned 6R serial robots: A geometric perspective for users. *Transactions of the CSME* **26**(1), 41–55.

Merlet, J-P. (2000) *Parallel Robots*. Kluwer, Dordrecht.

Press, W.H., Teukolsky, S.A., Vetterling W.T. and Flannery B.P. (1992) *Numerical Recipes in C*, 2nd ed. Cambridge University Press, Cambridge U.K.

Stanišić, M.M. and Engelberth, J.W. (1988) A geometric description of manipulator singularities in terms of singular surfaces. In: *Proceedings 1st International Workshop on Advances in Robot Kinematics*, Ljubljana, Slovenia, pp. 132–141.

Selig, J.M. (2005) *Geometric Fundamentals of Robotics*. Springer, New York.

Weyl, H. (1946) *The Classical Groups*. Princeton University Press, Princeton NJ.

Singularity Robust Jacobian Inverse Kinematics for Mobile Manipulators

Krzysztof Tchoń and Łukasz Małek

Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology, ul. Janiszewskiego 11/17, 50-372 Wrocław, Poland, e-mail: {krzysztof.tchon, lukasz.malek}@pwr.wroc.pl

Abstract. The paper is devoted to the singularity robust Jacobian inverse kinematics algorithm for mobile manipulators. The endogenous configuration space approach is assumed as a guideline. The main contribution of the paper consists in establishing completeness of this algorithm, and in stating its convergence condition in terms of the mobile manipulator's dexterity matrix. Computer simulations illustrate the performance and the convergence of the algorithm.

Key words: mobile manipulator, Jacobian inverse kinematics, singularity robust algorithm, performance, convergence.

1 Introduction

A mobile manipulator is a mobile platform equipped with a stationary on-board manipulator. In this paper we shall assume that the platform's motion is subject to nonholonomic constraints, while the on-board manipulator is holonomic. The notion of the mobile manipulator covers both the stationary manipulator (when the platform is immobilized) as well as the mobile platform (when the on-board manipulator is dismantled). Taking into account the properties of the platform and of the on-board manipulator, it is natural to adopt as a model of kinematics of the mobile manipulator a parametrized driftless control system with outputs. A convenient framework for the analysis of mobile manipulators has been provided by the the endogenous configuration space approach [7]. According to this approach, the mobile manipulator's kinematics are defined as the end point map of the driftless control system. The configurations of the mobile manipulators are identified with control actions exerted on the system. The linear approximation of the system determines the mobile manipulator's Jacobian. The Jacobian allows for distinguishing regular and singular configurations. The inverse kinematic problem is formulated as a control problem with prescribed time horizon. The Jacobian inverse kinematics algorithms are derived using the continuation method, that leads to the Ważewski–Davidenko differential equation [1,10]. A solution of the inverse kinematic problem is obtained as a limit point at infinity of the trajectory of a dynamic system associ-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 155–164.

ated with the Ważewski–Davidenko equation. A derivation and a performance study of several Jacobian inverse kinematics algorithms for mobile manipulators can be found in [7–9].

There are two difficulties arising in connection with the Jacobian inverse kinematics algorithms. The first of them results from a deterioration of the algorithm's performance in a vicinity of singular configurations. The second, even more fundamental, follows from the fact that trajectories of the associated dynamic system may exist on finite time intervals depending on initial conditions, what means that generally there is no guarantee that the limit points can be achieved. If the trajectories can be extended to infinity, independently of initial conditions, both the associated dynamic system as well as the inverse kinematics algorithm, will be called complete. In the literature concerned with motion planning of nonholonomic systems several specific results can be found establishing the completeness of the Jacobian pseudo-inverse algorithm for the unicycle and kinematic car platforms, and for the rolling ball [2–4].

In this paper we examine the singularity robust Jacobian inverse kinematics algorithm for mobile manipulators, defined by analogy with its counterpart for stationary manipulators, see [5]. By definition, the singularity robust algorithm operates at regular and singular configurations. Our main result consists in establishing completeness of this algorithm, and in providing a condition of its convergence. Contrary to the results for the Jacobian pseudo-inverse reported in the literature, our result is independent of the particular form of the kinematics equations.

The composition of this paper is the following. In Section 2 we remind necessary concepts of the endogenous configuration space approach. Section 3 introduces the singularity robust Jacobian inverse, and presents the main result. Section 4 is devoted to computer simulations. The paper concludes with Section 5.

2 Basic Concepts

As we have already mentioned, the kinematics of a mobile manipulator is modeled by the following parametrized driftless control system with outputs

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

In the system (1) the variables $q \in R^n$, $x \in R^p$ and $y \in R^r$ denote, respectively, the generalized platform coordinates, the joint positions of the on-board manipulator, and the task space coordinates. The dynamics of the control system come from the nonholonomic motion constraints imposed on the platform, the output map defines the end effector location in the task space in terms of the platform and manipulator variables. The steering actions applicable to the system (1) include the platform control functions $u(t)$ and the manipulator's joint positions $x \in R^p$ playing the role of parameters; both these actions will be referred to as the endogenous configura-

tion $(u(\cdot), x) \in \mathcal{X}$ of the mobile manipulator. The endogenous configuration space $\mathcal{X} \cong L^2_m[0,T] \times R^p$ is an infinite dimensional Hilbert space equipped with the inner product

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

where T is the control time horizon, and $R(t)$ and W denote symmetric, positive definite weight matrices. The form of the inner product reflects the decomposition of the mobile manipulator into the mobile platform and the on-board manipulator. A choice of the weight matrices depends on the type of the mobile manipulator or even on the type of its task; in particular these matrices should provide a proper scaling and compensate for inconsistencies of measure units of physically different quantities. The task space is a Euclidean space. To any initial platform coordinates $q_0 \in R^n$ and an endogenous configuration $(u(\cdot), x) \in \mathcal{X}$ there corresponds a platform trajectory $q(t) = \varphi_{q_0,t}(u(\cdot))$, and a task space trajectory $y(t) = k(q(t), x)$.

The kinematics of the mobile manipulator are defined by means of the end point map of the control system (1)

$$K_{q_0,T}(u(\cdot), x) = y(T) = k(\varphi_{q_0,T}(u(\cdot)), x). \tag{3}$$

The associated infinitesimal kinematics are represented by the Jacobian

$$J_{q_0,T}(u(\cdot), x)(v(\cdot), w) = C(T,x) \int_0^T \Phi(T,s) B(s) v(s) ds + D(T,x) w. \tag{4}$$

The Jacobian is computed using the linear approximation of the system (1) along $(u(t), x, q(t) = \varphi_{q_0,t}(u(\cdot))$, therefore

$$A(t) = \tfrac{\partial (G(q(t))u(t))}{\partial q},\ B(t) = G(q(t)),\ C(t,x) = \tfrac{\partial k(q(t),x)}{\partial q},\ D(t,x) = \tfrac{\partial k(q(t),x)}{\partial x}.$$

The matrix $\Phi(t,s)$ is the transition matrix satisfying the evolution equation

$$\frac{\partial}{\partial t} \Phi(t,s) = A(t) \Phi(t,s), \quad \Phi(s,s) = I_n.$$

Given the Jacobian, and a direction of motion η in the task space, we consider a Jacobian equation

$$J_{q_0,T}(u(\cdot), x)(v(\cdot), w) = \eta \tag{5}$$

that should be solved for $(v(\cdot), w)$. The Jacobian equation is solvable, if the configuration $(u(\cdot), x)$ is regular, what means full rank of the dexterity matrix

$$\mathcal{D}_{q_0,T}(u(\cdot), x) = D(T,x) W^{-1} D^T(T,x) + C(T,x) \int_0^T \Phi(T,s) B(s) R^{-1}(s) B^T(s) \Phi^T(T,s) ds\ C^T(T,x). \tag{6}$$

Whenever $\operatorname{rank}\mathcal{D}_{q_0,T}(u(\cdot),x) < r$, the configuration $(u(\cdot),x)$ is singular.

Having introduced the kinematics (3), we shall address the inverse kinematic problem: given the kinematics and a desirable point y_d in the task space, find a configuration $(u_d(\cdot),x_d)$ such that $K_{q_0,T}(u_d(\cdot),x_d) = y_d$. In control theoretic terminology the inverse kinematic problem amounts to defining a control $u_d(t)$ and a parameter x_d in the control system (1) initialized at q_0, that will drive its output at T to $y(T) = y_d$. The inverse kinematic problem for mobile manipulators is usually solved numerically, by means of Jacobian inverse kinematics algorithms.

A convenient way for deriving the Jacobian algorithms is offered by the continuation method [3,6]. In accordance with this method, we choose a differentiable curve $(u_\theta(\cdot),x(\theta))$ passing through an initial configuration $(u_0(\cdot),x_0) \in \mathcal{X}$, compute the task space error

$$e(\theta) = K_{q_0,T}\big(u_\theta(\cdot),x(\theta)\big) - y_d, \tag{7}$$

and require that the error obeys a differential equation

$$\frac{d\,e(\theta)}{d\theta} = -\gamma\, S(\theta)e(\theta). \tag{8}$$

The constant $\gamma > 0$ and the matrix $S(\theta)$ will ensure an asymptotic convergence of the error to 0. A differentiation of the error formula (7) produces a Ważewski–Davidenko equation

$$J_{q_0,T}\big(u_\theta(\cdot),x(\theta)\big)\frac{d}{d\theta}\big(u_\theta(\cdot),x(\theta)\big) = -\gamma\, S(\theta)e(\theta). \tag{9}$$

To proceed, let us take a map $J^{\#}_{q_0,T}(u(\cdot),x) : R^r \to \mathcal{X}$, and set

$$\frac{d}{d\theta}\big(u_\theta(\cdot),x(\theta)\big) = -\gamma\, J^{\#}_{q_0,T}\big(u_\theta(\cdot),x(\theta)\big)e(\theta). \tag{10}$$

A substitution of (10) into (9) shows that

$$S(\theta) = J_{q_0,T}\big(u_\theta(\cdot),x(\theta)\big)J^{\#}_{q_0,T}\big(u_\theta(\cdot),x(\theta)\big). \tag{11}$$

Now, if the matrix $S(\theta)$ renders the error (8) vanishing, the map $J^{\#}_{q_0,T}(u(\cdot),x)$ will be called a dynamic inverse of the Jacobian. If this is the case, the system (10) defines a Jacobian inverse kinematics algorithm. A solution of the inverse kinematic problem is computed as a limit at $\theta \to +\infty$ of the trajectory of (10). For example, by taking as $J^{\#}_{q_0,T}(u(\cdot),x)$ the Jacobian pseudo-inverse, we obtain the Jacobian pseudo-inverse inverse kinematics algorithm converging exponentially with the rate γ.

The design procedure of the inverse kinematics algorithms based on the dynamic system (10) may be impaired by two main defects. The first defect consists in the presence of kinematic singularities, the second results from the incompleteness of the system (10). The lack of completeness causes that the system trajectories may not extend to all $\theta \geq 0$, so the asymptotic behavior of the algorithm cannot be

defined. In the next section we shall present a Jacobian inverse that is free from both these defects.

3 Singularity Robust Inverse

Similarly as for stationary manipulators [5], the singularity robust Jacobian inverse for mobile manipulators is defined as a solution to the Jacobian equation (5) associated with the following optimization problem

$$\min_{(v(\cdot),w)} \big(\kappa \|(v(\cdot),w)\|^2_{RW} + \|J_{q_0,T}(u(\cdot),x)(v(\cdot),w) - \eta\|^2\big), \tag{12}$$

where $\kappa > 0$ is a parameter, $\|\cdot\|_{RW}$ denotes the norm induced by the inner product (2) in the endogenous configuration space, and $\|\cdot\|$ is the standard Euclidean norm. The objective function in (12) may be viewed either as a result of an aggregation of a two-component vector optimization problem or, when κ is small, as a perturbation of the objective function defining the Jacobian pseudo-inverse operator. Being a solution of (12), the singularity robust Jacobian inverse operator

$$J^{\#SRI}_{q_0,T}(u(\cdot),x) : R^r \longrightarrow \mathcal{X}$$

assumes the form [7]

$$\big(J^{\#SRI}_{q_0,T}(u(\cdot),x)\eta\big)(t) = \big[R^{-1}(t)B^T(t)\Phi^T(T,t)C^T(T,x),\ W^{-1}D^T(T,x)\big] \times \big(\kappa I_r + \mathcal{D}_{q_0,T}(u(\cdot),x)\big)^{-1}\eta. \tag{13}$$

By definition, the operator (13) does not suffer from the presence of singular configurations. Plugged into the algorithm (10), this operator ensures the local existence of trajectories of the dynamic system

$$\frac{d}{d\theta}\big(u_\theta(\cdot),x(\theta)\big) = -\gamma J^{\#SRI}_{q_0,T}\big(u_\theta(\cdot),x(\theta)\big)e(\theta). \tag{14}$$

Suppose temporarily that a trajectory $(u_\theta(t),x(\theta))$ of (14) exists for any $\theta \geq 0$, and compute the matrix (11)

$$S(\theta) = \big(u_\theta(\cdot),x(\theta)\big) = \mathcal{D}_{q_0,T}\big(u_\theta(\cdot),x(\theta)\big)\big(\kappa I_r + \mathcal{D}_{q_0,T}(u_\theta(\cdot),x(\theta))\big)^{-1}.$$

By the Ważewski inequality [11] the error (7) is bounded in the following way

$$\|e(0)\|\exp\left(\int_0^\theta \underline{\lambda}_{M(s)}ds\right) \leq \|e(\theta)\| \leq \|e(0)\|\exp\left(\int_0^\theta \overline{\lambda}_{M(s)}ds\right) \tag{15}$$

where $M(\theta) = -\frac{1}{2}\gamma(S(\theta) + S^T(\theta))$, and $\overline{\lambda}$, $\underline{\lambda}$ denote, respectively, the maximum and minimum eigenvalue of a symmetric matrix. The matrix $S(\theta)$ appears to be

symmetric, so $M(\theta) = -\gamma S(\theta)$, and using (15), it can be shown that the singularity robust Jacobian inverse kinematics algorithm converges, whenever the aforementioned integral diverges

$$\lim_{\theta\to+\infty}\int_0^{\theta}\frac{\underline{\lambda}_{\mathcal{D}_{q_0,T}(u_\alpha(\cdot),x(\alpha))}d\alpha}{\kappa+\underline{\lambda}_{\mathcal{D}_{q_0,T}(u_\alpha(\cdot),x(\alpha))}}=+\infty. \tag{16}$$

Now, we shall show that all trajectories generated by the singularity robust inverse algorithm indeed exist for every θ, i.e. the algorithm is complete. The operator norm of (13) can be computed in the following way

$$\|J^{\#SRI}_{q_0,T}(u(\cdot),x)\| = \sup_{\eta\neq 0}\frac{\|J^{\#SRI}_{q_0,T}(u(\cdot),x)\eta\|_{RW}}{\|\eta\|} = \overline{\lambda}^{1/2}_{(\kappa I_r+\mathcal{D}_{q_0,T}(u(\cdot),x))^{-2}\mathcal{D}_{q_0,T}(u(\cdot),x))}. \tag{17}$$

Furthermore, if $\lambda_{\mathcal{D}_{q_0,T}(u(\cdot),x)}$ denotes an eigenvalue of the dexterity matrix, then we observe that

$$\lambda_{(\kappa I_r+\mathcal{D}_{q_0,T}(u(\cdot),x))^{-2}\mathcal{D}_{q_0,T}(u(\cdot),x))} = \frac{\lambda_{\mathcal{D}_{q_0,T}(u(\cdot),x)}}{(\kappa+\lambda_{\mathcal{D}_{q_0,T}(u(\cdot),x)})^2}. \tag{18}$$

The right hand side of (18) has the form of the function $f(x) = x/(\kappa+x)^2$ for $x \geq 0$. It is easily checked that at $x = \kappa$ this function reaches its maximum equal to $1/4\kappa$. All these observations allow us to make a conclusion that the norm (17) is upper bounded, i.e.

$$\|J^{\#SRI}_{q_0,T}(u(\cdot),x)\| \leq \frac{1}{2}\kappa^{-1/2}. \tag{19}$$

The local existence of trajectories of the system (14), the boundedness of the error (15), and the bound (19) imply the completeness of the algorithm.

We conclude that the singularity robust Jacobian inverse kinematics algorithm operates both at regular and singular configurations, and is complete. The algorithm converges on condition (16) that is satisfied, if the trajectory $(u_\theta(\cdot), x(\theta))$ of the dynamic system (14) stays within the set of singular configurations only on a finite interval of the variable θ.

4 Computer Simulations

In this section we shall present results of applying the singularity robust Jacobian inverse kinematics algorithm to a mobile manipulator composed of kinematic car platform and an RTR on-board manipulator, shown in Figure 1. The variable $q = (\mathsf{x}, \mathsf{y}, \varphi, \psi) \in R^4$ describes the platform position, orientation and heading angle of its front wheels. Variables $x = (x_1, x_2, x_3) \in R^3$ and $y = (y_1, y_2, y_3) \in R^3$ refer to the joint and to the end effector position of the on-board manipulator. The lateral slip of the platform's wheels is not permitted. The control system (1) takes the following

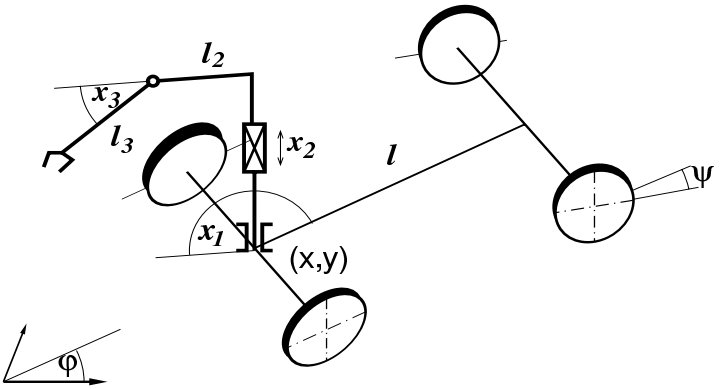


Fig. 1 Mobile manipulator.

Table 1 Largest task space errors.

q_0	$(10,0,\frac{\pi}{2},0)$	$(10,10,\frac{\pi}{4},\frac{\pi}{4})$	$(0,10,0,0)$	$(0,10,\frac{\pi}{2},\frac{\pi}{4})$
$\max\|e(\theta)\|$	$6.485424E-04$	$9.186227E-08$	$1.470058E-04$	$1.627936E-05$
q_0	$(0,0,0,\frac{\pi}{4})$	$(0,0,\frac{\pi}{2},0)$	$(10,10,-\frac{\pi}{4},0)$	$(-10,10,\frac{\pi}{4},-\frac{\pi}{4})$
$\max\|e(\theta)\|$	$1.991798E-13$	$2.092102E-13$	$2.877973E-07$	$4.256074E-05$

form

$$\begin{cases} \dot{q}_1 = u_1 \cos q_3 \cos q_4, \quad \dot{q}_2 = u_1 \sin q_3 \cos q_4, \quad \dot{q}_3 = u_1 \sin q_4, \quad \dot{q}_4 = u_2 \\ y = (q_1 + L\cos(x_1+q_3), q_2 + L\sin(x_1+q_3), x_2 + l_3 \sin x_3), \end{cases}$$

where $L = l_2 + l_3 \cos x_3$. In simulations the unit geometric dimensions $l = l_2 = l_3 = 1$ of the mobile manipulator were assumed. The inverse kinematic problems consisted in reaching the point $y_d = (0,0,0)$ from 8 initial platform's states q_0. Initial configuration of the on-board manipulator was set to $x_0 = (1,1,1)$, and the time horizon $T = 1$. The control $u(t)$ was devised from a 500 element basis consisting of piecewise constant functions. For each q_0 100 simulations of (14) were run, starting from a randomly generated $u_0(\cdot)$. Parameters of the algorithm $\gamma = 0.01$ and $\kappa = 10^{-6}$ were adopted. In each simulation 3000 steps of the algorithm were executed. In all these experiments the algorithm has converged. The results are collected in Table 1 that shows the largest task space errors for diverse q_0.

As can be seen from the table, the largest errors are generally acceptably small. Moreover, in all cases the shape of the platform path appears to be satisfactory and the error converges exponentially, as can be seen in Figures 2 and 3 corresponding to $q_0 = (0,10,\pi/2,0)$ and $u_0 = (1,1)$. On the other hand, the observed rate of convergence depends significantly on the initial control $u_0(\cdot)$ and the initial platform coordinates q_0. In simulations we have noticed a remarkable robustness of the singularity robust algorithm against kinematic singularities. To confront this aspect of the algorithm's performance with that of the Jacobian pseudo-inverse, two other simulations were run for the kinematic car type mobile platform shown

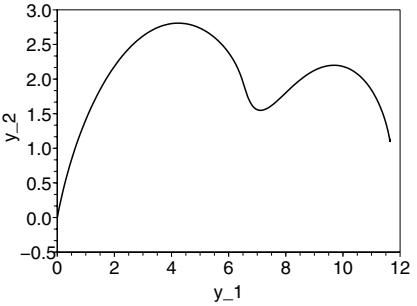


Fig. 2 Singularity robust inverse: end effector coordinates $(y_1(t), y_2(t))$.

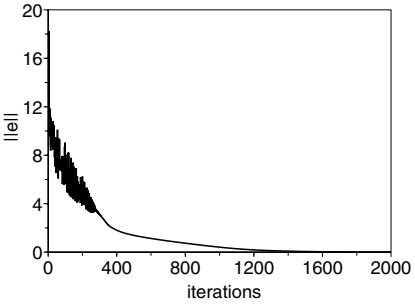


Fig. 3 Singularity robust inverse: end effector error.

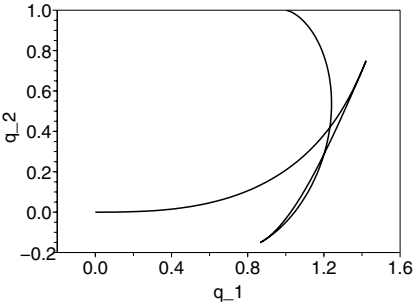


Fig. 4 Singularity robust inverse: platform path $(q_1(t), q_2(t))$.

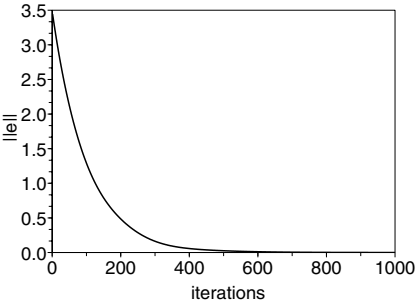


Fig. 5 Singularity robust inverse: end effector error.

in Figure 1 (the on-board manipulator dismantled). The initial control was chosen close to the singular control $u_0 = (0.001, 1)$. The initial platform coordinates were taken as $q_0 = (1, 1, 3.14, 1.57)$, the algorithm's parameters $\gamma = 0.01$ and $\kappa = 10^{-6}$. The control basis included 200 elements. In simulations 1000 iterations were allowed. Figures 4 and 5 demonstrate an economic manoeuvre and an exponential convergence of the singularity robust algorithm, opposed to the erratic performance and the lack of convergence of the Jacobian pseudo-inverse algorithm visualized in Figures 6 and 7.

5 Conclusions

We have demonstrated completeness of the singularity robust Jacobian inverse kinematics algorithm for mobile manipulators, and derived a condition for its convergence. Computer simulations have confirmed that the algorithm converges inde-

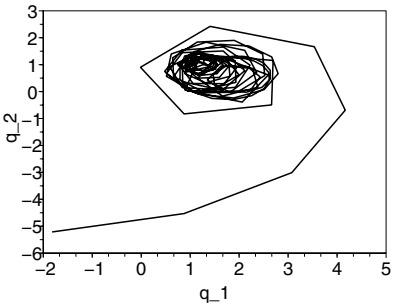


Fig. 6 Jacobian pseudo-inverse: platform path $(q_1(t), q_2(t))$.

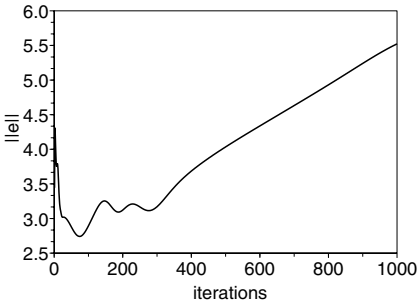


Fig. 7 Jacobian pseudo-inverse: end effector error.

pendently of the initial configuration, performs satisfactorily, and shows considerable robustness with respect to kinematic singularities.

Acknowledgment

This research was supported by the Foundation for Polish Science.

References

1. Davidenko, D.F. (1953), On a new method of numerically integrating a system of nonlinear equations, *Dokl. Akad. Nauk SSSR* **88**, 601–603.
2. Chelouah, A. and Chitour, Y. (2003), On the motion planning of rolling surfaces, *Forum Math.* **15**, 727–758.
3. Chitour, Y. and Sussmann, H.J. (1998), Motion planning using the continuation method, in *Essays on Mathematical Robotics*, J. Baillieul et al. (Eds.), Springer-Verlag, New York, pp. 91–125.
4. Chitour, Y. (2006), A homotopy continuation method for trajectories generation of nonholonomic systems, *ESAIM: Control, Optim. Calc. Var.* **12**, pp. 139-168.
5. Nakamura, Y. (1991), *Advanced Robotics: Redundancy and Optimization*, Addison-Wesley, Reading.
6. Richter, S.L. and DeCarlo, R.A. (1983), Continuation methods: Theory and applications, *IEEE Trans. Circuits Syst.* **30**, 347–352.
7. Tchoń, K. and Jakubiak, J. (2003), Endogenous configuration space approach to mobile manipulators: A derivation and performance assessment of Jacobian inverse kinematics algorithms, *Int. J. Control* **76**, 1387–1419.
8. Tchoń, K. and Jakubiak, J. (2005), A repeatable inverse kinematics algorithm with linear invariant subspaces for mobile manipulators, *IEEE Trans. Syst., Man, Cybern. B, Cybern.* **35**(5), 1051–1057.
9. Tchoń, K. (2006), Repeatable, extended Jacobian inverse kinematics algorithm for mobile manipulators, *Syst. Control Lett.* **55**, 87–93.

10. Ważewski, T. (1947), Sur l'évaluation du domaine d'existence des fonctions implicites réelles ou complexes, *Ann. Soc. Pol. Math.* **20**, 81–120.
11. Ważewski, T. (1948), Sur la limitation des intégrales des systemes d'équations différentielles linéaires ordinaires, *Studia Math.* **10**, 48–59.

Robots Based on Assur Group A (3.5)

Karl Wohlhart

Institute for Mechanics, Graz University of Technology, 8010 Graz, Austria,
e-mail: wohlhart@tugraz.at

Abstract. The paper presents position analyses of open normal Assur groups A (3.5). Planar mechanisms can be considered to be composed of link groups (Assur groups) with zero mobility relative to the links to which they are successively added. These Assur groups, which serve as modules in the synthesis and analysis of complex planar mechanisms, might adopt a certain number of positions which allow choosing different solutions for the engineering task. An open normal Assur group, for which we write A (3.5) in short, is an open kinematic chain of ternary links to which 5 binary links (legs) are attached. It is found that for a given set of system parameters an open normal Assur group A (3.5) held together exclusively by rotor joints might theoretically adopt a maximum of 54 different positions (real and complex). If the input rotor joints at the 5 legs of this Assur group are successively exchanged by 1, 2, 3, 4, or 5 prismatic input joints, then the number of its possible positions becomes 54, 46, 28, 16 or 8. An open Assur group A (3.5) might serve as a basic mechanism for a parallel redundant planar robot.

Key words: Assur groups, position analysis, Sylvester's elimination method.

1 Introduction

According to the definition given in the Terminology for the Theory of Machines and Mechanisms [1], an Assur group is, the "smallest kinematic chain which when added to, or subtracted from, a mechanism results in mechanism that has the same mobility as the original". In 1914 and 1915 the Russian scientist Assur published two great articles [2] in which he proposed a structural classification of planar linkages with lower kinematic pairs (rotor joints or prismatic joints). Assur mainly distinguishes two special types of what, according to Shukowsky, we now call the open and the closed normal chains. The open normal chain consists of n ternary and $n+2$ binary links interconnected by $2n+1$ rotary or prismatic joints. Via the $n+2$ (rotary or prismatic) half joints at the outer end of the binary links the normal open Assur group can be brought into contact with any other mechanism provided that at its contact points there are complementary half joints. If the Assur group is mounted onto a rigid body the assembly is a rigid structure. With the number of bodies $b = 2n+3$ and the number of joints $j = 3(n+1)$ admitting only one relative degree

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 165–175.

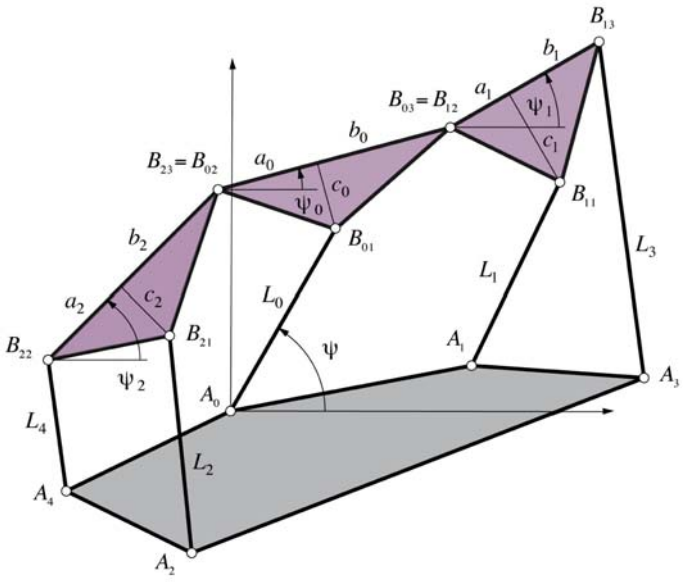


Fig. 1 Assur group A (3.5) mounted on a rigid body by five R-joints.

of freedom ($\Sigma m_i = j$) for the mobility of the mounted open normal Assur group, the topological formula of Grübler–Kutzbach yields: $m = \Sigma m_i - 3(j - b + 1) = 0$. In 1977, Artobolevski [3] proposed a nomenclature for Assur groups, according to which an open kinematic chain consisting of ternary bodies to which k binary links are attached is an Assur group of class 3 (as the greatest closed polygon has three sides); and it is of order k as the number of guiding joints or inputs equals the number of attached binary links. In short, it is an Assur group A (3.k). The first position analysis of the Assur group A (3.3) was given by Paisach in 1985 [4]. In 1993, Innocenti presented the position analysis of the Assur group A (3.4) [5]. In 1995, Lösch [6] published a position analysis of the higher Assur groups A (3.5), A (3.6) and A (4.4) on the basis of his "floating point Gröbner method". Recently it has been shown that the position analysis of a special Assur group simultaneously solves the position analysis of a number of other general (i.e. not normal) Assur groups: in [7] the authors derive from a nine-link Barranow truss nine Assur groups (with an equal number of assembly configurations) and numerically determine the possible positions for a given set of system parameters. The main subject of the present paper is the position analysis of the general normal Assur group A (3.5) whose input joints are partly prismatic joints. These general Assur groups have never been analyzed.

2 Position Analysis of the Assur group A (3.5)

Figure 1 shows an Assur group A (3.5) mounted onto a rigid body by rotor joints at the points A_α. Its system parameters are given with the lengths $L_\alpha = A_\alpha B_{0\alpha}$, the heights c_α and the segments a_α, b_α of the triangular links, the index α runs from 0 to 5. The position of the Assur group within its fixation is given with the four angles ψ, ψ_0, ψ_1 and ψ_2. The position analysis aims to find all sets of these four angles.

With

$$\begin{aligned} e_{01} &= \{\cos\psi_0, \sin\psi_0\}, \quad e_{02} = \{-\sin\psi_0, \cos\psi_0\}, \\ e_{11} &= \{\cos\psi_1, \sin\psi_1\}, \quad e_{12} = \{-\sin\psi_1, \cos\psi_1\}, \\ e_{21} &= \{\cos\psi_2, \sin\psi_2\}, \quad e_{22} = \{-\sin\psi_2, \cos\psi_2\}, \end{aligned}$$

the vertices of the ternary links are given by:

$$\begin{aligned} &B_{01} = L_0\{\cos\psi, \sin\psi\}, && B_{02} = B_{01} - e_{01}a_0 + e_{02}c_0, && B_{03} = B_{02} + e_{01}(a_0 + b_0), \\ &B_{11} = B_{03} + e_{11}a_1 - e_{12}c_1, && B_{12} = B_{03}, && B_{13} = B_{03} + e_{11}(a_1 + b_1), \\ &B_{21} = B_{02} - e_{21}b_2 - e_{22}c_2, && B_{22} = B_{02} + e_{21}(a_2 + b_2), && B_{23} = B_{02}. \end{aligned} \tag{1}$$

The four basic equations which determine the four position angles read:

$$\begin{aligned} &G_1 = (B_{11} - A_1)\cdot(B_{11} - A_1) - L_1^2 = 0, && G_2 = (B_{21} - A_2)\cdot(B_{21} - A_2) - L_2^2 = 0, \\ &G_3 = (B_{13} - A_3)\cdot(B_{13} - A_3) - L_3^2 = 0, && G_4 = (B_{22} - A_4)\cdot(B_{22} - A_4) - L_4^2 = 0. \end{aligned} \tag{2}$$

With

$$\begin{aligned} \sin\psi &= 2X/(1 + X^2), \quad \cos\psi = (1 - X^2)/(1 + X^2), \\ \sin\psi_0 &= 2Y/(1 + Y^2), \quad \cos\psi = (1 - Y^2)/(1 + Y^2), \\ \sin\psi_1 &= 2x/(1 + x^2), \quad \cos\psi_1 = (1 - x^2)/(1 + x^2), \\ \sin\psi_2 &= 2y/(1 + y^2), \quad \cos\psi_2 = (1 - y^2)/(1 + y^2), \end{aligned} \tag{3}$$

we obtain the four algebraic basic equations in X, Y, x and y

$$P_1(X, Y, x) = 0, \quad P_2(X, Y, x) = 0, \quad P_3(X, Y, y) = 0, \quad P_4(X, Y, y) = 0, \tag{4}$$

which we are going to solve with Sylvester's elimination method [9]. With the coefficients in the two series expansions (which are functions only of X and Y)

$$\begin{aligned} P_1 &= p_{11}x^2 + p_{12}x + p_{13} = 0, \quad P_2 = p_{21}x^2 + p_{22}x + p_{23} = 0, \\ P_3 &= p_{31}y^2 + p_{32}y + p_{33} = 0, \quad P_4 = p_{41}y^2 + p_{42}y + p_{43} = 0, \end{aligned} \tag{5}$$

we can define the following two matrices or two vector matrices

$$\mathbf{U} = \begin{bmatrix} 0 & p_{11} & p_{12} & p_{13} \\ p_{11} & p_{12} & p_{13} & 0 \\ 0 & p_{21} & p_{22} & p_{23} \\ p_{21} & p_{22} & p_{23} & 0 \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} 0 & p_{31} & p_{32} & p_{33} \\ p_{31} & p_{32} & p_{33} & 0 \\ 0 & p_{41} & p_{42} & p_{43} \\ p_{41} & p_{42} & p_{43} & 0 \end{bmatrix}, \tag{6}$$

$$\mathbf{x} = \{x^3, x^2, x, 1\}^T, \quad \mathbf{y} = \{y^3, y^2, y, 1\}^T. \tag{7}$$

The equations $P_1 = 0$, $P_2 = 0$ or $P_3 = 0$, $P_4 = 0$ can therewith be written as matrix equations

$$\mathbf{U}\mathbf{x} = 0, \quad \mathbf{V}\mathbf{y} = 0, \tag{8}$$

from which we obtain the following two equations in X and Y

$$F_1(X, Y) = \mathrm{Det}\,\mathbf{U} = 0, \quad F_2(X, Y) = \mathrm{Det}\,\mathbf{V} = 0. \tag{9}$$

Leaving aside the factor $(1 + X^2)(1 + Y^2)$, the series expansions of these two equations in Y read

$$\begin{aligned} F_1(X, Y) &= f_{11}Y^6 + f_{12}Y^5 + f_{13}Y^4 + f_{14}Y^3 + f_{15}Y^2 + f_{16}Y + f_{17} = 0, \\ F_2(X, Y) &= f_{21}Y^6 + f_{22}Y^5 + f_{23}Y^4 + f_{24}Y^3 + f_{25}Y^2 + f_{26}Y + f_{27} = 0, \end{aligned} \tag{10}$$

where the coefficients $f_{1\alpha}$ and $f_{2\alpha}$ are polynomials of order six in X. With the Sylvester matrix $\mathbf{S}$ and the vector matrix $\mathbf{Y}$

$$\mathbf{S} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & f_{11} & f_{12} & f_{13} & f_{14} & f_{15} & f_{16} & f_{17} \\ 0 & 0 & 0 & 0 & f_{11} & f_{12} & f_{13} & f_{14} & f_{15} & f_{16} & f_{17} & 0 \\ 0 & 0 & 0 & f_{11} & f_{12} & f_{13} & f_{14} & f_{15} & f_{16} & f_{17} & 0 & 0 \\ 0 & 0 & f_{11} & f_{12} & f_{13} & f_{14} & f_{15} & f_{16} & f_{17} & 0 & 0 & 0 \\ 0 & f_{11} & f_{12} & f_{13} & f_{14} & f_{15} & f_{16} & f_{17} & 0 & 0 & 0 & 0 \\ f_{11} & f_{12} & f_{13} & f_{14} & f_{15} & f_{16} & f_{17} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & f_{21} & f_{22} & f_{23} & f_{24} & f_{25} & f_{26} & f_{27} \\ 0 & 0 & 0 & 0 & f_{21} & f_{22} & f_{23} & f_{24} & f_{25} & f_{26} & f_{27} & 0 \\ 0 & 0 & 0 & f_{21} & f_{22} & f_{23} & f_{24} & f_{25} & f_{26} & f_{27} & 0 & 0 \\ 0 & 0 & f_{21} & f_{22} & f_{23} & f_{24} & f_{25} & f_{26} & f_{27} & 0 & 0 & 0 \\ 0 & f_{21} & f_{22} & f_{23} & f_{24} & f_{25} & f_{26} & f_{27} & 0 & 0 & 0 & 0 \\ f_{21} & f_{22} & f_{23} & f_{24} & f_{25} & f_{26} & f_{27} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{11}$$

$$\mathbf{Y} = \{Y^{11}, Y^{10}, Y^9, Y^8, Y^7, Y^6, Y^5, Y^4, Y^3, Y^2, Y, 1\}^T, \tag{12}$$

the equations $F_1 = 0$ and $F_2 = 0$ can be written in the form of a single matrix equation:

$$\mathbf{S}(X)\mathbf{Y} = 0, \tag{13}$$

whose coefficient determinant must vanish. The equation $\mathrm{Det}\,[\mathbf{S}(X)] = 0$ yields a polynomial of order 72 which can be factorized: the first factor reads $(1 + X^2)^9$ and

the second is a polynomial $F(X)$ of order 54

$$F(X) = \sum_{1}^{54} a_k X^k = 0, \tag{14}$$

which can be solved numerically, rendering 54 roots X_α (real or complex) for X. To find the corresponding values Y_α we can proceed as follows. By canceling the first row in the matrix $\mathbf{S}(X_\alpha)$ we obtain a matrix $\mathbf{S}^\bullet = \{\mathbf{s}_1^\bullet, \mathbf{s}_2^\bullet, \ldots, \mathbf{s}_{11}^\bullet, \mathbf{s}_{12}^\bullet\}$ and from $\mathbf{S}^\bullet(X_\alpha)\mathbf{Y} = 0$ we deduce:

$$Y_\alpha = \mathrm{Det}\,\{\mathbf{s}_1^\bullet, \mathbf{s}_2^\bullet, \ldots, \mathbf{s}_{10}^\bullet, -\mathbf{s}_{12}^\bullet\}/\mathrm{Det}\,\{\mathbf{s}_1^\bullet, \mathbf{s}_2^\bullet, \ldots, \mathbf{s}_{10}^\bullet, \mathbf{s}_{11}^\bullet\}. \tag{15}$$

With X_α and Y_α the matrices $\mathbf{U}(X_\alpha, \mathbf{Y}_\alpha)$ and $\mathbf{V}(X_\alpha, \mathbf{Y}_\alpha)$ are given. By canceling the first row in the matrix $\mathbf{U}(X_\alpha, \mathbf{Y}_\alpha)$ or $\mathbf{V}(X_\alpha, \mathbf{Y}_\alpha)$ we obtain a matrix $\mathbf{U}^\bullet = \{\mathbf{u}_1^\bullet, \mathbf{u}_2^\bullet, \mathbf{u}_3^\bullet, \mathbf{u}_4^\bullet\}$ or $\mathbf{V}^\bullet = \{\mathbf{v}_1^\bullet, \mathbf{v}_2^\bullet, \mathbf{v}_3^\bullet, \mathbf{v}_4^\bullet\}$ and find for the corresponding values of x_α or y_α:

$$\begin{aligned} x_\alpha &= \mathrm{Det}\,\{\mathbf{u}_1^\bullet, \mathbf{u}_2^\bullet, -\mathbf{u}_4^\bullet\}/\mathrm{Det}\,\{\mathbf{u}_1^\bullet, \mathbf{u}_2^\bullet, \mathbf{u}_3^\bullet\}, \\ y_\alpha &= \mathrm{Det}\,\{\mathbf{v}_1^\bullet, \mathbf{v}_2^\bullet, -\mathbf{v}_4^\bullet\}/\mathrm{Det}\,\{\mathbf{v}_1^\bullet, \mathbf{v}_2^\bullet, \mathbf{v}_3^\bullet\}. \end{aligned} \tag{16}$$

The angles determining the positions of the A (3.5) are then given by

$$\{\psi, \psi_0, \psi_1, \psi_2\}_\alpha = 2\arctan\{X, Y, x, y\}_\alpha, \quad \alpha = (1 \div 54). \tag{17}$$

2.1 Case Study A (3.5)

An Assur group A (3.5) with the system parameters p, L given by

$$p = \{a_0 = 3, b_0 = 4, c_0 = 2\},\ \{a_1 = 2, b_1 = 3, c_1 = 3\},\ \{a_2 = 3, b_2 = 4, c_2 = 2\},$$

$$L = \{L_0 = 6, L_1 = 8.072, L_2 = 9.207, L_3 = 8.364, L_4 = 6.049\},$$

is mounted with five R-joints at the following fixed points:

$$P = \{A_0 = \{0, 0\}, A_1 = \{5, 11\},\ A_2 = \{-5, 10\},\ A_3 = \{10, 6\},\ A{-}4 = \{-7, 1\}\}.$$

Of the 54 positions we found in the process of the calculations described in Section 2, we only report the twelve real positions,which are listed in Table 1 and shown in Figure 2.

The fact that there are theoretically 54 positions of the Assur group A (3.5) with five R input joints can be deduced from Wunderlich's statements about the order and circularity of higher coupler curves [8]. If the R-joint at point B_{03} in Figure 1 is open, point B_{12} describes a coupler curve of order $n_1 = 6$ and circularity $c_1 = 3$, and point B_{03} describes a coupler curve of order $n_2 = 2 \times 3^2 = 18$ and circularity

Table 1 The 12 real solutions of the position angles for an Assur group A (3.5).

ψ	ψ_0	ψ_1	ψ_2
79.6552	–118.717	81.5456	30.3308
95.4418	–101.195	78.0196	64.1477
151.98	–23.2615	58.9456	35.0558
24.0422	–44.4756	–158.456	11.5488
57.2965	–45.8389	–114.597	57.2915
98.5112	53.7393	–69.1771	112.209
112.068	69.7956	–17.2945	–6.39256
117.216	64.2246	14.6902	124.093
68.6197	155.92	33.0685	–20.4594
39.9024	97.6492	–86.675	–38.656
–6.61052	25.6593	–125.032	–46.8777
–16.01	5.46725	–166.24	–44.3627

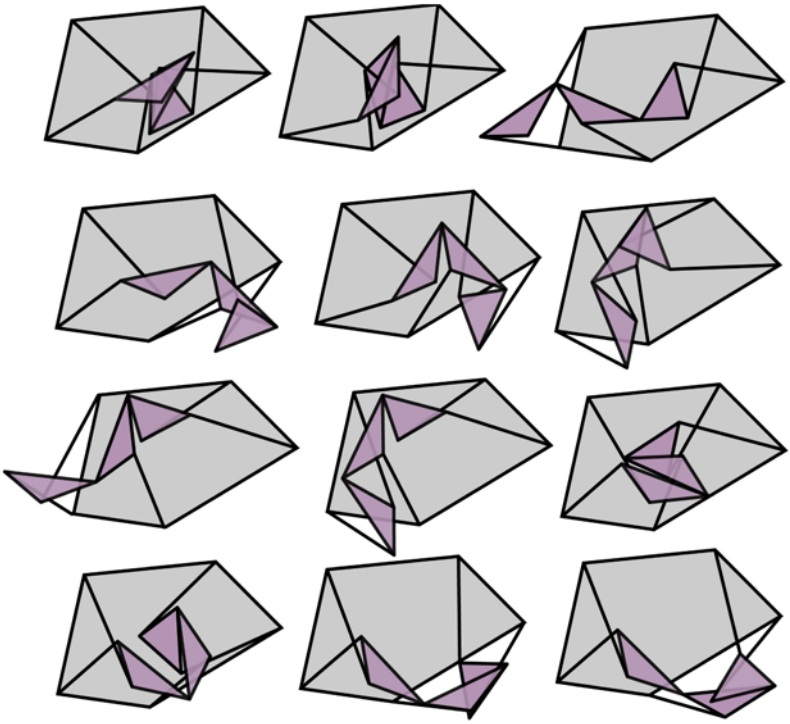


Fig. 2 The 12 positions of a normal Assur group A (3.5) with five R input joints.

$c_2 = 9$, and these curves intersect in $n_1 \times n_2 - 2(c_1 \times c_2) = 54$ points. Therefore 54 positions of the Assur group A (3.5) are possible.

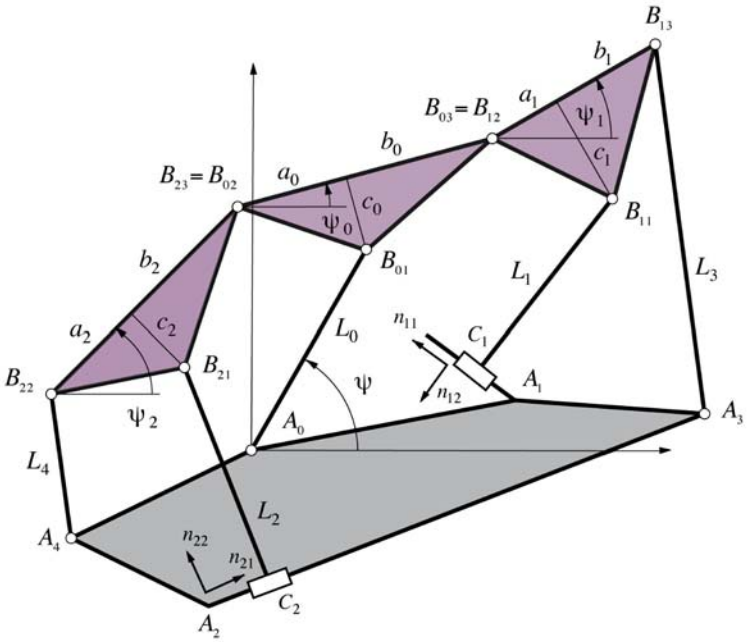


Fig. 3 Assur group A (3.5)2P mounted on a rigid body with two input P-joints.

3 Position Analysis of the Assur group A (3.5)*nP*

If of the five input joints n R-joints are exchanged by n P-joints, then we shall write for the Assur group: A(3.5)$n\,P$. In Figure 3 an Assur group with two input P-joints is shown: . The position of the Assur group within its fixation at the rigid body is again given with ψ, ψ_0, ψ_1 and ψ_2 as long as the R-joint at A_0 is not exchanged for a P-joint, and even the scheme of calculation is the same as above.

3.1 Case Studies A (3.5)nP: n = 1, 2, 3, 4

For these four cases we assume the same system parameters p and the same points of attachment P as in Section 2.1. For the lengths of the legs we put successively

$$L = \{L_0 = 6, L_1 = 7.997, L_2 = 9.207, L_3 = 8.364, L_4 = 6.049\},$$

$$L = \{L_0 = 6, L_1 = 7.997, L_2 = 9.198, L_3 = 8.364, L_4 = 6.049\},$$

$$L = \{L_0 = 6, L_1 = 7.997, L_2 = 9.198, L_3 = 8.349, L_4 = 6.049\},$$

$$L = \{L_0 = 6, L_1 = 7.997, L_2 = 9.198, L_3 = 8.349, L_4 = 5.804\},$$

and for the unit vectors through the corresponding points of attachment we put

$$n_{11} = \{-1, 0\},$$

$$n_{12} = \{0, -1\},$$

$$n_{11} = \{-1, 0\}, \quad n_{21} = (1/5)\{-3, -4\}$$

$$n_{12} = \{0, -1\}, \quad n_{22} = (1/5)\{4, -3\}$$

$$n_{11} = \{-1, 0\}, \quad n_{21} = (1/5)\{-3, -4\}, \quad n_{31} = (1/5)\{-4, 3\},$$

$$n_{12} = \{0, -1\}, \quad n_{22} = (1/5)\{4, -3\}, \quad n_{32} = (1/5)\{-3, -4\},$$

$$n_{11} = \{-1, 0\}, \quad n_{21} = (1/5)\{-3, -4\}, \quad n_{31} = (1/5)\{-4, 3\}, \quad n_{41} = \{0, -1\}$$

$$n_{12} = \{0, -1\}, \quad n_{22} = (1/5)\{4, -3\}, \quad n_{32} = (1/5)\{-3, -4\}, \quad n_{42} = \{1, 0\}.$$

Therewith we have to solve the following four basic equations successively which determine the four position angles:

$$G_1 = (B_{11} - A_1) \cdot n_{12} - L_1 = 0, \quad G_2 = (B_{21} - A_2) \cdot (B_{21} - A_2) - L_2^2 = 0,$$
$$G_3 = (B_{13} - A_3) \cdot (B_{13} - A_3) - L_3^2 = 0, \quad G_4 = (B_{22} - A_4) \cdot (B_{22} - A_4) - L_4^2 = 0.$$

$$G_1 = (B_{11} - A_1) \cdot n_{12} - L_1 = 0, \quad G_2 = (B_{21} - A_2) \cdot n_{22} - L_2 = 0,$$
$$G_3 = (B_{13} - A_3) \cdot (B_{13} - A_3) - L_3^2 = 0, \quad G_4 = (B_{22} - A_4) \cdot (B_{22} - A_4) - L_4^2 = 0.$$

$$G_1 = (B_{11} - A_1) \cdot n_{12} - L_1 = 0, \quad G_2 = (B_{21} - A_2) \cdot n_{22} - L_2 = 0,$$
$$G_3 = (B_{13} - A_3) \cdot n_{32} - L_3 = 0, \quad G_4 = (B_{22} - A_4) \cdot (B_{22} - A_4) - L_4^2 = 0.$$

$$G_1 = (B_{11} - A_1) \cdot n_{12} - L_1 = 0, \quad G_2 = (B_{21} - A_2) \cdot n_{22} - L_2 = 0,$$
$$G_3 = (B_{13} - A_3) \cdot n_{32} - L_3 = 0, \quad G_4 = (B_{22} - A_4) \cdot n_{32} - L_4 = 0.$$

The calculation procedure described in Section 2 yields:

A (3.5)1P: 54 positions, 10 real, A (3.5)2P: 46 positions, 8 real,
A (3.5)3P: 28 positions, 8 real, A (3.5)4P: 16 positions, 8 real.

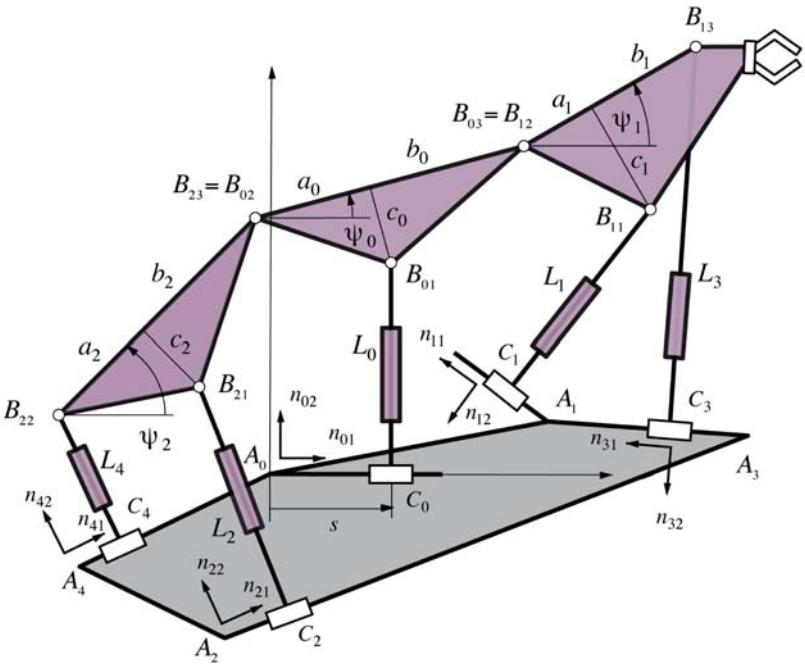


Fig. 4 Robot Assur group A (3.5)5P fixed on a rigid body with five input P-joints.

3.2 Case Study A (3.5)5P

Figure 4 shows an Assur group with five prismatic input joints serving as the basic mechanism for a redundant planar robot. The twofold redundancy (2) can be used to avoid singularity positions in which the robot would lose its maneuverability.

If five input joints are P-joints, the position of the Assur group within its fixations can be given by the distance s (in Figure 4) together with the three angles ψ_0, ψ_1 and ψ_2. With $B_{01} = \{s, L_6\}$ (instead of $B_{01} = L_0\{\cos\psi, \sin\psi\}$ in the set of formulas (1)), and with the parameters p, P as in Section 3.1

$$p = \{a_0 = 3, b_0 = 4, c_0 = 2\}, \{a_1 = 2, b_1 = 3, c_1 = 3\}, \{a_2 = 3, b_2 = 4, c_2 = 2\},$$

$$p = \{A_0 = \{0, 0\}, A_1 = \{5, 11\}, A_2 = \{-5, 10\}, A_3 = \{10, 6\}, A_4 = \{-7, 1\}\},$$

and with the unit vectors n as in Section 3.1 (4)

$$n_{11} = \{-1, 0\}, \quad n_{21} = (1/5)\{-3, -4\}, \quad n_{31} = (1/5)\{-4, 3\}, \quad n_{41} = \{0, -1\},$$

$$n_{12} = \{0, -1\}, \quad n_{22} = (1/5)\{4, -3\}, \quad n_{32} = (1/5)\{-3, -4\}, \quad n_{42} = \{1, 0\},$$

and finally with the lengths

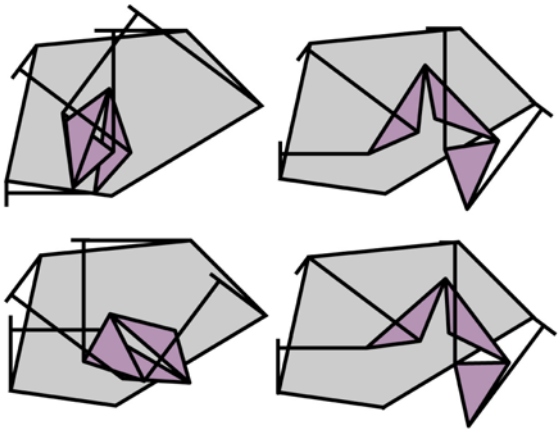


Fig. 5 Four real positions of a normal Assur group A (3.5)5P.

$$L = \{L_0 = 6, L_1 = 7.997, L_2 = 9.198, L_3 = 8.349, L_4 = 5.804\},$$

the four equations to be solved read as in Section 3.1 (4) and take, with $s = X$ and the last three parts of the formula set (1), the form of four algebraic equations as in formula (4). With the matrices $\mathbf{U}$, $\mathbf{V}$, $\mathbf{x}$, $\mathbf{y}$ defined in Section 2 by the formulas (6), (7), we find two equations in X and Y from the vanishing determinants:

$$\begin{aligned}\mathrm{Det}\,\mathbf{U} = 0 = F_1(X, Y) = (1 + Y^2)^2[f_{11}X^2 + f_{12}X + f_{13}] = 0,\\ \mathrm{Det}\,\mathbf{V} = 0 = F_2(X, Y) = (1 + Y^2)^2[f_{21}X^2 + f_{22}X + f_{23}] = 0,\end{aligned}$$

where the coefficients in the series expansions are polynomials in Y of order 4. They can be written in the form of a single matrix equation

$$\mathbf{S}(Y)\cdot\mathbf{X} = 0 \quad \text{with} \quad \mathbf{S} = \begin{bmatrix} 0 & f_{11} & f_{12} & f_{13} \\ f_{11} & f_{12} & f_{13} & 0 \\ 0 & f_{21} & f_{22} & f_{23} \\ f_{21} & f_{22} & f_{23} & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{X} = \{X^3, X^2, X, 1\}^T.$$

The condition $\det\mathbf{S} = 0$ yields a polynomial of order 8 in Y, which can be solved numerically. From $\mathbf{S}(Y_\alpha)\cdot\mathbf{X}_\alpha = 0$ the corresponding values X_α can then be obtained, e.g. as the third component of the null space vector $\mathbf{X}_\alpha$. From $\mathbf{U}(X_\alpha, Y_\alpha)\cdot\mathbf{x}_\alpha = 0$ or $\mathbf{V}(X_\alpha, Y_\alpha)\cdot\mathbf{y}_\alpha = 0$ the corresponding values x_α and y_α can be found in the same way. Finally with the sets $\{X, Y, x, y\}_\alpha$ also the sets $\{s, \psi_0, \psi_1, \psi_2\}$ are known. For the Assur group A (3.5)5P we have found eight positions, the four real of which are shown in Figure 5.

References

1. Leinonen, T. (Ed.) (1991), Terminology for the theory of machines and mechanisms, *Mechanism and Machine Theory* **26**(5), 450.
2. Assur, L.W. (1914, 1915), Isledovanie ploskich sterschnich mechanismow s nischnimi parami, *Iswestia Petrogradskowo Politechn.*, Petrograd, Russia.
3. Artobolevsky, I.I. (1977), *Théorie de Méchanismes et des Machines*, Mir, Moscow, Russia, p. 50.
4. Paisach, E.E. (1985), Opredelenie poloschenie swenev group trechpowodkowoi tschetirechswennich group Assura, *Maschinowedenie* **5**, 56–61.
5. Innocenti, C. (1993), Analytical determination of the intersections of two coupler-point curves generated by two four-bar linkages, in *Computational Kinematics*, J. Angeles et al. (Eds.), Kluwer Academic Publishers, Dordrecht, the Netherlands, pp. 251–262.
6. Lösch, S. (1995), Parallel redundant manipulators based on open and closed normal Assur chains, in *Computational Kinematics '95*, Kluwer Academic Publishers, Dordrecht, the Netherlands, pp. 251–260.
7. Lin, H., Qizheng, L. and Chonggao L. (2000), Closed-form displacement analysis for a nine-link Barranov truss or an eight-link Assur group, *Mechanism and Machine Theory* **35**(3), 379–390.
8. Wunderlich, W. (1963), Höhere Koppelkurven, *Österreichisches Ingenieur Archiv* **XVII**(3), 162–165.
9. van der Waerden, B.L. (1971), *Algebra I*, Springer, Berlin, Germany.

Kinematics of Free-Floating Systems through Optimal Control Theory

G. Le Vey

IRCCyN UMR-CNRS 6597 and Ecole des Mines de Nantes, 4 rue A. Kastler, BP 20722, F-44307 Nantes, France, e-mail: levey@emn.fr

Abstract. This work presents a new method for solving the inverse kinematical problem for free-floating space manipulators. It is based on a novel formulation of the problem within the framework of optimal control theory for discrete linear systems, thanks to a reconsideration of the energy conservation property of this class of systems. This way, inverse kinematics is obtained in a *purely deductive* manner, quite analogous to previous works by the author about dynamical forward and backward models, for either multibody or continuous hyperredundant actuated mechanical systems. One important by-product of the approach is an effective way for detecting *dynamical singularities* for the considered class of systems.

Key words: inverse kinematics, dynamical singularities, optimal control.

1 Introduction

The purpose of this work is to present a new, *recursive*, method that solves the inverse kinematical problem – i.e. computes the joint velocities, given the end-effector velocity – for a free-floating space system, the prototype of which is made of a spacecraft and a serial manipulator fixed on it, with vanishing gravity. For such systems, motion of the manipulator disturbs the attitude and position of the spacecraft, the converse being equally true, thereby making the localization of the manipulator end-effector inaccurate. To obviate these difficulties, one can either try to compensate for disturbed spacecraft motion by using control devices (fuel jets or reaction wheels) [6], which is generally a high energy consuming approach, or consider these interactions while solving the inverse kinematic problem [2, 7, 9–11]: the approach followed here falls into this second category. One interest in this problem comes from the fact that so-called *dynamical singularities* can occur in such systems, due to the fact that the spacecraft is not fixed as for usual terrestrial manipulators [8, 12] and these singularities cannot be predicted from the knowledge of the system geometric parameters solely. The approach followed hereafter, which is new, makes use of a suitable reformulation of the inverse kinematical problem as a multistage optimal control problem, with independent variable the label of the bod-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 177–184.

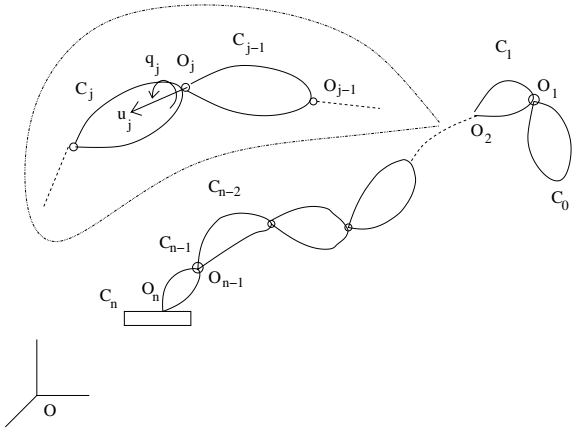


Fig. 1 Spacecraft-manipulator free-floating system.

ies in the chain. To this end, the energy conservation property is considered anew, leading to a *purely deductive* method for computing the searched after joint velocities, in the same spirit as previous work by the author on dynamics of actuated mechanical structures [3–5]. The paper consists of two main parts: in Section 3, formulation of the inverse kinematics as a discrete optimal control problem is given. Then, in Section 4, the solution of the latter is given, using classical results from optimal control theory, leading to a new, recursive algorithm for the inverse kinematics of the system under study. It is followed by a discussion on the by-products of the approach, especially the detection of *dynamical singularities*. Some conclusions are given in Section 5.

2 Notations

The spacecraft+manipulator system is described (see Figure 1), in a galilean reference frame with origin O, by one rigid body (the spacecraft) C_n and an serial chain (the manipulator), which is constituted by n rigid bodies, C_0 to C_{n-1}, labelled from the end-effector back to the body attached to the spacecraft (opposite to the usual numbering). To each body is attached a frame with origin O_j, about one axis of which, u_j, rotation only can be effected for actuation. Relevant geometric and inertial parameters appear in Table 1. In the sequel, dots over some quantity indicate differentiation wrt time. For a vector $y \in R^3$, $\hat{y}$ is the antisymmetric matrix such that $\forall z \in R^3$, $\hat{y}z = y \times z$ and $\times$ is the usual vector product. Whenever both formalisms are used, no confusion should occur.

Table 1 Geometric and inertia parameters.

q_j	Joint j position variable (a scalar)	M_j	Mass of C_j
G_j	mass center of body j	I_j	Inertia of body C_j about G_j
u_j	unit vector along joint j rotation axis	ω_j	Rotation velocity of C_j
L_j	vector $O_{j-1}O_j$	V_j	Galilean velocity of O_j
D_j	vector O_jG_j	V_{g_j}	Galilean velocity of G_j

3 Statement of the Kinematical Problem

The energy conservation property for free-floating systems is first reconsidered. Following this, the kinetic energy is then computed and the design (geometric/kinematical) constraints are derived. Eventually a convenient discrete evolutionary optimization problem, to be solved in Section 4, is posed that leads in a natural way to the new method for the inverse kinematics of the considered class of systems.

3.1 Problem Formulation

When considering a free-floating space system made of a platform (the spacecraft) and a serial manipulator, one can consider that no potential energy exists (the gravity vanishes) thereby reducing the total energy to its kinetic part. In that context, where no external forces are applied, the kinetic energy, T, of the system is a first integral, i.e. it is constant over the motion trajectories (other first integrals of course exist but energy is sufficient here for the derivations in view). But this must be true while considering that the constraints imposed must remain valid over the considered motion. One way for setting the above problem leads to *finding the stationary conditions ($\delta T = 0$) for the system kinetic energy subject to the constraints – in terms of velocities – that link the bodies altogether*. This is the main idea of the method exposed hereafter.

3.2 Kinetic Energy of the System

The kinetic energy of the complete system is $T = \sum_{i=0}^{n} T_j$, where the kinetic energy, T_j, of each body j writes, thanks to Koenigs theorem:

$$T_j = \frac{1}{2}\omega_j^T I_j \omega_j + \frac{1}{2} M_j V_{g_j}^T V_{g_j} \tag{1}$$

As it is usual to consider velocities at the joints, V_j (because actuation applies at the joints), one can use the fact that the field of velocities of a rigid body is a torsor in order to write: $V_{g_j} = V_j + \omega_j \times D_j$, leading to the expression of T, when gathering

terms:

$$T = \frac{1}{2}\sum_{j=0}^{n} \begin{pmatrix} V_j^T & \omega_j{}^T \end{pmatrix} \begin{pmatrix} M_j I & M_j \hat{D}_j^T \\ M_j \hat{D}_j & M_j \hat{D}_j \hat{D}_j^T + I_j \end{pmatrix} \begin{pmatrix} V_j \\ \omega_j \end{pmatrix} \tag{2}$$

3.3 Kinematical Constraints

For simplicity, the joints are supposed to be rotational only, but prismatic joints could be treated equally at the cost of greater computational complexity. The only modelling constraint on the multibody system is then that each body is rigid and is attached to its neighbours at the rotatory joints, so that, in a fixed (Galilean) reference frame, with origin O, one has again to write that the field of velocities for solid bodies is a torsor:

$$V_j = V_{j-1} + \omega_{j-1} \times L_j, \quad j = 1, \ldots, n \tag{3}$$

On the side of rotations, at each joint one has:

$$\omega_j = \omega_{j-1} + \dot{q}_j u_j \tag{4}$$

where $\dot{q}_j$ are the joint velocities. Gathering the results while using matrix notation for the ease of analogy with optimal control in standard Kalman form, one gets:

$$\begin{pmatrix} V_j \\ \omega_j \end{pmatrix} = \begin{pmatrix} I & \hat{L}_j^T \\ 0 & I \end{pmatrix} \begin{pmatrix} V_{j-1} \\ \omega_{j-1} \end{pmatrix} + \begin{pmatrix} 0 \\ u_j \end{pmatrix} \dot{q}_j \tag{5}$$

Equation (5) is the simple consequence, in terms of velocities, of the geometric constraints due to the design.

3.4 Kinematics as an Optimization Problem

As suggested by the above analysis and observing Eqs. (2) and (5), it is worth considering the following optimal control problem. Firstly, for simplifying the presentation and in accordance with usual notation of control theory, define the quantities (I is the 3×3 identity matrix):

$$X_j = \begin{pmatrix} V_j \\ \omega_j \end{pmatrix}; \quad U_j = \dot{q}_{j+1}; \quad F_j = \begin{pmatrix} I & \hat{L}_{j+1}^T \\ 0 & I \end{pmatrix}; \quad G_j = \begin{pmatrix} 0 \\ u_{j+1} \end{pmatrix}$$

$$A_j = \begin{pmatrix} M_j I & M_j \hat{D}_j^T \\ M_j \hat{D}_j & M_j \hat{D}_j \hat{D}_j^T + I_j \end{pmatrix}; \quad J(X_j) = \sum_{j=0}^{n} \frac{1}{2} X_j^T A_j X_j \tag{6}$$

where A_j is obviously symmetric positive and, as a rule, definite. The inverse kinematical problem posed in Section 3.1 can then be formulated as follows:

Find the critical points of J ($\delta J = 0$), under the set of constraints:

$$X_{j+1} = F_j X_j + G_j U_j, \quad j = 0, \ldots, n-1 \tag{7}$$

where the "initial" condition X_0 is known, as being the velocity of the end-effector. Obviously, solving this problem will lead to the inverse kinematical problem solution, as by definition, U_j is $\dot{q}_j$. For that purpose, classical results from LQ multistage optimal control theory and the weak Pontriaguine maximum principle are briefly recalled in the following section. Detailed computations can be found in, e.g., [1].

4 Joint Velocities as the Optimal Control

Introducing Lagrange multipliers (*costates*), the constraint equations are adjoined to the criterion J leading to an equivalent unconstrained optimization problem. The solution of the latter is obtained in summary as follows. An augmented criterion $\bar{J}$ is built:

$$\bar{J} = \frac{1}{2} X_n^T A_n X_n + \sum_{j=0}^{n-1} \left[\frac{1}{2} X_j^T A_j X_j - \lambda_{j+1}^T (X_{j+1} - F_j X_j - G_j U_j) \right] \tag{8}$$

and a critical point for $\bar{J}$ is obtained as a solution of $\delta \bar{J} = 0$, leading to the necessary (and sufficient as the criterion is quadratic positive definite) first order conditions [1]:

$$\frac{\partial \bar{J}}{\partial \lambda_j} = \frac{\partial \bar{J}}{\partial X_j} = \frac{\partial \bar{J}}{\partial U_j} = 0, \quad j = 0, \ldots, n-1 \tag{9}$$

Explicit computation of these conditions leads to ($j = 0, \ldots, n-1$):

$$\begin{cases} X_{j+1} = F_j X_j + G_j U_j \\ \lambda_j = F_j^T \lambda_{j+1} + A_j X_j \\ 0 = G_j^T \lambda_{j+1} \end{cases} \tag{10}$$

together with boundary conditions: X_0 (given velocity of the end-effector) and $\lambda_n = A_n X_n$, hence making the above a *two-point boundary-value problem*. The last of Eq. (10) does not give an expression for the optimal control, making it a *singular problem*. Instead, the second derivative $\partial^2 \bar{J} / \partial U^2$ vanishes and some extra work is in order to make U_j appear, that satisfies the optimality system. To this end, an efficient method for solving the above boundary value problem is the *sweep method*. It is based on the search for an expression of the costate λ_j that is linear in the state X_j, which writes that $\lambda_j = S_j X_j$, S_j a square matrix. Omitting the details

that are routine (see [1]), the optimal control sequence U_j is computed as follows ($j = 0, \ldots, n-1$):

1. Solve backwards the following equation for S_j, with terminal condition $S_n = A_n$ (as $\lambda_n = A_n X_n$):

$$S_j = F_j^T S_{j+1} F_j + (F_j^T S_{j+1} G_j)(G_j^T S_{j+1} G_j)^{-1}(G_j^T S_{j+1} F_j) + A_j \quad (11)$$

2. Compute the optimal control (actually a *feedback* control) as:

$$U_j = -(G_j^T S_{j+1} G_j)^{-1}(G_j^T S_{j+1} F_j) X_j \quad (12)$$

3. While solving the state equation (first of Eq. (10), which writes, after substitution of U_j:

$$X_{j+1} = \left(F_j - G_j (G_j^T S_{j+1} G_j)^{-1}(G_j^T S_{j+1} F_j)\right) X_j \quad (13)$$

4.1 Inverse Kinematics Algorithm

The above computations lead to the inverse kinematics algorithm:

– **Data**: L_j, D_j, M_j, I_j, V_0, ω_0, $S_n = A_n$
– **Begin**

1. **Compute** S_j, $j = n-1, \ldots, 0$ through Eq. (11)
2. **Compute** V_j, ω_j, $j = 1, \ldots, n$ through Eq. (13)
3. **Compute** $\dot{q}_j$, $j = 1, \ldots, n$ through Eq. (12)

– **End**

Notice that the complexity of the algorithm is low as there are only basic operations. In particular, due to the definition of G_j as a column-vector, there is no matrix inversion, as $G_j^T S_{j+1} G_j$ in Eq. (13) is a scalar.

4.2 Discussion-Dynamic Singularities Analysis

The above developments have exhibited a new recursive inverse kinematics algorithm for free-floating systems. But the novelty of the approach is not limited to it as it is shown below, in particular as far as *dynamical singularity analysis* is concerned.

- The second of Eq. (10), the adjoint equation, receives an obvious interpretation. Substituting for F_j, A_j, X_j, we can write:

$$\lambda_j = \begin{pmatrix} I & 0 \\ \hat{L}_{j+1} & I \end{pmatrix} \lambda_{j+1} + \begin{pmatrix} M_j I & M_j \hat{D}_j^T \\ M_j \hat{D}_j & M_j \hat{D}_j \hat{D}_j^T + I_j \end{pmatrix} \begin{pmatrix} V_j \\ \omega_j \end{pmatrix} \tag{14}$$

the second term in the second member of this recursion is the kinetic torsor of the j-th body. Thus for the recursion to be dimensionally homogeneous, λ_j has to be homogeneous to a similar kinetic quantity. Actually, as it is the dual quantity of the velocity for the kinetic energy metric, it is simply the kinetic torsor at joint j. Thus Eq. (14) gives a recursion for its computation, starting from $\lambda_n = A_n X_n$, X_n given as an output of the above algorithm.

- Relying upon the previous interpretation, reconsider now the definition of λ_j: $\lambda_j = S_j X_j$. λ_j being a kinetic torsor and X_j a velocity, S_j must be homogeneous to an *inertia* (which is consistent with the initialization $S_n = A_n$). Thus Eq. (11) gives a recursive way of computing the resulting inertias at each joint in the overall system. As S_j is computed *backwards* from n to 0, one sees that velocities at each joint depend on all inertias of the overall system, which was asserted in [8], thanks to a clever choice of configuration parameters.
- Inspection of Eq. (13) shows at once that the recursive computation of the X_j's, the velocities, can be done whenever the parenthesized expression, which obviously depends on inertias of the system through S_j, is not singular. This corroborates the results of [8]. Thus *singularities* (*kinematic* as well as *dynamic*, but for these last, *kinetic* would be a better descriptipon as only kinetic quantities intervene) may appear whenever the following 6×6 matrix is singular:

$$F_j - G_j (G_j^T S_{j+1} G_j)^{-1} (G_j^T S_{j+1} F_j) \tag{15}$$

Its possible rank deficiency can be easily checked at each step j. It can even be reduced to a 3×3 matrix rank check. Hence, one has an efficient $O(n)$ algorithm to check the presence of these dynamical singularities. This has to be compared to the usual $O(n^3)$ *Jacobian* determinant computation, a delicate computation task.

- The formulation of the inverse kinematical problem as a LQ-optimal control problem opens possibilities to address other questions related to the kinematics of the considered systems, but not touched upon in the present work, such as imposing bounds on joint velocities, estimation in case of uncertainty. This will be addressed in forthcoming publications.

5 Conclusions

A new recursive, efficient method has been presented for solving the inverse kinematics of free-floating space systems. It leads to an efficient, computationally attractive mean for dynamical singularities analysis. One important feature of the approach is that it is *purely deductive*, thanks to a suitable formulation as a constrained, discrete evolution, optimal control problem, making useless to enter a

priori into mechanical definitions, but letting interpretations come afterwards, once logical formal computations have been done. Extensions are being studied.

References

1. Bryson, A.E. and Ho, Y.C., *Applied Optimal Control*. Hemisphere Publishing, 1975 (Revised printing).
2. Dubowsky, S. and Torres, M.A., Path planning for space manipulators to minimize spacecraft attitude disturbances. In *Proceedings 1991 IEEE International Conference Robotics and Automation*, pp. 2522–2528, 1991.
3. Le Vey, G., Continuous Newton–Euler algorithms for geometrically exact flexible beams. In *Proceedings 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4163–4168, October 2006.
4. 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.
5. Le Vey, G., Dynamics and control of actuated parallel structures as a constrained optimization problem through Gauss' principle and Appell's equations. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 1480–1485, 2007.
6. Longman, R.W., Lindberg, R.E. and Zedd, M.F., Satellite-mounted robot manipulators – New kinematics and reaction moment compensation. *International Journal of Robotics Research* **6**(3), 884–889, 1987.
7. Papadopoulos, E. and Dubowsky, S., On the nature of control algorithms for free-floating space manipulators. *IEEE Transactions on Robotics and Automation* **7**(6), 750–758, 1991.
8. Papadopoulos, E. and Dubowsky, S., Dynamic singularities in free-floating space manipulators. *ASME Journal of Dynamic Systems, Measurement and Control* **115**, 44–52, 1993.
9. Reyhanoglu, M. and McClamroch, N.H., Planar reorientation maneuvers of space multibody systems using internal controls. *AIAA Journal on Guidance, Control and Dynamics* **15**(6), 1475–1480, 1992.
10. Umetani, Y. and Yoshida, K.. Resolved motion rate control of space manipulators with generalized Jacobian matrix. *IEEE Transactions on Robotics and Automation* **5**(3), 303–314, 1989.
11. Walsh, G.C. and Satry, S., On reorienting linked rigid bodies using internal motions. In *Proceedings 1991 IEEE Conference on Decision and Control*, pp. 1190–1195, 1991.
12. Xi, F. and Fenton, R.G., On the inverse kinematics of space manipulators for avoiding dynamic singularities. *ASME Journal of Dynamic Systems, Measurement and Control*, **119**(2), 340–346, 1997.

Genericity Conditions for Serial Manipulators

Peter Donelan

School of Mathematics, Statistics and Computer Science,
Victoria University of Wellington, New Zealand, e-mail: peter.donelan@vuw.ac.nz

Abstract. A generic, or more properly 1-generic, serial manipulator is one whose forward kinematic mapping exhibits singularities of given rank in a regular way. In this paper, the product-of-exponentials formulation of a kinematic mapping together with the Baker–Campbell–Hausdorff formula for Lie groups is used to derive an algebraic condition for the regularity.

Key words: generic manipulator, transversality, singularity, Lie group, product of exponentials.

1 Introduction

The idea of a generic manipulator was introduced by Pai and Leu [11], who observed that mathematical singularity theory (e.g. [4]) may provide powerful tools for exploring robot singularities. For a given manipulator one wants to understand the singularity locus, how it partitions the joint space and how its image appears in the workspace. A simple invariant of a singularity is its rank – the property of *1-genericity* requires certain regularity or, more precisely, transversality conditions for the forward kinematic mapping of a serial manipulator which ensure that the locus of singularities of fixed rank must be a submanifold (smooth subset of the joint space) whose dimension is determined by the number k of joints and the rank. In particular, there will be no singularities if the rank is greater than a certain number depending on k (either 1 or 2). In particular, only for $k = 6$ did Pai and Leu explicitly determine a general condition for 1-genericity of a kinematic mapping and went on to show that the property held for certain partitioned manipulators where one could consider separately rotational and translational kinematics. Subsequent work on generic manipulators has been undertaken by Tsai et al. [13] who extended these results to full 6-dof serial manipulators, Burdick [1] who obtained specific equations for the bifurcation sets where the regularity fails in the case of regional manipulators, and Lerbet and Hao [6] who, in studying over-constrained closed loop mechanisms, showed how the transversality condition relates explicitly to the algebra of the joint screws. The practical relevance of genericity was realised

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 185–192.

by Narasimhan and Kumar [10] who introduced a manipulator control algorithm adapted to handle generic singularities.

In this paper, a different approach is presented to obtain results like those of Hao and Lerbet, putting them in an algebraic context which shows how they can be generalised and potentially used to tackle substantial questions about the singularities of classes of manipulator. For example, in singularity theory, the term *generic* is used to describe any property of a family of mappings that is common to almost all members of the family. 'Almost all' has a topological meaning – the set should be open and dense. Openness ensures that any mapping with the property is surrounded by a neighbourhood of mappings that share the property, so that the property is stable. Density means that if a mapping fails to have the property, then arbitrarily close to it is one that does – there are no open regions where every mapping fails to have the property.

There is a specific set of tools that are used to guarantee genericity: transversality theorems. A survey of the technical details can be found in [2], where some of these results first appeared. There is one set of transversality theorems that guarantee genericity when one is interested in the infinite-dimensional space of *all* smooth mappings between given manifolds. In this setting, 1-genericity is automatically generic. However, families of serial manipulators are only ever finite-dimensional and there is no certainty that the special associated families of kinematic mappings sit nicely within the very large space of all mappings. In this setting, an Elementary Transversality Theorem is required and a transversality hypothesis must be satisfied before a conclusion of genericity can be drawn. The transversality condition established here would seem to be quite applicable in establishing theorems of this kind for 1-generic manipulators.

2 Jet Extensions of Kinematic Mappings

Suppose the forward kinematics of a serial manipulator with m joints is given by the smooth function $f : M^{(m)} \to SE(3)$, where M is the m-dimensional joint manifold (the superscript denoting the dimension) and $SE(3)$ is the 6-dimensional Lie group of Euclidean isometries describing the motion of the end-effector. A given set of joint variables $x \in M$ is a singularity of the robot kinematics if the rank of the derivative $Df(x)$ falls below the maximum possible value, i.e. the minimum of m and 6. The number $\min\{m, 6\} - \text{rank } Df(x)$ is called the *corank* of the singularity. We briefly present the key definitions; more details can be found in [2, 4, 7, 11].

One way to recognise a singularity is to consider the first-order Taylor expansion, with respect to some choice of coordinates for M and $SE(3)$, at points $x \in M$. This includes information on the derivative $Df(x)$ so it tells us whether there is a singularity. Regarding x as a variable, there is a smooth map $j^1 f$, called the *1-jet extension* of f, which assigns to each $x \in M$ the first-order Taylor expansion of f at x. The values of this map lie in a space called the *1-jet bundle*, which is itself a smooth manifold. Locally this looks like a product of an open piece of M locally

coordinatised by (part of) $\mathbb{R}^m$, an open piece of $SE(3)$ with coordinates in $\mathbb{R}^6$, and the vector space of matrices $M(\mathbb{R}^m, \mathbb{R}^6)$ – the *fibre* of the bundle – representing derivatives of a mapping $\mathbb{R}^m$ to $\mathbb{R}^6$ representing f in local coordinates. The matrices of fixed corank s form a submanifold of $M(\mathbb{R}^m, \mathbb{R}^6)$ of codimension $s(|6-m|+s)$, so there is a submanifold Σ^s of the jet bundle consisting of 1-jets of corank s having the same codimension.

The 1-jet extension $j^1 f$ is *transverse* to one of the submanifolds Σ^s, denoted $j^1 f \pitchfork \Sigma^s$, provided that whenever $x \in M$ and $j^1 f(x) \in \Sigma^s$, then the image of the derivative of $j^1 f$ at x spans a complement to the tangent space to Σ^s in the tangent space to the 1-jet bundle at $j^1 f(x)$. If $j^1 f$ is transverse to all the submanifolds Σ^s, $s = 0, 1, \ldots, \min\{m, 6\}$ then the manipulator is called *one-generic*, sometimes abbreviated to simply *generic*. This is sufficient to ensure that the inverse image $\Sigma^s f = (j^1 f)^{-1}(\Sigma^s)$ is either empty or a submanifold of M with the same codimension as Σ^s.

Transversality here depends on the second-order partial derivatives of f at x. There is an invariant part of this family of second-order derivatives called the *intrinsic second derivative*. Consider for each x_l, $l = 1, \ldots, m$, that part of the matrix $\left(\frac{\partial^2 f_j}{\partial x_i \partial x_l}\right)_{ij}$ given by restricting it to the kernel of $Df(x)$ and then projecting the result onto the cokernel, i.e. the quotient of $T_{f(x)}SE(3)$ by the image of $Df(x)$ (the column space of the Jacobian matrix). Each of these could be written as a $s \times (|6-m|+s)$ matrix with respect to some bases for the kernel and cokernel. A necessary and sufficient condition for transversality to Σ^s is that these m matrices span the vector space of all such matrices. Equivalently, the matrices obtained by restriction but not projection, together with the submatrices formed by taking s columns of the Jacobian should span the space of $6 \times s$ matrices.

A necessary condition for transversality to Σ^s is that $m \geq s(|6-m|+s)$. As a consequence, for 1-generic manipulators $\Sigma^s f$ is empty for $s \geq 2$ unless $m = 6, 7, 8$ in which case $\Sigma^2 f$ may be a a submanifold of codimension 4, 6 or 8 respectively. In [13], it was also noted that for reasons of symmetry the resulting dimensions in the cases $m = 7, 8$ are too small to occur transversely. However, one is often interested not simply in a given manipulator but a family of manipulators in which some design parameters may be altered. Suppose the design parameters lie in a manifold B of dimension k. Then there is a family of kinematic mappings which can be thought of as a function of the joint variables in M and parameters in B, $F : M \times B \to SE(3)$. For each $b \in B$, there is an ordinary kinematic mapping $F_b : M \to SE(3)$, given by $F_b(x) = F(x, b)$. So now we have a map

$$\Phi : M \times B \to J^1(M, SE(3)), \quad \Phi(x, b) = j^1 F_b(x). \tag{1}$$

In this situation one could encounter transversely singularity strata, Σ^s, in the jet bundle up to codimension $m + k$. The Elementary Transversality Theorem (see for example [4]) asserts that if If $\Phi \pitchfork \Sigma^s$ for all $0 \leq s \leq \min\{m, 6\}$ then the set of parameter values for which the individual kinematic mappings are 1-generic

$$\{b \in B : j^1 F_b \pitchfork \Sigma^s, 0 \leq s \leq \min\{m, 6\}\}$$

is open and dense in the parameter space B.

3 One-Genericity for Serial Manipulators

For a serial manipulator, the kinematic mapping as a function of the joint variables $\theta_1, \ldots, \theta_m$ can be written as a product of exponentials of the screws $X_1, \ldots, X_m$ representing the sequence of joints:

$$f(\theta_1, \theta_2, \ldots, \theta_k) = H \cdot e^{\theta_1 X_1}.e^{\theta_2 X_2} \cdots e^{\theta_m X_m}. \tag{2}$$

$H \in SE(3)$ determines the home configuration, $\theta_1 = \cdots = \theta_m = 0$, of the end-effector and, when considering the mapping in the neighbourhood of a given point, we may choose end-effector and base coordinates so that H is the identity. In this representation, the design parameters of the manipulator are only implicitly available via the sequence of screws $X_1, \ldots, X_m$. The exponential function, $\exp(\theta X)$ or $e^{\theta X}$, represents the position of a link attached by the joint represented by screw X to the previous link, after the joint has moved through joint variable θ (either angle of rotation for an R- or H-joint, or translation for a P-joint) as an element of $SE(3)$. It can be explicitly calculated in matrix form if screws and elements of $SE(3)$ are given matrix representations by means of the usual series expansion for the exponential. In order to simplify exposition and concentrate on the most important case, from here on we set $m = 6$.

Non-commutativity of Euclidean isometries means that the exponentials in (2) cannot be reordered. The Lie bracket of a pair of screws is another screw defined as follows: if X_1, X_2 are represented in Plücker coordinates by $(\boldsymbol{\omega}_i, \mathbf{v}_i)$, $i = 1, 2$ then

$$[X_1, X_2] = (\boldsymbol{\omega}_1 \wedge \boldsymbol{\omega}_2, \boldsymbol{\omega}_1 \wedge \mathbf{v}_2 - \boldsymbol{\omega}_2 \wedge \mathbf{v}_1)$$

where $\wedge$ is the standard vector product in $\mathbb{R}^3$. One way to think of the Lie bracket is that it measures the infinitesimal difference between the effect of a pair of screws or joints if one reverses the order in which they occur. The finite, rather than infinitesimal,consequence for exponentials is expressed in the formula of Baker–Campbell–Hausdorff; see, for example, [12]. For sufficiently small values of the joint variables, multiple applications of the Baker–Campbell–Hausdorff formula convert (2) to a single exponential in terms of the Lie brackets of pairs of joint screws:

$$f(\theta_1, \ldots, \theta_6) = \exp\left(\sum_{i=1}^{6} \theta_i X_i + \tfrac{1}{2} \sum_{1 \le i < j \le 6} \theta_i \theta_j [X_i, X_j] + O(3)\right) \tag{3}$$

where the coefficients of the order 3 and higher terms in the θ_is involve nested brackets of the X_is. The kinematic mapping encompasses the full capability of the end-effector motion; to obtain elementary kinematic notions such as velocity and acceleration one considers a time-dependent path in the jointspace so that each joint

variable is dependent on time t. Formulae for the velocity and acceleration that can be found in, for example, [8, 12] can be obtained by differentiating with respect to t, using the Chain Rule and the fact that $(d/dt)\exp(\theta(t)X)|_{t=0} = \dot{\theta}(0)X$. Higher-order derivatives are also considered in [9].

Since the exponential map on a Lie group provides a coordinate system on a neighbourhood of the identity, the kinematic mapping with respect to these exponential coordinates can be represented by the expression within the exponential in (3). From this we are able to derive the following expression for the 1-jet extension, see [2]:

$$\begin{aligned} j^1 f(\boldsymbol{\theta}) = \Bigg(\boldsymbol{\theta},\ & \sum_{i=1}^{6} \theta_i X_i,\ X_1 + \tfrac{1}{2}\sum_{j=2}^{6} \theta_j [X_1, X_j], \ldots, \\ & X_l - \tfrac{1}{2}\sum_{j=1}^{l-1} \theta_j [X_l, X_j] + \tfrac{1}{2}\sum_{j=l+1}^{6} \theta_j [X_l, X_j], \ldots, \\ & X_6 - \tfrac{1}{2}\sum_{j=1}^{5} \theta_j [X_6, X_j] \Bigg) + O(2) \end{aligned} \tag{4}$$

where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_6)$.

From this we can deduce the following conditions for transversality to the singularity strata Σ^s. This represents a different derivation and clarification of the result of Lerbet and Hao [6, appendix A].

Theorem 3.1 *Suppose f is a serial manipulator kinematic mapping given by (2). Necessary and sufficient conditions for $j^1 f$ to be transverse to Σ^s at $\theta_i = 0$, $i = 1, \ldots, 6$ are as follows:*

(a) *If $0 \in \Sigma^1 f$, let $\mathbf{c} \in \mathbb{R}^6$ span the kernel of $Df(0)$. Then the vectors*

$$X_1, \ldots, X_6, [c_1 X_1, X_2],\ [c_1 X_1 + c_2 X_2, X_3], \ldots, \left[\textstyle\sum_{i=1}^{4} c_i X_i, X_5\right]$$

span $\mathbb{R}^6$.

(b) *If $0 \in \Sigma^2 f$, let $\mathbf{c}, \mathbf{d}$ span the kernel of $DF(0)$. Then the 6×2 matrices (where $a \mid$ separates columns)*

$$\begin{aligned} & \left(X_i \mid X_j\right), \quad (1 \le i < \cdots < j \le 6), \\ & \left([c_1 X_1, X_2] \mid [d_1 X_1, X_2]\right),\ \left([c_1 X_1 + c_2 X_2, X_3] \mid [d_1 X_1 + d_2 X_2, X_3]\right), \ldots \\ & \qquad \ldots, \left(\left[\textstyle\sum_{i=1}^{4} c_i X_i, X_5\right] \,\middle|\, \left[\textstyle\sum_{i=1}^{4} d_i X_i, X_5\right]\right) \end{aligned}$$

span $M(2, 6)$.

(c) *$\Sigma^3 f$ is empty for $s \ge 3$.*

Proof. The matrices of second-order partial derivatives required for determining transversality, described in Section 2, are found by differentiating with respect to

each θ_i, $i = 1, \ldots, k$ the components in (4) that are in the fibre of the jet bundle. They are

$$\Upsilon_l = \tfrac{1}{2}\left([X_1, X_l] \cdots [X_{l-1}, X_l]\ \mathbf{0} - [X_{l+1}, X_l] \cdots - [X_6, X_l]\right) \tag{5}$$

for $l = 1, \ldots, 6$. These must be restricted to the kernel of the Jacobian matrix. Suppose $\mathbf{c} \in \ker Df(0)$ then

$$Df(0)\mathbf{c} = \sum_{i=1}^{6} c_i X_i = \mathbf{0} \tag{6}$$

while

$$\Upsilon_l \mathbf{c} = \frac{1}{2}\left[c_1 X_1 + \cdots + c_{l-1} X_{l-1} - c_{l+1} X_{l+1} - \cdots - c_6 X_6, X_l\right]. \tag{7}$$

Adding $\frac{1}{2}\times$(6) to (7) and using the fact that $[X_l, X_l] = 0$ gives

$$\Upsilon_l \mathbf{c} = \left[c_1 X_1 + \cdots + c_{l-1} X_{l-1}, X_l\right] \tag{8}$$

and the theorem follows from the general condition for transversality to the singularity strata in Section 2.

Notes

1. It is important to observe that this is only a local condition for transversality – it does not guarantee the global property of one-genericity unless it applies at every singular configuration of a serial manipulator. To apply the result in that way requires the presentation of the kinematic mapping to be re-calibrated so that the joint screws $X_1, \ldots, X_6$ are those in the current singular configuration. Although they are not the same as those at another configuration, they are related.
2. A different formulation of the theorem could be given by subtracting $\frac{1}{2}\times$(6) from (7) to obtain
$$\Upsilon_l \mathbf{c} = \left[-c_{l+1} X_{l+1} - \cdots - c_6 X_6, X_l\right]. \tag{9}$$
3. The theorem can be generalised readily to cases where $m \neq 6$.
4. The theorem can be generalised to the case of parametrised families of kinematic mappings $F : M \times B \to SE(3)$. The function Φ as in (1) can be written in the form (4), with the parameters $u_1, \ldots, u_k$ in B appearing explicitly in the formula. Then for transversality at $(0, 0) \in M \times B$, in addition to the matrices in Theorem 3.1, we can also use the k matrices arising from the fibre components of $\partial\Phi/\partial u_i$, $i = 1, \ldots, k$ restricted to the kernel of $DF_0(0)$.

In [6, proposition 4.4], the authors observed that a necessary condition for transversality to Σ^1 follows from Theorem 3.1(a): the vector subspace Δ_1 spanned by the screws $X_1, \ldots, X_6$ together with their pairwise brackets $[X_i, X_j]$, $1 \leq i < j \leq 6$

must be the whole 6-dimensional screw space, usually denoted $\mathfrak{se}(3)$. In particular, the subalgebra Δ generated by $X_1, \ldots, X_6$ must be the whole of $\mathfrak{se}(3)$. In the case $m = 6$ these conditions are trivial since then $X_1, \ldots, X_6$ span a 5-space, but there are no subalgebras of dimension 5 so the conditions are automatically satisfied. However the conditions are not sufficient, as can be seen by considering the case where the screw dependency is defined by $c_1 X_1 + c_3 X_3 = 0$ and the bracket $[X_1, X_2] = 0$. In this case all the brackets in Theorem 3.1(a) vanish so the screws cannot span $\mathfrak{se}(3)$.

Note that Δ is an invariant of the manipulator [6] and in the case that it is a proper subalgebra of $\mathfrak{se}(3)$, it is sensible to regard its corresponding subgroup as the range of the kinematic mapping so that one considers, for example, a translational or a rotational manipulator (see for example [6, 11]). In that case of course the dimensions of the singularity strata change.

In general, there is a terminating nested sequence of vector subspaces

$$\Delta_0 \subseteq \Delta_1 \subseteq \cdots \subseteq \Delta_r = \Delta$$

where Δ_0 is the subspace spanned by $X_1, \ldots, X_m$ and $\Delta_{i+1} = \Delta_i + [\Delta_0, \Delta_i]$ for $i \geq 0$ (see [5, 12]). It is fairly easy to show that not only is the sequence independent of the spanning set of screws for Δ_0, so it depends only on the screw system spanned by them, moreover the sequence is invariant up to conjugation (i.e. under the adjoint action of the Euclidean group) and hence the necessary condition $r = 1$ for transversality to Σ^1 can be determined from the Gibson–Hunt class of the screw system [3]. For example, if $m = 4$ and there is a singular configuration at which the 4 joint screws $X_1, \ldots, X_4$ span a 3-system of type IB_2 with pitch modulus $h_\beta = 0$, so that a spanning set of screws, in Plücker coordinates, is

$$(1, 0, 0, 0, 0, 0), \quad (0, 1, 0, 0, 0, 0), \quad (0, 0, 0, 1, 0, 0)$$

then $\Delta_r = \mathfrak{se}(3)$ for $r = 2$ but not $r = 1$. It follows that in such a configuration the serial manipulator kinematic mapping is *not* transverse to Σ^1.

4 Conclusions

A straightforward condition for testing for manipulator genericity has been presented. The conditions in Theorem 3.1 would seem to lend themselves to further analysis of genericity for families of serial manipulators and the formula for the jet extension of a kinematic mapping can be extended to consider higher-order singularities and hence to the study of bifurcation sets such as those separating families of cuspidal and non-cuspidal regional manipulators.

References

1. Burdick, J.W. (1995) A classification of 3R regional manipulator singularities and geometries, *Mechanism and Machine Theory* **30**(1), 71–89.
2. Donelan, P.S. (2007) Singularity-theoretic methods in robot kinematics, *Robotica* **25**(6), 641–659.
3. Gibson, C.G. and Hunt, K.H. (1990) Geometry of screw systems I & II, *Mechanism and Machine Theory* **25**(1), 1–27.
4. Golubitsky, M. and Guillemin, V. (1973) *Stable Mappings and Their Singularities*, Springer Verlag, New York.
5. Hao, K. (1998) Dual number method, rank of a screw system and generation of Lie subalgebras, *Mechanism and Machine Theory* **33**(7), 1063–1084.
6. Lerbet, J. and Hao, K. (1999) Kinematics of mechanisms to the second order – Application to the closed mechanisms, *Acta Applicandae Mathematicae* **59**(1), 1–19.
7. Levine, H.I. (1971) Singularities of differentiable manifolds, in *Proceedings of Liverpool Singularities Symposium 1*, Lecture Notes in Mathematics, Springer Verlag, Berlin, pp. 1–89.
8. Rico Martínez, J.M. and Duffy, J. (1996) An application of screw algebra to the acceleration analysis of serial chains, *Mechanism and Machine Theory* **31**(4), 445–457.
9. Müller, A. and Maißer, P. (2003) A Lie-group formulation of kinematics and Dynamics of constrained MBS and its application to analytical mechanics, *Multibody System Dynamics* **9**, 311–352.
10. Narasimhan, S. and Kumar, V. (1994) A second order analysis of manipulator kinematics in singular configurations, in *Proceedings 23rd Biennial Mechanisms Conference*, Minneapolis, ASME, New York, pp. 477–484.
11. Pai, D.K. and Leu, M.C. (1992) Genericity and singularities of robot manipulators, *IEEE Trans. Robotics and Automation* **8**(5), 545–559.
12. Selig, J.M. (2005), *Geometric Fundamentals of Robotics*, New York, Springer.
13. Tsai, K.Y., Arnold, J. and Kohli, D. (1993) Generic maps of mechanical manipulators, *Mechanism and Machine Theory* **28**(1), 53–64.

Alternative Forms for Displacement Screws and Their Pitches

I.A. Parkin

School of Information Technologies, The University of Sydney, N.S.W. 2006, Australia, e-mail: ian@it.usyd.edu.au

Abstract. It is well established that finite displacement screws effective for the (incompletely specified) relocation of a *body with symmetries* form linearly combined sets if they are of a *sin*-screw form $\hat{\mathbf{S}} = \sin\frac{1}{2}\hat{\theta}\,\hat{\mathbf{s}}$, characterised by pitch $P_S = \frac{1}{2}\sigma/\tan\frac{1}{2}\theta$. This paper shows that screws of indefinitely many other functional forms may be derived, each with a correspondingly distinct definition of pitch, which in the same kinematical situations will also form sets of screws that are linearly combined with dual coefficients. As example, screws of form $\hat{\mathcal{S}} = \sin\hat{\theta}\,\hat{\mathbf{s}}$, of pitch $P_{\hat{\mathcal{S}}} = d/\tan\theta$, are evaluated that describe displacement of a *point-line*.

Key words: kinematics, screw theory, finite displacement screw, pitch.

1 Introduction

Screws of a particular *sin*-screw form, $\hat{\mathbf{S}} = \sin\frac{1}{2}\hat{\theta}\,\hat{\mathbf{s}}$, characterised by pitch $P_S = \frac{1}{2}d/\tan\frac{1}{2}\theta$, have recently found use in representing the finite displacement of a rigid body through a dual angle $\hat{\theta} = \theta + \varepsilon\, d$, $-\pi < \theta \leq \pi$, about a *screw-axis* sited in the unit line $\hat{\mathbf{s}}$, with $|\hat{\mathbf{s}}| = 1$. Using that sin-screw form it is found, when a body *with spatial symmetries of figure* is relocated – or, equivalently, when a displacement is *incompletely specified* – that the (possibly infinite) set of screws available to the body in achieving the relocation is described by linear combination of a small basis of screws [2, 6, 7].

In this paper we show that these properties are by no means unique to the 'sin half-angle' screw form and that, for any such kinematic context, we may derive an indefinitely large number of screw forms of quite different definition (and pitch), each of which occur in similarly constituted linear combinations, formed with dual coefficients in general.

It is not our purpose to advocate use of screw forms other than the sin-screw form: that form, which – as the vector sub-component – has intimate connection with the *unit biquaternion* for the displacement (see Eq. (4)), appears to represent the displacement with least sign-ambiguity and to be the simplest to manipulate in formal analysis. Rather, the purpose is to point out that when – in the course of

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 193–202.

exploratory geometric analysis – a screw of different definition or a different pitch-form appears, it does not necessarily betoken a distinct physical phenomenon but may simply be, in effect, an *alias* for the sin-screw under the kind of derivation just mentioned.

A number of investigations have recently been made into the formal underpinnings of finite displacements which have turned up suggestive measures of pitch for the helicoidal vector fields under study. In one of these [4], for example, the pitch measure $P = d/\sin\theta$ has emerged which, though not that of the sin-screw, is recognisable as the pitch-form associated with the screw $\hat{\mathbf{T}} = \tan\frac{1}{2}\hat{\theta}\,\hat{\mathbf{s}}$, obtained from the sin-screw by the simple step of dividing by $\cos\frac{1}{2}\hat{\theta}$ [3, 9, 11].

To show the ease of creating screw-forms, a new screw-form defined in terms of the *full-* rather than *half-*dual angle, viz. $\hat{\mathcal{s}} = \sin\hat{\theta}\,\hat{\mathbf{s}}$, characterised by pitch $P_{\hat{\mathcal{s}}} = d/\tan\theta$, is used in a typical kinematic context.

2 Notation and Basic Geometry

We write a *screw* $\hat{\mathbf{S}}$ as a 3-vector of dual numbers

$$\left.\begin{aligned} &\hat{\mathbf{S}} = |\hat{\mathbf{S}}|\,(1{+}\varepsilon\, p)\,\hat{\mathbf{s}},\ \ \hat{\mathbf{s}} = \mathbf{l}{+}\varepsilon\,\mathbf{M} \\ &\hat{\mathbf{s}}^{\mathbf{2}} = \mathbf{l}^{\mathbf{2}}{+}\varepsilon\,2\,\mathbf{l}\cdot\mathbf{M} = 1{+}\varepsilon\,0,\ \ \mathbf{l}\times\mathbf{M} = \mathbf{R} \end{aligned}\right\} \tag{1}$$

in which ε is a quasi-scalar such that $(a{+}\varepsilon\, b = c{+}\varepsilon\, d) \Leftrightarrow (a = c) \wedge (b = d)$ for all real a, b, c, and d, and satisfying $\varepsilon^2 = 0$. $|\hat{\mathbf{S}}|$ is the real *magnitude* and p is the real *pitch* of the screw $\hat{\mathbf{S}}$, and $\hat{\mathbf{s}}$ (written in lower case) is its *normalised line* which, regarded as a screw in its own right, has unit magnitude and zero pitch. The line $\hat{\mathbf{s}}$ of the screw is spatially located by the *direction 3-vector* of direction cosines $\mathbf{l} = (l,\ m,\ n)$, and by the *moment 3-vector* $\mathbf{M} = (P,\ Q,\ R)$ which determines its *origin-radius* 3-vector $\mathbf{R}$.

Two screws $\hat{\mathbf{S}}_\mathbf{1}, \hat{\mathbf{S}}_\mathbf{2}$ are *perpendicular* if $\mathbf{l}_\mathbf{1}\cdot\mathbf{l}_\mathbf{2} = 0$, and *orthogonal* if $\hat{\mathbf{S}}_\mathbf{1}\cdot\hat{\mathbf{S}}_\mathbf{2} = 0$, which implies that each intersects the other at right angles. The *cross product* $\hat{\mathbf{S}}_\mathbf{1}\times\hat{\mathbf{S}}_\mathbf{2}$ is sited in their *common perpendicular*.

We represent the typical right-handed reference frame by orthogonal normalised axial lines $\hat{\mathbf{x}}_\mathbf{i}$, $\hat{\mathbf{y}}_\mathbf{i}$, and $\hat{\mathbf{z}}_\mathbf{i}$ for which

$$\left.\begin{aligned} &\hat{\mathbf{x}}_\mathbf{i}\cdot\hat{\mathbf{y}}_\mathbf{i} = \hat{\mathbf{y}}_\mathbf{i}\cdot\hat{\mathbf{z}}_\mathbf{i} = \hat{\mathbf{z}}_\mathbf{i}\cdot\hat{\mathbf{x}}_\mathbf{i} = 0,\ \ \hat{\mathbf{x}}_\mathbf{i}^{\mathbf{2}} = \hat{\mathbf{y}}_\mathbf{i}^{\mathbf{2}} = \hat{\mathbf{z}}_\mathbf{i}^{\mathbf{2}} = 1, \\ &\hat{\mathbf{x}}_\mathbf{i}\times\hat{\mathbf{y}}_\mathbf{i} = \hat{\mathbf{z}}_\mathbf{i},\ \ \hat{\mathbf{y}}_\mathbf{i}\times\hat{\mathbf{z}}_\mathbf{i} = \hat{\mathbf{x}}_\mathbf{i},\ \ \hat{\mathbf{z}}_\mathbf{i}\times\hat{\mathbf{x}}_\mathbf{i} = \hat{\mathbf{y}}_\mathbf{i}. \end{aligned}\right\} \tag{2}$$

If, in some common frame, we know a general screw $\hat{\mathbf{S}}$ and the i-frame axes $\hat{\mathbf{x}}_\mathbf{i}$, $\hat{\mathbf{y}}_\mathbf{i}$, $\hat{\mathbf{z}}_\mathbf{i}$, we *transform* that screw into i-frame coordinates by

$$\hat{\mathbf{S}}_\mathbf{i} = \begin{bmatrix} \hat{\mathbf{x}}_\mathbf{i}^{\mathbf{T}} \\ \hat{\mathbf{y}}_\mathbf{i}^{\mathbf{T}} \\ \hat{\mathbf{z}}_\mathbf{i}^{\mathbf{T}} \end{bmatrix} \hat{\mathbf{S}} = \begin{bmatrix} \hat{\mathbf{x}}_\mathbf{i}\cdot\hat{\mathbf{S}} \\ \hat{\mathbf{y}}_\mathbf{i}\cdot\hat{\mathbf{S}} \\ \hat{\mathbf{z}}_\mathbf{i}\cdot\hat{\mathbf{S}} \end{bmatrix}$$

in which the first-written matrix is 3×3 dual orthogonal.

3 Specification of a Finite Displacement Screw

We represent the general finite displacement of a body – comprising *translation* through distance d and *rotation* through angle θ, $-\pi < \theta \leq \pi$, about the unit *screw axis* $\hat{\mathbf{s}}$ ($\hat{\mathbf{s}}^{\mathbf{2}} = 1$) – by constructing the dual angle

$$\tfrac{1}{2}\hat{\theta} = \tfrac{1}{2}\theta + \varepsilon\, \tfrac{1}{2}d \quad \text{so that} \quad \sin \tfrac{1}{2}\hat{\theta} = \sin \tfrac{1}{2}\theta + \varepsilon\, \tfrac{1}{2}d \cos \tfrac{1}{2}\theta,$$

and by then writing the *sin-screw*

$$\hat{\mathbf{S}} = \sin \tfrac{1}{2}\hat{\theta}\, \hat{\mathbf{s}} = \sin \tfrac{1}{2}\theta\, (1 + \varepsilon\, P_S)\, \hat{\mathbf{s}} \quad \text{where} \quad P_S = \tfrac{1}{2}d / \tan \tfrac{1}{2}\theta. \tag{3}$$

The sin-screw resultant, $\hat{\mathbf{S}}$, of successively applying two such screws, first $\hat{\mathbf{S}}_{\mathbf{1}} = \sin \frac{1}{2}\hat{\theta}_1\, \hat{\mathbf{s}}_{\mathbf{1}}$ and then $\hat{\mathbf{S}}_{\mathbf{2}} = \sin \frac{1}{2}\hat{\theta}_2\, \hat{\mathbf{s}}_{\mathbf{2}}$, is conveniently written

$$\begin{bmatrix} \cos \frac{1}{2}\hat{\theta} \\ \hat{\mathbf{S}} \end{bmatrix} = \begin{bmatrix} \cos \frac{1}{2}\hat{\theta}_1 \cos \frac{1}{2}\hat{\theta}_2 \; - \; \hat{\mathbf{S}}_{\mathbf{1}} \cdot \hat{\mathbf{S}}_{\mathbf{2}} \\ \cos \frac{1}{2}\hat{\theta}_2\, \hat{\mathbf{S}}_{\mathbf{1}} \; + \; \cos \frac{1}{2}\hat{\theta}_1\, \hat{\mathbf{S}}_{\mathbf{2}} \; - \; \hat{\mathbf{S}}_{\mathbf{1}} \times \hat{\mathbf{S}}_{\mathbf{2}} \end{bmatrix}, \tag{4}$$

which comprises the *biquaternion product* rule [1].

An alternative form of finite displacement screw, which derives from the sin-screw on division by $\cos \frac{1}{2}\hat{\theta}$, is the *tan-screw*, written [11]

$$\hat{\mathbf{T}} = \tan \tfrac{1}{2}\hat{\theta}\, \hat{\mathbf{s}} = \tan \tfrac{1}{2}\theta\, (1 + \varepsilon\, P_T)\, \hat{\mathbf{s}} \quad \text{where} \quad P_T = d / \sin \theta. \tag{5}$$

Such manipulation of Eq. (4) yields the corresponding tan-screw resultant [11] of applying two tan-screws, first $\hat{\mathbf{T}}_{\mathbf{1}}$, then $\hat{\mathbf{T}}_{\mathbf{2}}$, viz.

$$\hat{\mathbf{T}} = \frac{\hat{\mathbf{T}}_{\mathbf{1}} \; + \; \hat{\mathbf{T}}_{\mathbf{2}} \; - \; \hat{\mathbf{T}}_{\mathbf{1}} \times \hat{\mathbf{T}}_{\mathbf{2}}}{1 \; - \; \hat{\mathbf{T}}_{\mathbf{1}} \cdot \hat{\mathbf{T}}_{\mathbf{2}}}. \tag{6}$$

Using the sin-screw form of Eq. (3) it is found, when a body *with spatial symmetries of figure* is relocated – or, equivalently, when a displacement is *incompletely specified* in some coordinate(s) – that the (possibly infinite) set of screws available to the body in achieving the relocation is described by linear combination of a small basis of screws [2, 6, 7]. These findings, each dealing with particular kinematic instances, have been generalised to treatment of the *symmetry screws* of *any* body shape, and have been shown to extend equally to unit biquaternions in linear combinations [8,10]. This generalisation has been re-expressed in terms of the tan-screw form of Eq. (5) [9].

4 Generalisation to Other Screw Forms

We now show that this property – the capacity to form kinematically significant linearly combined sets – is by no means retricted to sin-screws and tan-screws, or to screws of their pitch-forms. For suppose that some kinematic situation (such as those cited above) is described by such a set of sin-screws so that the typical screw of the set is characterised by the dual-linear form

$$\hat{\mathbf{S}} = \sin \tfrac{1}{2}\hat{\theta}\, \hat{\mathbf{s}} = \hat{\mathcal{X}}(\ldots)\, \hat{\mathbf{x}} \;+\; \hat{\mathcal{Y}}(\ldots)\, \hat{\mathbf{y}} \;+\; \hat{\mathcal{Z}}(\ldots)\, \hat{\mathbf{z}}, \tag{7}$$

where $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, $\hat{\mathbf{z}}$ are mutually intersecting orthogonal unit lines for which

$$\hat{\mathbf{x}} \cdot \hat{\mathbf{y}} = \hat{\mathbf{y}} \cdot \hat{\mathbf{z}} = \hat{\mathbf{z}} \cdot \hat{\mathbf{x}} = 0, \;\; \hat{\mathbf{x}}^{\mathbf{2}} = \hat{\mathbf{y}}^{\mathbf{2}} = \hat{\mathbf{z}}^{\mathbf{2}} = 1,$$

and the dual-valued coefficients $\hat{\mathcal{X}}(\ldots)$, $\hat{\mathcal{Y}}(\ldots)$, $\hat{\mathcal{Z}}(\ldots)$ are functions of one or more spatial variables applying in the particular kinematic context.

Since $\hat{\mathbf{s}}^{\mathbf{2}} = 1$, from Eq. (7) we readily derive

$$\sin^2 \tfrac{1}{2}\hat{\theta} = \hat{\mathcal{X}}^2(\ldots) + \hat{\mathcal{Y}}^2(\ldots) + \hat{\mathcal{Z}}^2(\ldots). \tag{8}$$

If we assume that the sign of $\sin \frac{1}{2}\theta$ has been incorporated into the direction specified for the line $\hat{\mathbf{s}}$, so that we may safely adopt the positive square root of $\sin^2 \frac{1}{2}\theta$ wherever it occurs, we may multiply both sides of Eq. (7) by arbitrary powers g, h of $\sin \frac{1}{2}\hat{\theta}$, $\cos \frac{1}{2}\hat{\theta}$ to obtain

$$\begin{aligned} \sin^{g+1} \tfrac{1}{2}\hat{\theta} \cos^h \tfrac{1}{2}\hat{\theta}\, \hat{\mathbf{s}} = {} & \left\{\hat{\mathcal{X}}^2() + \hat{\mathcal{Y}}^2() + \hat{\mathcal{Z}}^2()\right\}^{g/2} \left\{1 - \hat{\mathcal{X}}^2() - \hat{\mathcal{Y}}^2() - \hat{\mathcal{Z}}^2()\right\}^{h/2} \\ & \times \left\{\hat{\mathcal{X}}()\, \hat{\mathbf{x}} \;+\; \hat{\mathcal{Y}}()\, \hat{\mathbf{y}} \;+\; \hat{\mathcal{Z}}()\, \hat{\mathbf{z}}\right\}, \end{aligned}$$

which, on the right-hand side, is again a linear combination of the same orthogonal basis screws with coefficient functions parameterised in the same spatial variables. The item on the left, when dual angles are expanded, takes the form

$$\begin{aligned} & \sin^{g+1} \tfrac{1}{2}\theta \left\{1 + \varepsilon\, \tfrac{1}{2} d \cot \tfrac{1}{2}\theta\right\}^{g+1} \cos^h \tfrac{1}{2}\theta \left\{1 - \varepsilon\, \tfrac{1}{2} d \tan \tfrac{1}{2}\theta\right\}^h \hat{\mathbf{s}} \\ & \quad = \sin^{g+1} \tfrac{1}{2}\theta \cos^h \tfrac{1}{2}\theta \left\{1 + \varepsilon\, \tfrac{1}{2} d \left\{(g+1) \cot \tfrac{1}{2}\theta - h \tan \tfrac{1}{2}\theta\right\}\right\} \hat{\mathbf{s}}, \end{aligned}$$

which is a screw of pitch $p = \frac{1}{2} d \{(g+1) \cot \frac{1}{2}\theta - h \tan \frac{1}{2}\theta\}$.

We may, further, observe that sums of such terms, with arbitrarily chosen exponents g_i and h_i, and combined with arbitrarily chosen dual-valued functions $C_i()$, of the same spatial parameters, such as

$$\begin{aligned} & \textstyle\sum_i C_i() \sin^{g_i+1} \tfrac{1}{2}\hat{\theta} \cos^{h_i} \tfrac{1}{2}\hat{\theta}\, \hat{\mathbf{s}} \\ & \quad = \textstyle\sum_i C_i() \left\{\hat{\mathcal{X}}^2() + \hat{\mathcal{Y}}^2() + \hat{\mathcal{Z}}^2()\right\}^{g_i/2} \left\{1 - \hat{\mathcal{X}}^2() - \hat{\mathcal{Y}}^2() - \hat{\mathcal{Z}}^2()\right\}^{h_i/2} \\ & \quad \times \left\{\hat{\mathcal{X}}()\, \hat{\mathbf{x}} \;+\; \hat{\mathcal{Y}}()\, \hat{\mathbf{y}} \;+\; \hat{\mathcal{Z}}()\, \hat{\mathbf{z}}\right\}, \end{aligned} \tag{9}$$

yield similar linear combinations.

We may, therefore, synthesise an arbitrarily large set of functions $\hat{\mathcal{F}}(\hat{\theta})$ of the dual angle $\hat{\theta}$ to serve as multipliers of the screw-axis line $\hat{\mathbf{s}}$ in screw forms $\hat{\mathcal{F}}(\hat{\theta})\,\hat{\mathbf{s}}$ which can be expressed as sets of screws deriving as linear combinations of the chosen basis.

5 The Full-Angle Sin Screw

To exemplify the results of the preceding section we now introduce a new screw form – an "$\hat{\mathcal{S}}$-screw" – defined in terms of the *full* dual angle of displacement; thus, in terms of quantities defined in Section 3,

$$\hat{\mathcal{S}} = \sin\hat{\theta}\,\hat{\mathbf{s}} = \sin\theta\,(1+\varepsilon\,P_{\hat{\mathcal{S}}})\,\hat{\mathbf{s}} \quad \text{where} \quad P_{\hat{\mathcal{S}}} = d/\sin\theta. \tag{10}$$

(whereas the definition of $\hat{\mathbf{S}}$ at Eq. (3) involved the dual *half*-angle). Since $\hat{\mathcal{S}} = 2\cos\frac{1}{2}\hat{\theta}\,\hat{\mathbf{S}}$, we may write the resultant, $\hat{\mathcal{S}}$, of successively applying two such screws, first $\hat{\mathcal{S}}_1 = \sin\hat{\theta}_1\,\hat{\mathbf{s}}_\mathbf{1}$ and then $\hat{\mathcal{S}}_2 = \sin\hat{\theta}_2\,\hat{\mathbf{s}}_\mathbf{2}$, as the doubled product of the two entries on the right in Eq. (4), viz.

$$\begin{aligned}\hat{\mathcal{S}} = \cos\hat{\theta}_2\,\hat{\mathcal{S}}_1 + \cos\hat{\theta}_1\,\hat{\mathcal{S}}_2 - \tfrac{1}{2}\left[1 - \frac{\hat{\mathcal{S}}_1\cdot\hat{\mathcal{S}}_2}{(\cos\hat{\theta}_1+1)\,(\cos\hat{\theta}_2+1)}\right]\hat{\mathcal{S}}_1\times\hat{\mathcal{S}}_2 \\ -\tfrac{1}{2}\left[\frac{\hat{\mathcal{S}}_1}{\cos\hat{\theta}_1+1} - \frac{\hat{\mathcal{S}}_2}{\cos\hat{\theta}_2+1}\right]\times(\hat{\mathcal{S}}_1\times\hat{\mathcal{S}}_2).\end{aligned} \tag{11}$$

To provide an exemplary set of $\hat{\mathcal{S}}$-screws in linear combination, we could adapt a general method used elsewhere [8–10] in order to generate them directly from the *symmetry* $\hat{\mathcal{S}}$*-screws* of a body undergoing displacement. But the visible growth in complexity of Eq. (11) – a necessary component in the development – when it is contrasted with its analogue in Eq. (4), makes this a tortuous course to follow.

Instead, we shall broadly follow the prescription of the preceding section. Having outlined the known solution to a particular kinematic context as it is expressed in sin-screws $\hat{\mathbf{S}}$, we shall convert those sin-screws to $\hat{\mathcal{S}}$-screws by multiplying by an expression for $2\cos\frac{1}{2}\hat{\theta}$.

6 Finite Displacement of a Body with Symmetries

We will restrict attention to properties of the *point-line* object, for which the set of *symmetry screws* – the totality of finite displacements (screws) which leave the the object invariant – consists of all possible pure rotations of the object about the line component i.e. all sin-screws of the form

$$\hat{\mathbf{S}} = \sin\psi\,\hat{\mathbf{l}}, \quad -\pi < \psi \leq \pi, \tag{12}$$

where $\hat{\mathbf{l}}$ is the unit line component of the point-line object.

However, it will better serve to provide geometric context and to illustrate the provenance of many linearly combined sets of screws if we consider the symmetry screws of a body to be written more generally as

$$\hat{\mathbf{S}} = \hat{L}\,\hat{\mathbf{i}} + \hat{M}\,\hat{\mathbf{j}} + \hat{N}\,\hat{\mathbf{k}},$$

where $\hat{L}$, $\hat{M}$, $\hat{N}$ are dual-valued coefficient functions and $\hat{\mathbf{i}}$, $\hat{\mathbf{j}}$, $\hat{\mathbf{k}}$ are orthonormal axial lines fixed in the body.

Consider that we observe such a body to undergo a relocation which – because its symmetries restrict our ability to distinguish apparently equivalent locations – is, in effect, an *incompletely specified displacement*. Our goal is to identify all finite displacement screws which are capable of producing the observed relocation. We may generate the typical screw

- by, firstly, applying some particular displacement (screw) that carries the body from its initial location into one of its symmetrically equivalent final locations; this, so called, *pilot screw* is singled out for this role only and is not otherwise distinguished among the screws effective for the displacement;
- by then applying to the body – in that final location – a typical member of the set of *symmetry screws* which leaves the body apparently invariant by carrying it into a symmetrically equivalent final location.

Thus, every screw effective for the observed relocation may be obtained as the *resultant* of applying the *pilot screw* and one such *symmetry screw*.

Let the pilot displacement comprise translation d_Z and rotation θ_Z, $-\pi < \theta_Z \leq \pi$, about a unit line $\hat{\mathbf{z}}$ so, by Eq. (10), the pilot screw is

$$\hat{\mathbf{S}}_\mathbf{Z} = \sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{z}} \quad \text{where} \quad \hat{\theta}_Z = \theta_Z + \varepsilon\, d_Z.$$

We adopt $\hat{\mathbf{z}}$ and two further lines $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$ – which together satisfy orthonormality conditions of Eqs. (2) – as the reference frame for all results.

Now the pilot displacement carries this xyz-frame, embedded in the body, from an *initial location* $\hat{\mathbf{x}}_\mathbf{i}$, $\hat{\mathbf{y}}_\mathbf{i}$, $\hat{\mathbf{z}}_\mathbf{i}$ to a *final location* $\hat{\mathbf{x}}_\mathbf{f}$, $\hat{\mathbf{y}}_\mathbf{f}$, $\hat{\mathbf{z}}_\mathbf{f}$ with the z-axis $\hat{\mathbf{z}} = \hat{\mathbf{z}}_\mathbf{i} = \hat{\mathbf{z}}_\mathbf{f}$ in common. It greatly simplifies later working to define the orthonormal axes $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, $\hat{\mathbf{z}}$ to be the *mean lines* of those extremum axes, which are then expressed in that reference location by

$$\begin{aligned}
\hat{\mathbf{x}}_\mathbf{i} &= \cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{x}} - \sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{y}}, & \hat{\mathbf{x}}_\mathbf{f} &= \cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{x}} + \sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{y}},\\
\hat{\mathbf{y}}_\mathbf{i} &= \sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{x}} + \cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{y}}, & \hat{\mathbf{y}}_\mathbf{f} &= -\sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{x}} + \cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{y}},\\
\hat{\mathbf{z}}_\mathbf{i} &= \hat{\mathbf{z}}, & \hat{\mathbf{z}}_\mathbf{f} &= \hat{\mathbf{z}}.
\end{aligned}$$

We may now evaluate the resultant of applying the *pilot screw*,

$$\hat{\mathbf{S}}_\mathbf{1} = \sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{z}},$$

and the *symmetry screw* $(\hat{L},\ \hat{M},\ \hat{N})$ as expressed at the *final location*,

$$\hat{\mathbf{S}}_{\mathbf{2}} = \hat{L}(\cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{x}}+\sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{y}})-\hat{M}(\sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{x}}-\cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{y}})+\hat{N}\hat{\mathbf{z}},$$

for which $\sin^2\frac{1}{2}\hat{\theta}_2 = \hat{L}^2+\hat{M}^2+\hat{N}^2$, so that $\cos\frac{1}{2}\hat{\theta}_2 = \sqrt{1-\hat{L}^2-\hat{M}^2-\hat{N}^2}$. For the resultant sin-screw, by use of Eq. (4) we find

$$\cos\tfrac{1}{2}\hat{\theta} = \sqrt{1-\hat{L}^2-\hat{M}^2-\hat{N}^2}\cos\tfrac{1}{2}\hat{\theta}_Z-\hat{N}\sin\tfrac{1}{2}\hat{\theta}_Z, \tag{13}$$

$$\hat{\mathbf{S}} = \hat{L}\hat{\mathbf{x}}+\hat{M}\hat{\mathbf{y}}+\left[\sqrt{1-\hat{L}^2-\hat{M}^2-\hat{N}^2}\sin\tfrac{1}{2}\hat{\theta}_Z+\hat{N}\cos\tfrac{1}{2}\hat{\theta}_Z\right]\hat{\mathbf{z}}, \tag{14}$$

the second of which shows, in its generalised terms, the provenance of many linearly combined sets of screws of the kind considered in this paper.

7 Half-Angle Sin Screws for Displacement of the Point-Line

We can now make these results specific to the *point-line* symmetry object. Within the displacing xyz-frame, we consider the *line*-component of the element to lie parallel with the $\hat{\mathbf{y}}$-axis. Generality is lost if the *line* is constrained to lie *on* the $\hat{\mathbf{y}}$-axis since some *point* of the line is then required – atypically – to traverse the screw axis $\hat{\mathbf{z}}$ itself during the course of the displacement. Instead, we specify that the *line*-component intersects the axis $\hat{\mathbf{x}}$ in a point at distance τ from $\hat{\mathbf{y}}$, and we adopt that point as the *point*-component.

So located, the symmetry screws of the point-line in Eq. (12) are expressed in the functional forms

$$\begin{bmatrix}\hat{L}\\ \hat{M}\\ \hat{N}\end{bmatrix} = \sin\psi\begin{bmatrix}0\\ 1\\ \varepsilon\,\tau\end{bmatrix},$$

in which the real parameter ψ, $-\pi < \psi \leq \pi$, may be arbitrarily chosen.

Since $\hat{L}^2+\hat{M}^2+\hat{N}^2 = \sin^2\psi$, so that $\sqrt{1-\hat{L}^2-\hat{M}^2-\hat{N}^2} = \cos\psi$, with these values Eqs. (13, 14) simplify to:

$$\cos\tfrac{1}{2}\hat{\theta} = \cos\psi\cos\tfrac{1}{2}\hat{\theta}_Z\ -\ \varepsilon\,\tau\sin\psi\sin\tfrac{1}{2}\hat{\theta}_Z, \tag{15}$$

$$\hat{\mathbf{S}} = \sin\psi\left[\hat{\mathbf{y}}\ +\ \varepsilon\,\tau\cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{z}}\right]\ +\ \cos\psi\sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{z}}, \tag{16}$$

the second of which shows the *two-system* [5] of screws expected for the displacement of a point-line [7], as generated by linear combination of the mutually perpendicular, but not intersecting, sin-screws

$$\hat{\mathbf{y}}\ +\ \varepsilon\,\tau\cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{z}},$$
$$\sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{z}},$$

with the real coefficient functions $\sin\psi$, $\cos\psi$ parameterised by the variable ψ, $-\pi < \psi \leq \pi$.

This basis pair, found earlier as both sin- and tan-screws [8,9], has been made the subject of a theorem in a recent treatment of the point-line situation which adopts their directions, and the mid-point between them, as defining a *canonical system* for this kinematic context [12]. This adoption is not based on any fundamental characteristic of the screws which distinguishes them from any others in the two-system, but by the convenience for the human observer of indentifying the locations of those particular screws within the physical reality of a practical situation.

At the values $\sin\psi = \pm\sin\frac{1}{2}\hat{\theta}_Z/\sqrt{1+\sin^2\frac{1}{2}\hat{\theta}_Z}$, $\cos\psi = 1/\sqrt{1+\sin^2\frac{1}{2}\hat{\theta}_Z}$, the two central *principal screws* [5] of the two-system are selected, viz.

$$\hat{\mathbf{S}}_{\pm} = \frac{\sin\frac{1}{2}\hat{\theta}_Z}{\sqrt{1+\sin^2\frac{1}{2}\hat{\theta}_Z}}\Big\{\pm\big[\hat{\mathbf{y}} + \varepsilon\,\tau\cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{z}}\big] + \hat{\mathbf{z}}\Big\}, \tag{17}$$

which are orthogonal, so that $\hat{\mathbf{S}}_+ \cdot \hat{\mathbf{S}}_- = 0$, and therefore intersect one another at right angles. In terms of these, the general member screw of Eq. (16) may be restated, thus

$$\hat{\mathbf{S}} = \frac{\sqrt{1+\sin^2\frac{1}{2}\hat{\theta}_Z}}{2}\left\{\left[\cos\psi+\frac{\sin\psi}{\sin^2\frac{1}{2}\hat{\theta}_Z}\right]\hat{\mathbf{S}}_+ + \left[\cos\psi-\frac{\sin\psi}{\sin^2\frac{1}{2}\hat{\theta}_Z}\right]\hat{\mathbf{S}}_-\right\}.$$

Normalistion aside, this equation typifies the linear combination of orthogonal basis screws with (more usually) dual-valued coefficient functions which is represented in general form at Eq. (7).

8 Full-Angle Screws for Displacement of the Point-Line

As proposed earlier, we form the full-angle $\hat{\mathcal{S}}$-screws for displacement of the point-line object by multiplying the $\cos\frac{1}{2}\hat{\theta}$ expression of Eq. (15) into the half-angle $\hat{\mathbf{S}}$-screw expression of Eq. (16), and doubling: in some respects, the least-rearranged outcome, viz.

$$\begin{aligned}\hat{\mathcal{S}} = {} & (\sin 2\psi\cos\tfrac{1}{2}\hat{\theta}_Z - \varepsilon\,2\tau\sin^2\psi\sin\tfrac{1}{2}\hat{\theta}_Z)\big[\hat{\mathbf{y}} + \varepsilon\,\tau\cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{z}}\big] \\ & + (\cos^2\psi\sin\hat{\theta}_Z - \varepsilon\,\tau\sin 2\psi\sin^2\tfrac{1}{2}\hat{\theta}_Z)\,\hat{\mathbf{z}},\end{aligned} \tag{18}$$

is the most informative in that it preserves the identity of the lines of the basis screws – which, of course, have not changed – while revealing that the coefficients are no longer purely real.

The attempt, at those same basis screws, to allow a number of inherent references to the full-angle $\hat{\theta}_Z$ to express themselves, as represented in the rearrangement:

$$\hat{\mathcal{S}} = \sin 2\psi\left[\cos\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{y}} + \varepsilon\,\tau\cos\hat{\theta}_Z\,\hat{\mathbf{z}}\right] + \cos^2\psi\sin\hat{\theta}_Z\,\hat{\mathbf{z}} - 2\varepsilon\,\tau\sin^2\psi\sin\tfrac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{y}}, \tag{19}$$

not only leaves the half-angle $\frac{1}{2}\hat{\theta}_Z$ still much in evidence but also splits the identity of the basis screw $\hat{\mathbf{y}}+\varepsilon\,\tau\cos\frac{1}{2}\hat{\theta}_Z\,\hat{\mathbf{z}}$ into two less comprehensible portions. This unattractive situation is not appreciably changed if the $\hat{\mathcal{S}}$-screws are re-expressed in terms of the central orthogonal *principal screw* basis of Eq. (17).

So, while we have demonstrated the feasibility, and ease, of expressing a chosen kinematic situation in terms of full-angle $\hat{\mathcal{S}}$-screws, we do not find the representation which they offer to be as directly informative as that found earlier for half-angle sin-screws at Eq. (16).

9 Conclusion

It has been shown that in any kinematic situation which is described by a set of sin-screws in linear combination, whether with real or dual coefficients, screws of indefinitely many other functional forms may be derived, each with a correspondingly distinct definition of pitch, which in the same kinematic situation will also form sets of screws that are linearly combined, with – generally – dual coefficients.

It is clear that the scope of the demonstration could have been broadened: sin-screws were adopted as the base type from which other screw-forms might be derived because they have been central to most discussions of finite displacement screws in the recent literature; a focus was maintained on derivation of trigonometrically-related functional forms, rather than arbitrarily general forms, because these appear to be the most relevant to current and future work and might, in any case, lead to quite general forms by way of Fourier synthesis.

As stated at the outset, the purpose was not to proffer new screw forms for use but to indicate where connections may lie between screw- and pitch-forms newly arising in analysis and those which have been used before.

Acknowledgement

The author acknowledges the assistance of the School of Information Technologies, the University of Sydney, in preparation of this paper.

References

1. Bottema, O. and Roth, B., *Theoretical Kinematics*, North-Holland Publishing Company, Amsterdam (1979). Reprinted Dover, New York (1990).
2. Huang, C. and Roth, B., Analytic expressions for the finite screw systems. *Mechanism and Machine Theory* **29**, 207–222 (1994).
3. Huang, C. and Wang, J-C., The finite screw system associated with the displacement of a line. *Journal of Mechanical Design*, *Trans. ASME* **125**, 105–109 (2003).
4. Huang, C., Kuo, W. and Ravani, B., On the linear line complex and helicoidal vector field associated with homologous lines of a finite displacement. In *Proceedings 12th IFToMM World Congress on the Theory of Machines and Mechanisms*, Besançon, France, Paper CK541, pp. 6 (2007).
5. Hunt, K.H., *Kinematic Geometry of Mechanisms.* Clarendon Press, Oxford (1990).
6. Hunt, K.H. and Parkin, I.A., Finite displacements of points, planes and lines *via* screw theory. *Mechanism and Machine Theory* **30**, 177–192 (1995).
7. Parkin, I.A., A third conformation with the screw systems: finite twist displacements of a directed line and point. *Mechanism and Machine Theory* **27**, 177–188 (1992).
8. Parkin, I.A., The role of body symmetry in determining screws for finite displacement of a rigid body. In *Proceedings NATO ASI on Computational Methods in Mechanisms* Vol. 2, Varna, Bulgaria, June 16–28, pp. 145–154 (1997).
9. Parkin, I.A., Linear systems of tan-screws for finite displacement of a rigid body with symmetries. In *Proceedings Sixth International Workshop on Advances in Robot Kinematics*, Strobl, Austria, June–July, J. Lenarčič and M.L. Husty (Eds.), Kluwer Academic Publishers, pp. 317–326 (1998).
10. Parkin, I.A., The screws for finite displacement of a rigid body expressed in terms of its symmetry screws. In *Proceedings of CK2005, International Workshop on Computational Kinematics*, Cassino, Italy, May, Paper 08CK2005, pp. 15 (2005).
11. Yang, A.T., Calculus of screws. In *Basic Questions of Design Theory*, W.R. Spillers (Ed.), Elsevier, New York, pp. 265–281 (1974).
12. Zhang, Y. and Ting, K-L., On the basis screws and screw systems of point-line and line displacements. *Journal of Mechanical Design*, *Trans. ASME* **126**, 56–62 (2004).

MOTION PLANNING AND MOBILITY

Time-Invariant Strategies in Coordination of Human Reaching

Satyajit Ambike and James P. Schmiedeler

Department of Mechanical Engineering, The Ohio State University, Columbus, OH 43210, U.S.A., e-mail: {ambike.1, schmiedeler.2}@osu.edu

Abstract. This paper addresses validation of a curvature-theory-based time-invariant inverse kinematic model and a related tracking algorithm for human motor control of reaching motions. Human subjects made unconstrained reaching motions in the horizontal plane to fixed targets at three self-selected speeds. Consistent shoulder/elbow joint speed ratios for motions to the same target across speeds were observed, indicating a time-invariant planning strategy. The inverse kinematic model's technique of relating joint motions with a Taylor series expansion is in concert with the leading joint hypothesis. With this approach the tracking algorithm successfully replicated the experimental wrist trajectories, and also predicted the previously observed elbow-led motions for reaching in the ipsilateral hemifield. The elbow leads the arm motion in this hemifield because the shoulder approaches a dwell. A computationally frugal strategy of intermittent path correction based on two error parameters is proposed.

Key words: curvature theory, motor control, arm kinematics, leading joint hypothesis.

1 Introduction

It has been proposed that the fundamental internal model employed by the human central nervous system (CNS) for motor coordination of reaching motions is a time-invariant inverse kinematic model [1] that can be mathematically described using two-degree-of-freedom curvature theory for planar motions [8]. This mathematical model matches the instantaneous geometric properties of the desired wrist path to the instantaneous geometric properties of a corresponding path in the joint space of the mechanism, i.e. the human arm. The elbow and shoulder joint motions are coordinated through a Taylor series expansion such that one joint leads, and the other is subordinate. This approach is implemented to track paths of arbitrary length within a planar mechanism's workspace via a tracking algorithm [2]. The use of the Taylor series to relate joint motions echoes the recent leading joint hypothesis (LJH) [5] which states that in multi-joint limb motion one joint, usually the proximal joint, acts as a leading joint by moving independent of the other joints. The subordinate joints then move such that the required end-effector motion is achieved. This hypothesis grew in part from the work of Levin et al. [7] and Dounskaia et al. [4],

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 205–214.

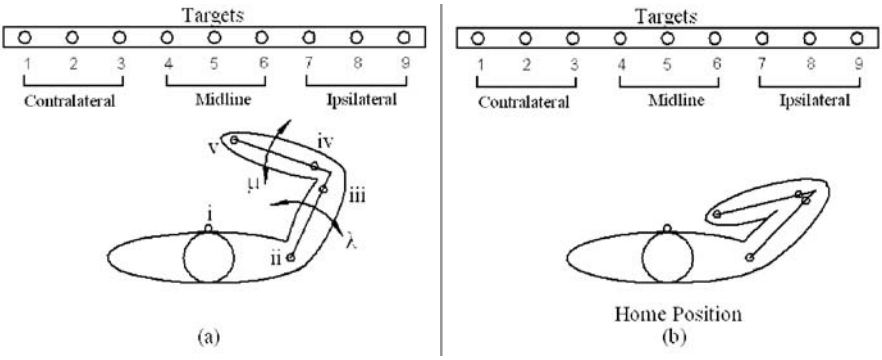


Fig. 1 Schematic of experiment; (a) sensor locations with roman numerals – λ and μ indicate upper and forearm angular displacements from their current positions; (b) the home position.

who also found that the shoulder is the leading joint for most hand movements in the horizontal plane. Movements in the the ipsilateral hemifield are elbow-led. This paper analyzes data from horizontal reaching motions performed by healthy human subjects to validate the time-invariant model and the path-tracking algorithm.

2 Experimental Methods

Thirteen subjects with no history of neurological injury participated in this study, complete details of which are in [3]. As shown in Figure 1, subjects sat upright facing a set of nine targets (incandescent lights) arranged equidistantly along a horizontal line. The targets were positioned with the fifth light in the subject's mid-line sagittal plane. An electromagnetic spatial tracking system recorded arm motions with sensors placed at the sternum, shoulder (scapula), lower bicep, upper forearm, and wrist (midpoint between the ulnar and radial styloids).

Each test began with the subject's arm in a horizontal home position as shown in Figure 1b, with the elbow bent to place the tip of the index finger near the sternum marker (marker *i* in Figure 1a). A single target was illuminated, followed by an audio cue of 'slow', 'normal', or 'fast', indicating the speed with which to reach toward the target. Normal indicated the subject's natural speed, while slow and fast indicated exaggerated deviations from normal. The subject extended the arm to point at the target and returned to the home position. Subjects were instructed to maintain the horizontal orientation of the arm during the motion and to consistently use the same three speeds throughout the experiment. The sequence of target illumination and the speed cues was predefined, but unknown and seemingly random to the subject. The subjects were unrestrained, but were instructed to keep their torsos as steady as possible, especially for contralateral reaching. The experiment ended when the subject had pointed to all nine lights at all three speeds, first with the right hand and then with the left, yielding 27 tests for each hand.

3 Model and Algorithm Validation

The following analysis aims to validate the hypothesis that a two-degree-of-freedom (DOF) curvature-theory model [8], along with the authors' tracking algorithm [2] can describe the internal inverse-kinematic model used by the CNS for planar arm motions. The algorithm enables planar path tracking with two-DOF planar mechanisms. Consequently, the human arm is modeled as a planar revolute-revolute (RR) mechanism, with the shoulder and elbow being parallel-axis revolutes. A planar trajectory best representing the 3-dimensional wrist (marker v in Figure 1a) data, called the *experimental path*, is first obtained. The validation is done by demonstrating that a fourth-order polynomial fit to the experimental path, called the *desired path*, can be tracked using the algorithm. The algorithm tracks the desired path using the equivalent RR mechanism, the parameters of which, viz. the arm segment lengths and the fixed shoulder location, are input to the algorithm. These are obtained by analyzing the marker data from each test. The arm lengths for a particular subject are constant, but the shoulder location varies for each test, as described below. The tracking results indicate that the algorithm can effectively track the desired paths. The experimental speed ratios, defined below, are obtained from the experimental path via inverse kinematics, and the algorithm generates the modeling speed ratios based on the desired path. Comparison of the experimental and modeling speed ratios provides corroboration for the LJH.

3.1 Data Analysis and Experimental Speed Ratios

This section describes the methodology to obtain planar experimental and desired paths, the equivalent RR mechanism for each test, and the joint trajectories and speed ratios for tracking the experimental paths.

To obtain the experimental and desired paths, the motion of all the arm markers is first expressed in a sternum-marker-fixed frame to eliminate any artifact due to small torso movements. Next, the portion of the data with no arm motion, which occurs at the beginning, at the end, and briefly at full arm extension, is eliminated. A best-fit plane, termed the *motion plane*, is obtained such that the distance between the position data of all the markers and the plane is minimized in a least-squares sense. Each test has a unique motion plane onto which the data is projected. The arm motion for each test is considered to lie entirely in its corresponding motion plane. The projected wrist (marker v in Figure 1a) data is the experimental path, and the fourth-order polynomial fit to the experimental path serves as the desired path.

The parameters of the equivalent mechanism are obtained as follows. To obtain the forearm length, the motion of the wrist (marker v in Figure1a) relative to the upper arm (marker iii in Figure1a) is plotted for each test. This relative wrist path is assumed to be a circular arc of radius equal to the distance from the elbow to marker v, and a circle is fit in the least-squared sense [6] to each of the 27 relat-

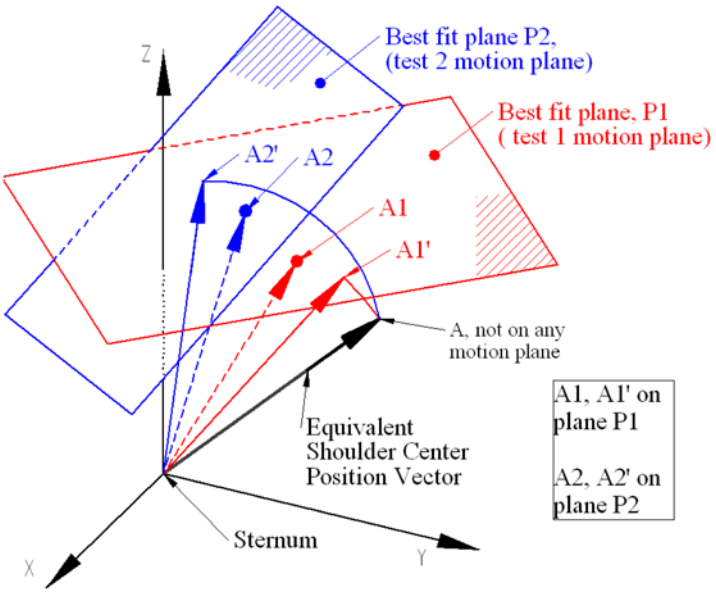


Fig. 2 Locating the shoulder center for each test. The equivalent shoulder center A is obtained by averaging the magnitudes and directions of the best-fit circle centers (A1 and A2). Point A does not lie in any motion plane. A1′ is the final shoulder center location for test 1 obtained by rotating the position vector of point A until it lies in the motion plane P1. A2′ lies in motion plane P2.

ive wrist paths. The arithmetic mean of the least-squares-fit radii gives the forearm length of the equivalent mechanism. The center of each best-fit circle gives the elbow position in a coordinate frame fixed at marker iii for the particular test and will henceforth be called the elbow. The motion of the elbow, rather than marker iii, with respect to the sternum frame is used to obtain the upper arm length and the shoulder joint position. A circle is fit in a least-squares sense to the elbow motion expressed in the sternum frame for each test. The arc length of the elbow motion varies significantly for pointing to different targets. The elbow-path length for reaching in the contralateral hemifield will be larger than that for reaching to a target in the ipsilateral hemifield. Circles fit to longer elbow paths will provide more accurate estimates of the upper arm length. Therefore, the upper arm length for the equivalent mechanism is a weighted average of the radii of the fit circles, with the weights based on the arc lengths of the corresponding elbow paths. Longer paths have higher weights. Note that the relative wrist paths are independent of the target location since each motion involves similar flexion/extension of the elbow joint. Thus, the arithmetic mean for the forearm length described above is justified.

The human shoulder does not remain stationary relative to the trunk during arm motion. Consequently, the elbow motion relative to the trunk will not be circular, and may be considered as a resultant of an upper-arm rotation about some center and some motion of that center relative to the trunk. Modeling the shoulder joint as a

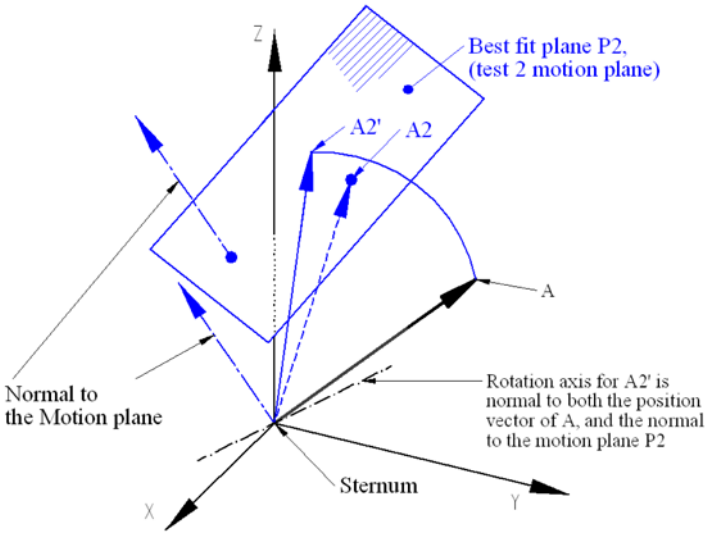


Fig. 3 Rotation axis for locating the unique shoulder center for each test. The axis passes through the sternum and is perpendicular to the position vector of the equivalent shoulder center and the normal to the corresponding motion plane.

single-DOF, fixed-axis revolute is a rough approximation necessitated by the nature of this analysis. The center of the best-fit circle to an elbow motion provides a poor estimate of the center location of such a 'shoulder-approximating' revolute joint. The following procedure, also illustrated in Figure 2, provides a better estimate. To locate a fixed shoulder center, it is required to fix the direction and magnitude of its position vector relative to the sternum-fixed frame. The circle fit to the elbow motion for each test gives a center in the motion plane for the test, e.g. points A1 and A2 in Figure 2. These are 3-dimensional position vectors expressed in the sternum-fixed frame XYZ in Figure 2. The 27 centers are combined into a single equivalent shoulder center by taking the weighted average of the directions and magnitudes of these vectors separately, and then combining the results. The same weights from the upper arm length computation are used. Point A in Figure 2 represents the equivalent shoulder center. Because of the averaging operation, the equivalent shoulder center does not lie in any of the 27 motion planes, in general. Therefore, for each test, the equivalent shoulder center vector is rotated until a unique shoulder center position in the test's motion plane is obtained, e.g. points A1′ and A2′ in Figure 2. This approach retains the length of the equivalent shoulder center position vector, and compromises only the small change in its direction. The rotation of the equivalent center's position vector is illustrated in Figure 3. The position vector of point A rotates about an axis passing through the sternum. To get the shoulder center for test 2, i.e. point A2′, the axis of rotation is perpendicular to the position vector of A and the normal to the motion plane P2. Therefore, the rotation axis for each test will pass through the sternum, but will have a different orientation. Once the

Table 1 Inputs to the path-tracking algorithm.

Parameter	Value
Angle displacement	2°
Reduced angle displacement	1°
Permissible position error	0.5 cm
Permissible tangential error	25°
Limit for curvature and rate of change of curvature	1.5×10^{-3} cm^{-1}, cm^{-2}
Speed ratio limit	15

equivalent mechanism is defined, the shoulder and elbow joint displacements, λ and μ respectively, for tracking the corresponding experimental paths are obtained via inverse kinematics. The forearm and upper arm lengths and the shoulder center determined above are used for this calculation. The experimental first-order speed ratios are obtained from the joint trajectories as

$$n = \frac{d\mu}{d\lambda} = \frac{\dot{\mu}}{\dot{\lambda}}, \tag{1}$$

where $\dot{\lambda}$ and $\dot{\mu}$ are the joint velocities of the shoulder and elbow obtained by numerically differentiating the joint displacements.

3.2 Model-Determined Speed Ratios

This section outlines the functioning of the path-tracking algorithm and the computation of the speed ratios. The fourth-order polynomial for the desired path, along with the arm-segment lengths and the shoulder center determined in the last section are inputs to the algorithm. Other inputs to the algorithm are the parameters listed in Table 1, which were constant for all tests and subjects. These values are obtained by trial and error.

To track the desired path, the tracking system starts from an initial pose corresponding to the initial position of the subject's arm, with the wrist on the desired path. The curvature properties, viz. the tangent direction and the curvature of the desired path are computed. Curvature theory [8] provides a way of mapping these values onto the shoulder-based first- and second-order speed ratios. The first-order speed ratio is defined by Equation (1), and the second-order speed ratio is $n' = dn/d\lambda$. Next, the leading joint-angle displacement, λ, is chosen. If the curvature properties exceed the limits in Table 1, the reduced angle displacement is chosen. Otherwise, the standard angle displacement is employed. The subordinate (elbow) joint displacement μ is obtained via the second-order Taylor series

$$\mu = n\lambda + \frac{n'}{2}\lambda^2. \tag{2}$$

Second-order tracking is implemented because first-order tracking produces significant wrist oscillations about the desired path. If n' is greater than the ratio limit in Table 1, the elbow is chosen as the leading joint instead, and the inverse of Equation (2) is used. The speed ratio magnitude increases when the leading joint approaches a dwell, so the ratio limit detects this phenomenon. This joint switching strategy is critical for producing good tracking. The strategy agrees with recent studies supporting the leading joint hypothesis (LJH) [5]. The LJH states that in multiarticular limb motion there exists a leading joint, usually the proximal (shoulder) joint, that creates the foundation of the required motion by moving independent of the other joints. Based on this motion, the subordinate joint(s) move such that the required end-effector motion is obtained. The advantage of the LJH is that it allows the decomposition of the control problem into hierarchical components, which simplifies the control process [5]. For this reason, the algorithm chooses the shoulder joint as the leading joint, except when it switches based on the speed ratio.

As the mechanism advances through the computed joint displacements, the motion produces error in the tracking, such that the wrist no longer lies on the desired path. The angle displacement in Table 1 refers to the leading joint displacement after which the tracking system computes the tracking error by sensing the location of the *error point*, and the tangent direction of the desired path at the error point. The error point is the point on the desired path closest to the current wrist position. Two error parameters, viz. *position error* and *tangential error*, are computed. Position error is the distance between the error point and the current wrist position, while tangential error is the angle between the wrist path tangent and the desired path tangent at the error point. If the current position or tangential error exceeds the permissible limits specified in Table 1, path correction is accomplished by changing the wrist trajectory tangent direction such that further motion reduces the position error. New speed ratios corresponding to the corrected tangent are obtained. The process repeats until the desired path ends. The strategy of computing the speed ratios only when some error parameter is exceeded serves to reduce the computational load on the system. The modeling speed ratios are thus generated by the algorithm based on the geometric properties of the desired path and the generated wrist path.

4 Experiment and Modeling Comparison

The experimental paths for most tests were curves with low curvature. The algorithm, using second-order joint coordination, was able to track the paths to a high accuracy of 0.5 cm position error, for path lengths between 10 to 20 cm. Each motion was split into an outward motion toward the target and an inward motion back to the home position. Figures 4 and 5 show typical results. The experimental and modeling speed ratios for each were compared separately. Figure 4 shows the speed ratios plotted against the normalized arc length for all speeds for an inward motion from a light in the ipsilateral hemifield. Figure 5 shows the ratios for all speeds for an outward motion to a light in the contralateral hemifield. Figures 4a

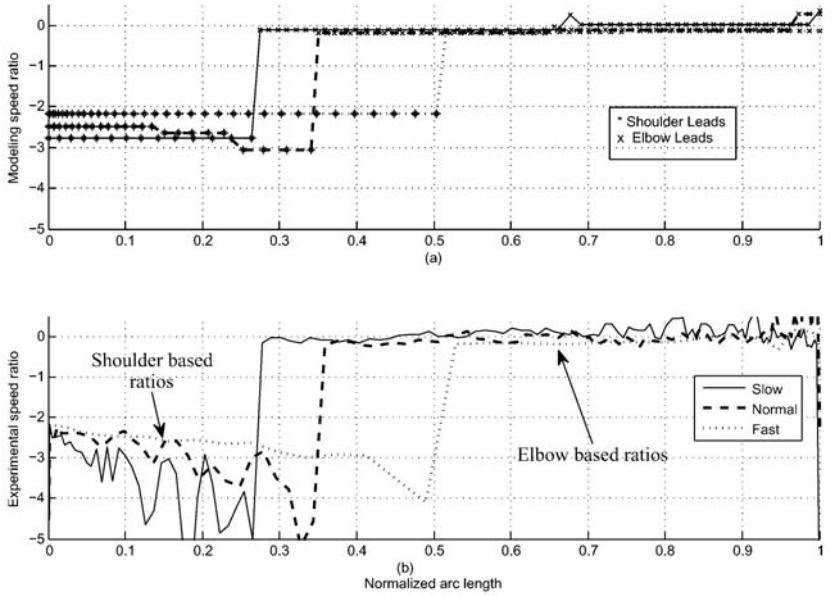


Fig. 4 Ipsilateral reaching: (a) the modeling first-order speed ratios for one subject pointing inward from light seven in Figure 1, with the right hand; (b) the experimental ratios for the same motion.

and 5a show the modeling ratios determined by the algorithm for tracking the desired paths. In Figure 4a, a portion of the path is traversed with the shoulder leading, and the remainder with the elbow leading. The shoulder-based first-order speed ratio is defined by Equation (1), and the elbow-based ratio is its inverse. As indicated in Figure 4b, the shoulder- or elbow-based ratio is plotted for the corresponding portion of the experimental path. In Figure 5a, the entire path was traversed with the shoulder leading, hence Figure 5b plots all shoulder-based ratios.

The sharp jumps in the speed ratios in Figure 4 correspond to the elbow taking over as the leading joint for ipsilateral motions, a behavior observed for both hands in all subjects. This result agrees with the findings of Levin et al. [7] that the elbow exerts the maximum interaction torque on the shoulder for motions that involve small shoulder joint displacement and large elbow joint displacement, indicating an elbow-led movement. Dounskaia et al. [4] made the same observation in their experiment dealing with drawing movements. Dounskaia [5] proposed that the elbow becomes the leading joint for motions where the mechanical advantage of the shoulder cannot be used. The present work suggests a more fundamental explanation. A leading joint cannot achieve good tracking if it is approaching a dwell because any error in the leading joint motion will be amplified at the wrist. Further, if the leading joint is stationary, the motion of the subordinate joint cannot be derived from it. For ipsilateral reaching, the shoulder joint motion is small. Hence, the elbow takes over as the leading joint for ipsilateral reaching motions. A rapid in-

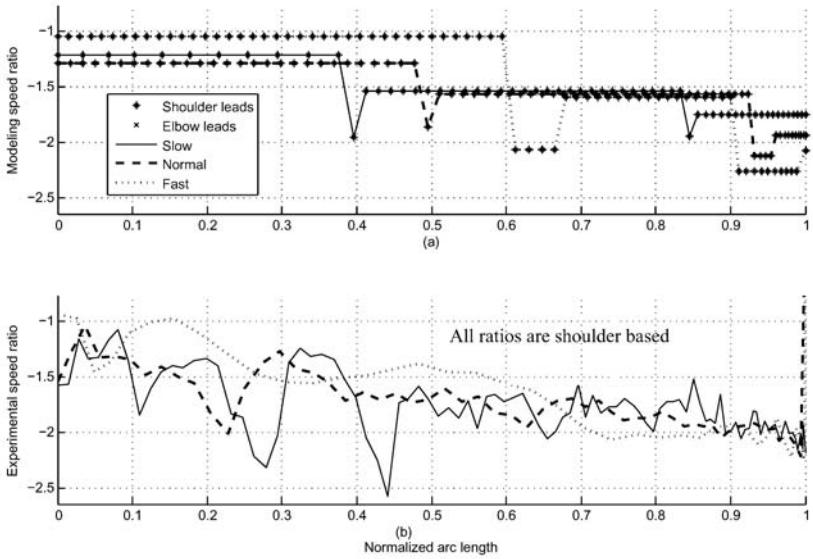


Fig. 5 Contralateral reaching: (a) the modeling first-order speed ratios for one subject pointing outward to light eight in Figure 1, with the left hand; (b) the experimental ratios for the same motion.

crease in the speed ratios indicates a joint's proximity to a dwell, and the algorithm successfully predicts this behavior via the speed ratio limit.

In contrast, the reaching movements in the contralateral hemifield do not exhibit change in the leading joint. Further, the experimental ratios in Figure 5b show gentle trends as compared to the sharp jumps in the ipsilateral hemifield. The modeling ratios follow similar trends, but in steps resulting from the algorithm's computationally frugal strategy of recalculating the ratios only when the error is exceeded. Each jump in the ratios indicates a correction and a subsequent recalculation of the speed ratios. The jumps in the speed ratios in Figure 5a are comparable to the amplitude of noise in the experimental data. Hence, the experimental data does not serve to validate or invalidate the idea of intermittent path correction. Additionally, no noticeable pattern emerged from analyzing the inward and outward motions separately.

5 Conclusions

This paper presents experimental validation of a planar, time-invariant inverse kinematic model for human motor control of reaching, and a path-tracking algorithm based on the time-invariant model. Experimental data was collected by measuring the arm motion of healthy individuals as they pointed to fixed targets in the horizontal plane at three self-selected speeds. An equivalent planar revolute-revolute

mechanism approximating the subject's arm was obtained through data analysis. The algorithm used the equivalent mechanism to successfully track the experimental paths closely. Second-order tracking is required to accurately match the experimental path, suggesting that motor planning may involve second-order joint coordination. The experimental first-order speed ratios for pointing to a target at three different speeds are similar, indicating time-invariant planning in the central nervous system. The algorithm correctly predicts the leading joint for motions in different hemifields. The experimental data and the model together corroborate the leading joint hypothesis by predicting elbow-led motions in the ipsilateral hemifield. Since the subordinate joint motion is derived from the motion of the leading joint, poor tracking results are obtained if a leading joint is near a dwell. This is because any error at the leading joint will be amplified at the wrist. Further, at a dwell, the subordinate joint motion cannot be derived from the stationary leading joint. These are more fundamental reasons for observed elbow-led motions in the ipsilateral hemifield than previously proposed.

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. IIS-0546456 to James Schmiedeler.

References

1. Ambike, S., Schmiedeler, J.P., Modeling time invariance in human arm coordination, in: J. Lenarčič, B. Roth (Eds.), *Advances in Robot Kinematics*, pp. 177–184. Springer, Dordrecht (2006).
2. Ambike, S., Schmiedeler, J.P., A methodology for implementing the curvature theory approach to path tracking with planar robots, *Mechanisms and Machine Theory*, in press (2007).
3. Crear, D.E., A study on time invariance in horizontal planar reaching movements, Masters Thesis, The Ohio State University (2007).
4. Dounskaia, N., Ketcham, C.J., Stelmach, G.E., Commonalities and differences in control of a large set of drawing movements, *Experimental Brain Research* **146**, 11–25 (2002).
5. Dounskaia, N., The internal model and the leading joint hypothesis: Implications for control of multi-joint movements, *Experimental Brain Research* **166**, 1–16 (2005).
6. Gander, W., Golub, G.H., Strebel, R., Fitting of circles and ellipses, *BIT Numerical Mathematics* **34**, 558–578 (1994).
7. Levin, O., Ouamer, M., Steyvers, M., Swinnen, S.P., Directional tuning effects during cyclic two-joint arm movements in the horizontal plane, *Experimental Brain Research* **141**, 471–484 (2001).
8. Lorenc, S.J., Stanisic, M.M., Hall Jr, A.S., Application of instantaneous invariants to the path tracking control problem of planar two degree-of-freedom systems: A singularity-free mapping of trajectory geometry, *Mechanism and Machine Theory* **30**, 883–896 (1995).

Mobility and Higher Order Local Analysis of the Configuration Space of Single-Loop Mechanisms

A. Müller[1] and J.M. Rico[2]

[1]*Institute of Mechatronics at the Chemnitz University of Technology, Reichenhainer Strasse 88, 09126 Chemnitz, Germany, e-mail: andreas.muller@ieee.org*
[2]*Departamento de Ingeniería Mecánica, FIMEE, Universidad de Guanajuato, Carr. Salamanca-Valle de Santiago 3.8+1.5 km, Salamanca, Gto. México, e-mail: jrico@salamanca.ugto.mx*

Abstract. This contribution attempts a clear definition of the mobility and phenomena such as overconstrained, underconstrained and kinematotropical mechanisms in view of the configuration space. The mobility is determined via a higher order approximation of the configuration space. The central object is the tangent cone that is determined by higher derivatives of the closure map. Explicit expression for the derivatives up to 4^{th} degree are given. We further propose the concept of a kinematic tangent cone, and put its use for mobility analysis up for discussion.

Key words: mobility, configuration space, singularities, higher order analysis.

1 Introduction

Any mobility criterion whatsoever attempts to estimate the dimension of the configuration space V – the solution set of the constraints. Higher order kinematic analysis is a means to determine the structure the of the configuration space around a given configuration, as pursued in [2, 5, 7]. Such an analysis approximates curves in V passing though a specific configuration $q \in V$. Using time as curve parameter, the analysis consists in determining admissible velocities, accelerations, jerks, and so forth. The local DOF can be deduced from the dimension of the solution sets on the different approximation levels. Clearly, the first order analysis reveals the instantaneous DOF, but the approximation order sufficient to deduce the finite DOF is yet unknown. However, the information revealed by the second, third or fourth order is usually enough to determine the linkage finite mobility. Nevertheless, higher order approximations can always be performed for a particular mechanism. The higher order analysis shall also enable to identify singular configurations, by comparing the instantaneous and finite DOF. This fails for underconstrained mechanisms, where this discrepancy is not due to a singularity of V (Section 3).

In this paper the concept of kinematic tangent cone is employed for the mobility analysis of single-loop mechanisms that rests on higher order approximations of the closure constraints. The advantageous feature is that the sufficient approximation order can be identified.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 215–224.

2 The Kinematics of a Single Kinematic Loop

The kinematic loop is regarded as a chain of one-DOF joints. Higher-DOF joints are replaced by successive one-DOF joints. The joints of a kinematic loop are numbered in increasing order with $a = 1, \ldots, n$.

Denote with $Y_a \in \mathbb{R}^6$ the screw coordinate vector of joint a deduced from a global coordinate system. If $\widehat{Y}_a \in se(3)$ denotes the corresponding $se(3)$ matrix [6], the geometric loop constraints $f(q) = I$ must be fulfilled, where $f : \mathbb{V}^n \to SE(3)$ is the closure map

$$f(q) := e^{\widehat{Y}_1 q^1} \cdots e^{\widehat{Y}_n q^n}. \tag{1}$$

q^a are the joint variables, and $q \in \mathbb{V}^n$ is the representing point in the parameter space $\mathbb{V}^n$. The geometric constraints define the *configuration space* of the single-loop mechanism

$$V := f^{-1}(I) = \{q \in \mathbb{V}^n \mid f(q) = I\}, \tag{2}$$

where $q \in V^n$ are admissible representing points.

At any $q \in V$ holds $f(q) = I$, and the differential of the closure map is $df(q)(x) = df(q)(x) f^{-1}(q) = \widehat{H}_a(q) x^a \in se(3)$, with the instantaneous screw coordinate vectors

$$H_a(q) := Ad_{e^{\widehat{Y}_1 q^1} \cdots e^{\widehat{Y}_{a-1} q^{a-1}}} Y_a \tag{3}$$

(for brevity, the summation convention $H_{ab\ldots} x^b \equiv \sum_b H_{ab\ldots} x^b$ is used in the paper). Ad_C denotes the 6×6 matrix representation of the adjoint map $Ad : SE(3) \times se(3) \to se(3)$, when using screw coordinates. Moreover, $H(q)$ is the Jacobian of f at $q \in V$. Preferably, $q = 0$ is assigned to the reference configuration, so that $H_a(0) = Y_a$.

3 The Configuration Spaces of a Mechanism

The motion of a mechanism corresponds to a curve in V, and its local structure reveals admissible motions. Since f is an analytic map between manifolds, V is a real analytic variety.

Denote with $d(q)$ the rank of the constraint Jacobian $H(q)$, and with $d_{\max} := \max_{q \in \mathbb{V}^n} d(q)$. $U(q)$ denotes an open neighborhood of q in $\mathbb{V}^n$.

The *differential* (or instantaneous) *DOF* of a mechanism is $\delta_{\text{diff}}(q) := n - d(q) = \dim \ker H(q)$. A point $q \in V$ is *regular* iff there is a neighborhood $U(q)$ such that $\delta_{\text{diff}}(q) = \delta_{\text{diff}}(p)$ for all $p \in U(q) \cap V$, otherwise it is *singular*. For V exists a stratification in a finite number of (not necessarily connected) i-dimensional ($i = 0, 1, \ldots$) manifolds $\mathcal{M}_i$ of regular and manifolds $\mathcal{S}_i$ of singular points [10]. The *local DOF* $\delta_{\text{loc}}(q) := \dim_q V$ is the local dimension of V. This is the highest dimension of manifolds (later called modes) passing through q. If q is regular, then V is locally an analytic $\delta_{\text{loc}}(q)$-dimensional manifold. The *global DOF* δ is the highest local dimension of V. The connected components of V are

called the *assemblies*, and the connected submanifolds of V (only comprising regular points) are the *motion modes* of the mechanism. If a connected component of V comprises motion modes of different dimensions, the mechanism is said to be *kinematotropic* [11]. Denote with $\deg(q) := \delta_{\text{diff}}(q) - \delta_{\text{loc}}(q)$ the *degree* of q that quantifies the discrepancy of the differential and local DOF. It is worth to emphasize that the differential DOF characterizes the instantaneous mobility of the mechanism, whereas the local DOF determines the finite mobility, and is a topological property of the configuration space. The differential DOF may even be different from the local DOF in a mode of the mechanism. If there is a regular configuration $q \in V$ for which $\deg(q) > 0$, the mechanism is called *underconstrained*. A mechanism is called *overconstrained* if in a motion mode $d(q) < d_{\max}$.

4 Higher Order Local Analysis of a Single Loop

There are two geometric objects that reveal the local structure of the configuration space: the tangent (vector) space and the tangent cone. The concept of tangent cone was introduced by Whitney [9]. Lerbet was the first who applied this concept to the analysis of mechanisms in [2]. The tangent cone to V is the best local approximation of V in regular and singular configurations. Consequently, its dimension equals the local mechanism DOF.

The tangent space to V at $q \in V$ is generated by the ideal $\mathbf{I}(V, q)$ of analytic germs over q, that vanish on V [9]:

$$T_q V = \{x \mid df(x) = 0, f \in \mathbf{I}(V, q)\}, \tag{4}$$

where $df(q)$ is the differential of f at q. In regular points, $T_q V$ consists of tangent vectors to V, and $\dim T_q V = \dim_q V$, while in singular points $\dim T_q V > \dim_q V$. It is, however, not always true that in regular points $\delta_{\text{diff}}(q) = \dim T_q V$. Generally, the tangent space consists of admissible instantaneous motions so that its dimension shall reveal the differential DOF. But for underconstrained mechanisms the dimension of the tangent space is only a lower bound on the differential DOF. Moreover, the differential DOF is determined by the closure map, and not by the structure of the configuration space. Thus $T_q V$ locally characterizes the topology of V, up to first order, but not necessarily the first order kinematics.

The best local approximation of V in any point is the tangent cone. The tangent cone to V in q is determined as [10]

$$C_q V = \left\{x \in \mathbb{R}^n \mid d^{(*)} f(q)(x), f \in \mathbf{I}(V, q)\right\} \subseteq T_q V. \tag{5}$$

$d^{(*)} f(q_0)(q)$ is the leading homogeneous polynomial in the Taylor expansion of f at q_0

$$f(q_0 + q) = f(q_0) + \frac{1}{1!} d^{(1)} f(q_0)(q) + \cdots + \frac{1}{m!} d^{(m)} f(q_0)(q) + \cdots \tag{6}$$

i.e. $d^{(*)} f\,(q) := d^{(m)} f\,(q)$ so that $d^{(k)} f\,(q_0) \equiv 0, k < m$ and $d^{(m)} f\,(q_0) \not\equiv 0$.

$C_q V$ is a faithful approximation of V in any point q, and indeed $\delta_{\mathrm{loc}}(q) = \dim C_q V = \dim_q V$. This is the set of tangent vectors on curves in V passing through q. Generally $C_q V$ is a cone in $\mathbb{R}^n$, which degenerates to a vector space in regular points. In fact, q is a regular point of V if and only if $C_q V = T_q V$.

The obvious difficulty in applying the above definition is the lack of a constructive definition of the ideal $\mathbf{I}(V, q)$, and neither (4) nor (5) allow for an immediate evaluation. The tangent cone can further be determined via higher order analysis [2, 7]. In both contributions explicit expressions for higher derivatives of the closure map f are given as prerequisite for higher order analyses. Equivalent expressions were reported in [5] that are stated below, in Equations (10) and (11). The sufficient order is found in the course of the analysis of a particular mechanism.

In this paper we propose to approach the problem by further investigating the closure map. The central object is the set of closure maps that can be obtained by changes of the reference frame in which the closure map is expressed

$$S_f := \left\{ M f M^{-1} \mid M \in SE(3) \right\}. \tag{7}$$

S_f is the similarity class of f with respect to frame transformations.

Following (4), it is now possible to introduce a 'tangent space' using the similarity class of f. For $h \in S_f$ holds $h(q) = I$ in $q \in V$, so that $dh(q)(x) = dh(q)(x)\, h^{-1}(q) \in se(3)$. With a slight abuse of notation, denote with $dh^i(q)(x)$ the components of the differential in vector representation. The vector space

$$T_q V\left(S_f\right) := \left\{ x \mid dh^i(q)(x) = 0, h \in S_f \right\} \equiv \ker H(q) \tag{8}$$

should be called the *kinematic tangent space* to V at q as it represents the first order kinematic properties of the mechanism. This is the definition of the tangent space of V generally found in the literature [3], although $T_q V$ (representing the structure of V) is generally a subspace of $T_q V\left(S_f\right)$. The set

$$C_q V\left(S_f\right) := \left\{ x \mid d^{(*)} h^i(q)(x) = 0, h \in S_f \right\} \subseteq T_q V\left(S_f\right) \tag{9}$$

should be called the *kinematic tangent cone* at $q \in V$.

Denote with $H^{(i)}(q)(x) := d^{i-1} H^{(1)}(q)(x)$, where $H^{(2)}(q)(x) = H^{(2)}_{ab}(q)\, x^a x^b$, with $H^{(2)}_{ab} := \partial_{q^b} H^{(1)}_a$, and so forth for higher orders, where $H^{(1)}_a := H_a$. $H^{(i)}(q)(x)$ comprises homogenous polynomials in $x \in \mathbb{R}^n$ of degree i. From (3) follows

$$\widehat{H}^{(2)}_{ab} = [\widehat{H}^{(1)}_b, \widehat{H}^{(1)}_a], \quad b < a. \tag{10}$$

Higher order derivatives follow by repeated application of $\partial_{q^c}[\widehat{H}^{(1)}_b, \widehat{H}^{(1)}_a] \equiv [\partial_{q^c} \widehat{H}^{(1)}_b, \widehat{H}^{(1)}_a] - [\partial_{q^c} \widehat{H}^{(1)}_a, \widehat{H}^{(1)}_b]$ and the Jacobi identity. In particular

$$\widehat{H}^{(3)}_{abc} = \begin{cases} [[\widehat{H}^{(1)}_a, \widehat{H}^{(1)}_b], \widehat{H}^{(1)}_c], & c < b < a \\ [[\widehat{H}^{(1)}_a, \widehat{H}^{(1)}_c], \widehat{H}^{(1)}_b], & b \le c < a \end{cases} \tag{11}$$

$$\widehat{H}^{(4)}_{abcd} = \begin{cases} [\widehat{H}^{(1)}_c, [\widehat{H}^{(1)}_b, [\widehat{H}^{(1)}_d, \widehat{H}^{(1)}_a]]] \,, \; c < b \leq d < a \\ [\widehat{H}^{(1)}_b, [\widehat{H}^{(1)}_c, [\widehat{H}^{(1)}_d, \widehat{H}^{(1)}_a]]] \,, \; b \leq c \leq d < a \\ [\widehat{H}^{(1)}_d, [\widehat{H}^{(1)}_c, [\widehat{H}^{(1)}_b, \widehat{H}^{(1)}_a]]] \,, \; d < c < b < a \\ [\widehat{H}^{(1)}_c, [\widehat{H}^{(1)}_d, [\widehat{H}^{(1)}_b, \widehat{H}^{(1)}_a]]] \,, \; c \leq d < b < a \\ [\widehat{H}^{(1)}_d, [\widehat{H}^{(1)}_b, [\widehat{H}^{(1)}_c, \widehat{H}^{(1)}_a]]] \,, \; d < b \leq c \leq a \\ [\widehat{H}^{(1)}_b, [\widehat{H}^{(1)}_d, [\widehat{H}^{(1)}_c, \widehat{H}^{(1)}_a]]] \,, \; b \leq d < c < a \end{cases}$$

In $q \in V$, the vector space of loop velocities is

$$D_q := \text{span}\,(\widehat{H}^{(1)}_1(q), \ldots, \widehat{H}^{(1)}_n(q)) \subseteq se\,(3)\,. \tag{12}$$

Denote with $\overline{D}$ the union of the vector spaces D_q at all (not necessarily admissible) configurations $q \in \mathbb{V}^n$. It is well-known in screw theory that $\overline{D}$ can be constructed via the sequence of vector spaces

$$D^{(i+1)}_q := D^{(i)}_q + \{[X, Y] \mid X \in D^{(1)}_q, Y \in D^{(i)}_q\}, \tag{13}$$

with $D^{(1)}_q := D_q$. E.g., $D^{(2)}_q$ is the vector space spanned by $\widehat{H}^{(1)}_a(q)$ and their Lie brackets (screw products) [6]. In each step the dimension of $D^{(i)}$ increases by at least one, until $D^{(\kappa+1)}_q = D^{(\kappa)}_q$, and, because $se\,(3)$ is six-dimensional, $\kappa < 6$. The respective vector space is denoted $\overline{D} := D^{(\kappa)}_q$. $\overline{D}$ is obviously closed with respect to the Lie bracket, and is hence a $se\,(3)$ subalgebra - screw algebra generated by $Y_1, \ldots, Y_n$. Moreover, $\kappa \leq \kappa_{\max}$, with $\kappa_{\max}$ for the respective subalgebra. $\overline{D}$ is indeed independent of q, but $\kappa_{\max}$ depends on q. The subalgebra $\overline{D}$ is called the *closure algebra* [8]. The Lie group generated by the algebra $\overline{D}$ is the image space of f.

A closer look at (10) reveals that $\widehat{H}^{(2)}_{ab}$ contains all linearly independent Lie brackets of the vector fields $\widehat{H}^{(1)}_{ab}$. Also $\widehat{H}^{(3)}_{abc}$ contains all brackets of the $\widehat{H}^{(2)}_{ab}$, and so forth for $\widehat{H}^{(i)}_{ab\ldots}$. Consequently

$$D^{(i)}_q := \text{span}\big(\widehat{H}^{(1)}_a(q)\,, \widehat{H}^{(2)}_{ab}(q), \ldots, \widehat{H}^{(i)}_{ab\ldots}(q)\big). \tag{14}$$

In the definition (9) of the kinematic tangent cone, the nonvanishing components of $d^{(i)} f\,(q)$, and thus of $H^{(i)}(q)$, need to be identified. These components depend on the reference frame, which is why the equivalence class S_f appears in (9). Let M be the transformation of $D^{(1)}_q$ in a minimal basis on the closure algebra. Denote

$$\begin{aligned} B^{(1)}_a(q) &:= Ad_M H^{(1)}_a(q) \\ B^{(i)}_{ab\ldots}(q) &:= Ad_M H^{(i)}_{ab\ldots}(q)\,, \text{ with } \widehat{H}^{(i)}_{ab\ldots}(q) \notin D^{(i-1)}_q, i > 1, \end{aligned} \tag{15}$$

so that $B^{(\kappa+1)}_{ab\ldots}(q) \equiv 0$. The maps $B^{(i)}(q)(x) := B^{(i)}_{ab\ldots}(q)\, x^a x^b \ldots$ are homogenous in x of degree i.

Denote with P_i the operator that removes all those components from $\widehat{X} \in D_q^{(i)}$ that appear in the expansion of $D_q^{(i-1)}$, where P_1 is the identity operator. In vector notation, P_i is a projector matrix. It is now possible to introduce cones of degree i as

$$K_q^i V := \{x \mid P_1 B^{(1)}(q)(x) = \ldots = P_i B^{(i)}(q)(x) = 0\}. \tag{16}$$

$K_q^i V$ is defined by the leading homogeneous polynomials in the expansion of the components of the closure map up to order i.

Obviously $K_q^1 V \equiv T_q V(S_f)$, and $\delta_{\text{diff}}(q) \equiv T_q V(S_f) = n - \dim D_q^{(1)}$. The recursive form $K_q^i V = \{x \in K_q^{i-1} V \mid P_i B^{(i)}(q)(x) = 0\}$ shows that the cones are incremental refinements.

The kinematic tangent cone can be constructed when the constraint map, and thus $\overline{D}$ is expressed in a minimal basis [4].

Theorem 1. *In $q \in V$ let M in (15) be the transformation of $D_q^{(1)}$ in a minimal basis. Then*

$$C_q V(S_f) = K_q^k V, \tag{17}$$

where k is the smallest such that $P_1 \cdot \ldots \cdot P_k \cdot P_{k+1} = 0$. It holds $k \leq \kappa$, with κ such that $D_q^{(\kappa)} = D_q^{(\kappa+1)}$.

Remark 1. It can be shown that for *se* (3) always $\kappa \leq 4$. Consequently $C_q V(S_f)$ is a cone of at most fourth degree. However, for all mechanisms that were investigated by the authors, $C_q V(S_f)$ is of first or second degree.

5 Examples

The following two examples show how the presented methodology can be used to determine the local DOF in singular configurations of overconstrained and underconstrained mechanisms. The details of the analyses are omitted.

5.1 Spatial 6R Mechanism with Partitioned Mobility

The spatial 6R mechanism in Figure 1a is analyzed in the shown configuration q_0. The closure algebra is determined by sequence of vector spaces $D_{q_0}^{(1)} \subset \cdots \subset D_{q_0}^{(4)} = \overline{D} = se(3)$, with $\dim D_{q_0}^{(1)} = 2$. According to (17) the kinematic tangent cone is $C_{q_0} V\left(S_f\right) = K_{q_0}^4 V\left(S_f\right)$. The cones up to fourth order are

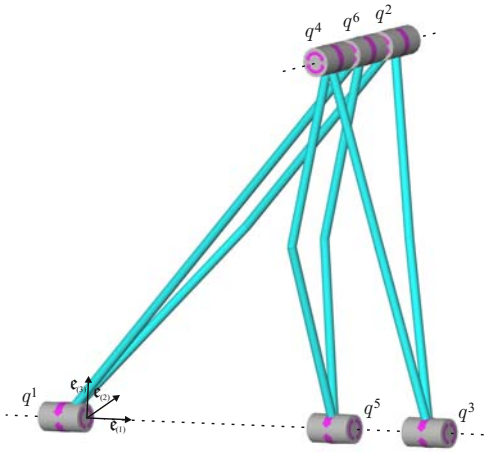


Fig. 1 Overconstrained 6R mechanism in a singular configuration.

$$
\begin{aligned}
K^1_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^6 \mid x = (-s-u, -t-v, s, t, u, v)\right\},\\
K^2_{q_0} V\left(S_f\right) &= K^{2,1}_{q_0} V\left(S_f\right) \cup K^{2,2}_{q_0} V\left(S_f\right) \cup K^{2,3}_{q_0} V\left(S_f\right),\\
\text{with} \quad K^{2,1}_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^6 \mid x = \left(-\tfrac{t\,u}{t+v}, -t-v, -\tfrac{u\,v}{t+v}, t, u, v\right)\right\},\\
K^{2,2}_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^6 \mid x = (-s, 0, s, -v, 0, v)\right\},\\
K^{2,3}_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^6 \mid x = (-s-u, 0, s, 0, u, 0)\right\},\\
K^3_{q_0} V\left(S_f\right) &= K^{3,1}_{q_0} V\left(S_f\right) \cup \ldots \cup K^{3,5}_{q_0} V\left(S_f\right),\\
\text{with} \quad K^{3,1}_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^6 \mid x = (0, -v, -u, 0, u, v)\right\},\\
K^{3,2}_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^6 \mid x = (0, -t-v, 0, t, 0, v)\right\},\\
K^{3,3}_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^6 \mid x = (-u, -t, 0, t, u, 0)\right\},\\
K^{3,4}_{q_0} V\left(S_f\right) &= K^{2,2}_{q_0} V\left(S_f\right),\ K^{3,5}_{q_0} V\left(S_f\right) = K^{2,3}_{q_0} V\left(S_f\right)\\
K^4_{q_0} V\left(S_f\right) &= K^3_{q_0} V\left(S_f\right).
\end{aligned}
$$

The kinematic tangent space $T_{q_0} V\left(S_f\right) \equiv K^1_{q_0} V\left(S_f\right)$ is a four-dimensional vector space in $\mathbb{R}^6$. The higher order analysis [7] shows that the tangent cone $C_q V$ equals the kinematic tangent cone. The kinematic tangent cone at q_0 is the union of five two-dimensional vector spaces. The latter are tangent spaces $T_{q_0} V^i = K^{3,i}_{q_0} V\left(S_f\right)$, $i = 1, \ldots, 5$ to the irreducible components V^i of the configuration space intersecting in q_0:

$$
\begin{aligned}
V^1 &= \left\{q = (0, -q^6, -q^5, 0, q^5, q^6)\right\}, \quad V^2 = \left\{q = (0, -q^4 - q^6, 0, q^4, 0, q^6)\right\},\\
V^3 &= \left\{q = (-q^5, -q^4, 0, q^4, q^5, 0)\right\}, \quad V^4 = \left\{q = (-q^3, 0, q^3, -q^6, 0, q^6)\right\},\\
V^5 &= \left\{q = (-q^3 - q^5, 0, q^3, 0, q^5, 0)\right\}.
\end{aligned}
$$

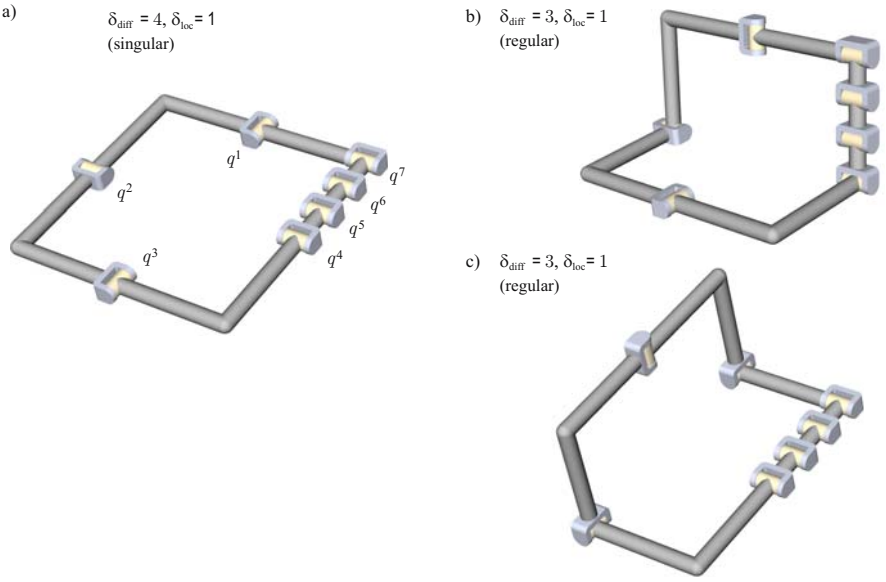


Fig. 2 Overconstrained and underconstrained 7R mechanism in singular and regular configurations.

The local DOF is $\delta_{\text{loc}}(q_0) \equiv \dim C_{q_0} V = 2$. This mechanism is overconstrained. Since $\dim K^1_{q_0} V\left(S_f\right) > \dim K^2_{q_0} V\left(S_f\right) > \dim C_{q_0} V\left(S_f\right)$, there are four-dimensional singular motions of first order and three-dimensional singular motions of second order.

5.2 Underconstrained 7R Mechanism

Figure 2a shows a 7R mechanism in the reference configuration q_0. The closure algebra is determined by the sequence of vector spaces $D^{(1)}_{q_0} \subset D^{(2)}_{q_0} = \overline{D} = se(3)$, with $\dim D^{(1)}_{q_0} = 3$. The kinematic tangent cone is $C_{q_0} V\left(S_f\right) = K^2_{q_0} V\left(S_f\right)$, where

$$
\begin{aligned}
K^1_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^7 \mid x = (-s, -t + u + 2v, s, t, -2u - 3v, u, v)\right\}, \\
K^2_{q_0} V\left(S_f\right) &= K^{2,1}_{q_0} V\left(S_f\right) \cup K^{2,2}_{q_0} V\left(S_f\right), \\
\text{with} \quad K^{2,1}_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^7 \mid x = (0, -t, 0, t, 0, 0, 0)\right\}, \\
K^{2,2}_{q_0} V\left(S_f\right) &= \left\{x \in \mathbb{R}^7 \mid x = (-s, 0, s, 0, 0, 0, 0)\right\}.
\end{aligned}
$$

The kinematic tangent space $T_{q_0} V\left(S_f\right) \equiv K^1_{q_0} V\left(S_f\right)$ is a four-dimensional vector space in $\mathbb{R}^7$, i.e. $\delta_{\text{diff}}(q_0) = 4$. Again the tangent cone $C_q V$ can be shown to be identical to the kinematic tangent cone. The latter is the union of the one-

dimensional tangent spaces to the two one-dimensional motion modes V^i intersecting in q_0:

$$V^1 = \left\{q = (0, -x^4, 0, x^4, 0, 0, 0)\right\}, \quad V^2 = \left\{q = (-x^3, 0, x^3, 0, 0, 0, 0)\right\}.$$

In q_0 the mechanism possesses the local DOF $\delta_{\text{loc}}(q_0) = 1$. Observing $\delta_{\text{diff}}(q_0) < \delta_{\text{loc}}(q_0)$ suggests that q_0 is a singular configuration. In fact, q_0 is a bifurcation point of V. From this configuration that the mechanism can enter the motion modes in Figures 2b and 2c. Analyzing the configuration q_1 in Figure 1b leads to $D^{(1)}_{q_1} \subset D^{(2)}_{q_1} = \overline{D} = se(3)$, with $\dim D^{(1)}_{q_1} = 4$, and

$$K^1_{q_1} V\left(S_f\right) = \left\{x \in \mathbb{R}^7 \mid x = (0, -t + u + 2v, 0, t, -2u - 3v, u, v)\right\},$$
$$K^2_{q_1} V\left(S_f\right) = \left\{x \in \mathbb{R}^7 \mid x = (0, -t, 0, t, 0, 0, 0)\right\}.$$

Hence, $\delta_{\text{diff}}(q_1) = 3$ and $\delta_{\text{loc}}(q_1) = 1$. Although $\delta_{\text{diff}}(q_1) < \delta_{\text{loc}}(q_1)$ the configuration q_1 is not singular since δ_{diff} is constant in a neighborhood of q_1 in V. Rather, the mechanism is underconstrained in this motion mode, with degree $\deg q_1 = 2$. Moreover, the mechanism is overconstrained. Both modes, (b) and (c) comprise regular points of degree 2. The kinematic tangent spaces at any point in modes (b and (c) are three-dimensional vector spaces, which cannot only consist of tangent vectors to the one-dimensional manifold. It moreover comprises tangent vectors and necessarily infinitesimal displacements. The latter displacements correspond to infinitesimal motions of joints 4, 5, 6 and 7.

It is obvious that it cannot be concluded whether or not a configuration is singular by just comparing the differential and local DOF. Such a classification must be based on the (proper) tangent space $T_q V$.

6 Summary

The higher order analysis of the configuration of single-loop mechanisms was addressed in this paper aiming at the mobility determination. The concept of tangent space and cone has been recalled as a means to tackle the mobility problem, given that the dimension of the tangent cone equals the local mobility in a given configuration. Noting that the methods for a higher order analysis, and thus for the determination of the tangent cone, proposed in [2, 7] and elsewhere do not yield the order that is sufficient for the computation of the mobility, the concept of kinematic tangent cone was introduced. The latter is determined by the equivalence class of closure maps obtained by frame transformations.

As prerequisite for the investigation, a clear definition of the mobility was given in view of the configuration space as a variety. In particular, overconstrained and underconstrained mechanisms are distinguished. It is pointed out that singular configurations of underconstrained mechanisms can not be detected by simply compar-

ing the differential and local DOF. Moreover, this classification in fact requires the two differential geometric object, namely the tangent space and tangent cone.

References

1. Gogu, G., Mobility of mechanism: A critical review, *Mech. Mach. Theory* **40**, 1068–1097 (2005).
2. Lerbet, J., Analytic geometry and singularities of mechanisms, *ZAMM, Z. angew. Math. Mech.* **78**(10b), 687–694 (1999).
3. Maisser, P., Differential geometric methods in multibody system dynamics, *Nonlinear Analysis, Theory, Methods & App.* **30**(8), 5127–5133 (1997).
4. Müller, A., *Singuläre Phänomene in der Kinematik von Starrkörpermechanismen*, Shaker Verlag (2004).
5. Müller, A., Geometric characterization of the configuration space of rigid body mechanisms in regular and singular points, ASME International Design Engineering Technical Conferences, September 22–28, Long Beach, California, USA (2005).
6. Murray, R.M., Li, Z. and Sastry, S.S., *A Mathematical Introduction to Robotic Manipulation*, CRC Press (1993).
7. Rico, J.M., Cervantes, J.J., Díez, C.R., Gallardo, J. and Aguilera, L.D., Kinematic analyses in screw form and the determination of mobility and connectivity in single-loop kinematic chains, *Mech. Mach. Theory*, submitted (2007).
8. Rico, J., Gallardo, J. and Ravani, B., The Lie algebra and the mobility of kinematic chains, *Journal of Robotic Systems* **20**, 477–499 (2003).
9. Whitney, H., Tangents to an analytic variety, *Annals of Math.* **81**, 496–549 (1965).
10. Whitney, H., Local properties of analytic varieties, in: *Differential and Combinatorial Topology*, Princeton University Press (1965).
11. Wohlhart, K., Kinematotropic linkages, in: J. Lenarčič, V. Parent-Castelli (Eds.), *Recent Advances in Robot Kinematics*, Kluwer, pp. 359–368 (1996).

Optimization of a Test Trajectory for SCARA Systems

J.-F. Gauthier[1], J. Angeles[1] and S. Nokleby[2]

[1]*Centre for Intelligent Machines, McGill University, Montreal, Quebec, Canada H3A 2A7, e-mail: {jfgauthier, angeles}@cim.mcgill.ca*
[2]*Faculty of Engineering and Applied Science, University of Ontario Institute of Technology, Oshawa, Ontario, Canada L1H 7K4, e-mail: scott.nokleby@uoit.ca*

Abstract. The synthesis of a smooth curve to implement the test trajectory used in SCARA systems is the subject of this paper. The test trajectory includes square corners between its vertical and horizontal segments, which are sources of velocity and acceleration discontinuities. Lamé curves are used to smooth the corners to provide G^2-continuity throughout the test trajectory. Moreover, to make the overall trajectory as smooth as possible, the parameters defining this curve should be selected so as to minimize a cost function. The trajectory is thus synthesized by minimizing the root-mean-square (rms) value of the kinetic energy time-derivative, subject to inequality constraints, using the Orthogonal Decomposition Algorithm, which is based on gradient evaluations.

Key words: optimization, SCARA, Cartesian trajectory planning.

1 Introduction

One of the most typical tasks assigned to industrial robots is pick-and-place operations (PPO). It is highly desirable that the PPO be a smooth trajectory, which is accomplished by securing the continuity of the curve, its tangent and its curvature. Paul [11], Taylor [14], and Luh et al. [10], first proposed paths made up of straight line segments connected together by smooth transitions with controlled acceleration. These techniques, which define the path in Cartesian-space, imply an off-line computation of the joint variables. It is advantageous to compute a trajectory in the joint-space if the user is interested in defining the trajectory on-line, as in [1, 9]. In this work, the trajectory will be defined in Cartesian-space.

Many researchers have attacked the problem of minimizing the cost and time of moving a manipulator along a specific Cartesian path, see for instance [4, 6, 12, 13]. In general, their approaches consist of optimizing the joint motions with respect to the dynamics of a manipulator. The approach presented in this paper differs in that the focus is on the optimization of the Cartesian trajectory itself, to serve as a test-cycle for SCARA (Selective Compliance Assembly Robot Arm) systems, serial or parallel, executing PPO. In this vein, the geometric and inertial parameters of the system at hand are not taken into account. The trajectory is optimized by minimizing

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 225–234.

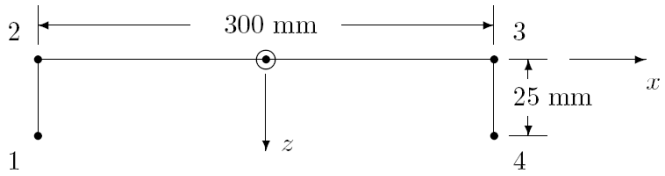


Fig. 1 SCARA systems test trajectory.

the variation of the kinetic energy of the payload alone. Moreover, the motivation of this work being the optimization of the McGill Schönflies Motion Generator (SMG) [3], the focus is on four-dof (degree-of-freedom) Cartesian trajectory optimization.

Typically, SMG, or SCARA systems, are used for PPO in industry. A trajectory for this type of operation has been adopted by the industry in order to provide a basis for comparison among the different systems on the market. This trajectory involves a vertical upward translation of 25 mm, a horizontal translation of 300 mm and a final vertical downward translation of 25 mm as shown in Figure 1. The moving platform (MP) of the SCARA system has to move through this trajectory back and forth with a rotation of 180°, along the horizontal segment, in a given cycle time. Adept Technology[1] boasts cycle times of 500 ms, while EPSON[2] claims cycle times of 409 ms.

As specified, the test trajectory includes square corners between its vertical and horizontal segments, which are obvious sources of acceleration discontinuities. These corners have to be smoothed in order to remove these discontinuities. The smoothing is typically done using circular arcs and ellipses, which are not the most suitable choices, since these types of curves do not provide G^2-continuity when connected to straight segments. G^2-continuity means position, tangent and curvature continuity over a given geometric curve. Even when suitable curves like cubic splines or clothoids [5] have been used to smooth the corners, no optimization of their parameters has ever been made to render the trajectory as smooth as possible.

The work reported here aims at the synthesis of an optimized G^2-continuous trajectory to be used in SCARA systems. The parameters defining the trajectory are to be determined such that a compromise between accelerations and maximum velocities is made, in order to render the trajectory as smooth as possible. To achieve this objective, the trajectory is optimized, under inequality constraints, by minimizing the root-mean-square (rms) value of the time-derivative of the kinetic energy per unit mass of the payload.

[1] http://www.adept.com/products/pdf/Quattro_ds.pdf

[2] http://robots.epson.com/downloads/brochurefiles/EPSON_E2S_SCARA_Robots.pdf

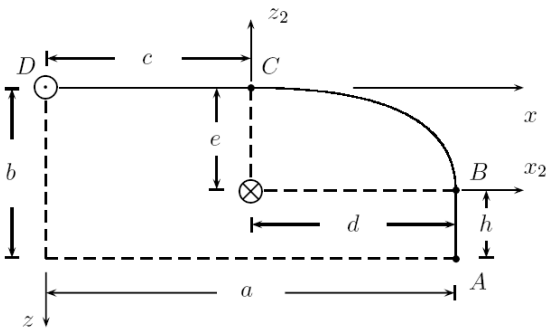


Fig. 2 Half of the proposed trajectory with its parameters.

2 The Proposed Smooth Trajectory

To simplify the derivations in the ensuing analysis, only the right half of the smoothed trajectory will be synthesized, as shown in Figure 2. Parameters a and b in Figure 2 are held constant to respect the standard test trajectory, at 150 (mm) and 25 (mm), respectively. Moreover, the time t_{AD}, required to traverse half of the trajectory, equals one quarter of the cycle time of the test trajectory. Finally, the rotation of the end-effector between points A and D is 90°. Therefore, the optimization is conducted by a suitable choice of parameters c and h. The curve parameters d and e obey a linear relation with parameters c and h, namely, $d = a - c$ and $e = b - h$. The respective ranges of values for these parameters are $0 \leq c \leq a$ and $0 \leq h \leq b$ to ensure that the solution has a physical meaning. The rotation of the optimized trajectory is started as soon as the MP starts to move in the x direction, i.e., at point B. For the synthesis of the curve BC, splines, clothoids and Lamé curves can be used. Lamé curves are chosen because they allow a parametrization of the curve in Cartesian coordinates and are analytic throughout the required path.

Lamé curves are defined by the equation

$$|u|^m + |v|^m = 1 \qquad m = 1, 2, \ldots \tag{1}$$

whence it is apparent that even-order, i.e., for m even, Lamé curves are analytic everywhere. As $m \to \infty$, Eq. (1) leads to a square shape. As the Lamé curve with $m = 2$ is a circle, which has a constant curvature, and hence, cannot be blent smoothly with a line, the next smallest curve of this family that allows for such blending is the cubic Lamé curve. Therefore, as can be seen in Figure 2, for given parameters d and e in the coordinate frame x_2-z_2, the equation of the Lamé curve of choice becomes

$$\left|\frac{x_2}{d}\right|^3 + \left|\frac{z_2}{e}\right|^3 = 1 \tag{2}$$

which apparently fails to be analytic everywhere. However, by limiting the curve to the first quadrant, the absolute value in Eq. (2) can be dispensed with, thereby rendering the cubic Lamé curve open but analytic everywhere.

For Eq. (1), the coordinates u and v can be defined explicitly in terms of θ [8] as

$$u(\theta) = 1/(1 + \tan^m \theta)^{1/m}, \quad v(\theta) = \tan\theta/(1 + \tan^m \theta)^{1/m} \tag{3}$$

where θ is measured with respect to the u-coordinate axis and is defined positive counterclockwise. To apply the above parametrization to the coordinate frame x_2-z_2, with $m = 3$, one can simply use an affine transformation scaling values of x_2 and z_2 by means of parameters d and e respectively, namely,

$$x_2(\theta) = d/(1 + \tan^3 \theta)^{1/3}, \quad z_2(\theta) = e\tan\theta/(1 + \tan^3 \theta)^{1/3} \tag{4}$$

The length l_c of the Lamé curve, defined by Eq. (2) can be expressed as

$$l_c = \int \mathrm{d}l = \int \sqrt{\mathrm{d}{x_2}^2 + \mathrm{d}{z_2}^2} \tag{5}$$

in which $\mathrm{d}x_2$ and $\mathrm{d}z_2$ are the infinitesimal versions of the increments Δx_2 and Δz_2. Eqs. (4) are used to slightly rearrange Eq. (5), thus obtaining

$$l_c = \int_0^{\pi/2} \sqrt{x_2'(\theta)^2 + z_2'(\theta)^2}\mathrm{d}\theta = \int_0^{\pi/2} \frac{1 + \tan^2\theta}{(1 + \tan^3\theta)^{4/3}} \sqrt{d^2 \tan^4\theta + e^2}\mathrm{d}\theta \tag{6}$$

However, there is a problem when using Eq. (6) as it is expressed, since, when $\theta \to \pi/2$, $\tan\theta \to \infty$, which poses a numerical problem when computing the integral. To circumvent this problem, the integral computation is divided in two parts by using a reflection, with respect to a line $x_2 = z_2$, transforming the curve F into a curve $F^\star$ (see Figure 3). This affine transformation creates an intersection point P, for which $\theta = 45° = \pi/4$. Obviously, because of the reflection properties, the length of the curve segment between points P and C is identical to the length of the curve segment between points P and $C^\star$. Therefore, the total Lamé curve length l_c can be computed as

$$\begin{aligned} l_c = &\int_0^{\pi/4} \frac{1 + \tan^2\theta}{(1 + \tan^3\theta)^{4/3}} \sqrt{d^2 \tan^4\theta + e^2}\mathrm{d}\theta \\ &+ \int_0^{\pi/4} \frac{1 + \tan^2\theta}{(1 + \tan^3\theta)^{4/3}} \sqrt{d^{\star 2} \tan^4\theta + e^{\star 2}}\mathrm{d}\theta \end{aligned} \tag{7}$$

where $d^\star = e$ and $e^\star = d$.

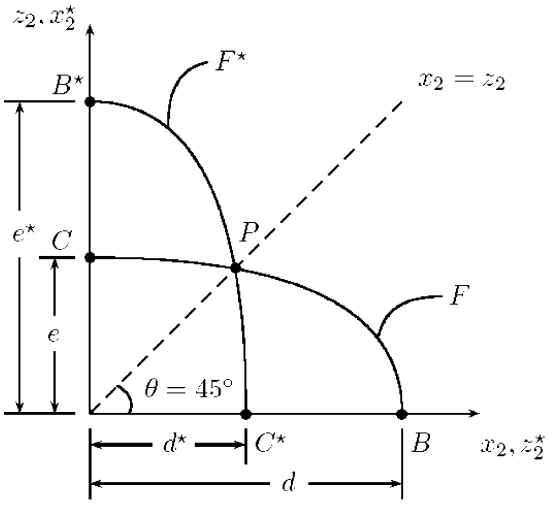


Fig. 3 Reflection of Lamé curve with respect to a line $x_2 = z_2$.

3 Trajectory Velocity Profile

To generate a motion along a trajectory, two important steps are required [5]: (1) to define a geometry parameterized with respect to the displacement s along the trajectory; and (2) to define a velocity profile stating the relation of this displacement over time. The Cartesian position, velocity and acceleration parameterization of the trajectory geometry, shown in Figure 2, with respect to the displacement s, starting from point A at $t = 0$, is explained in detail in [7].

The trajectory, as introduced in Figure 2, can be divided, either in translation or in rotation, into two periods: an acceleration period and a constant velocity period. To define the overall velocity profile of the trajectory, piecewise polynomials are used to blend these two periods [15]. In the translation case, since the MP has zero velocity at point A, it is obvious that an acceleration period follows any motion from point A. The acceleration period in translation ends either at point B or at point C, followed by a constant-velocity period. Now, for the rotation, since it is not possible to start the rotation from point A, any velocity profile first starts with a constant zero velocity period, followed by an angular acceleration period that can either end at point D or be followed by a maximum constant angular velocity period. For the velocity profile in this work, the angular acceleration period ends at point D.

To relate the accelerating displacement of the MP with time over the trajectory, a 4-5-6-7 interpolating polynomial is used in order to have smooth first-, second-, and third-order derivatives for the displacements $s(t)$ and $\phi(t)$, where $\phi(t)$ is the angular displacement of the MP as a function of time. A non-dimensional 4-5-6-7 polynomial is defined as

$$p(\tau) = -20\tau^7 + 70\tau^6 - 84\tau^5 + 35\tau^4 \tag{8}$$

for $\tau \in [0, 1]$, so that its first derivative, with respect to τ, starts with a value of zero at $\tau = 0$, reaches a maximum value of $35/16$ at $\tau = 0.5$ and then returns to zero at $\tau = 1$.

Using suitable scaling factors, time can be related to τ, as $s(t)$ or $\phi(t)$ can be related to $p(\tau(t))$. It is then possible to define the position, velocity and acceleration in translation or in rotation during their respective acceleration period. To scale the polynomial of Eq. (8) in displacement, one can use the scaling factors Δs_a and $\Delta\phi_a$, defined as

$$\Delta s_a = (s_e - s_s) \leq (h + l_c + c), \quad \Delta\phi_a = (\phi_e - \phi_s) = \frac{\pi}{2} \tag{9}$$

where s_s and s_e are, respectively, the starting and ending positions along the trajectory during the translation acceleration period, while $\phi_s = 0$ and $\phi_e = \pi/2$ are the equivalents during the rotation acceleration period. For the time scale, one can use Δt_t and Δt_r, defined as

$$\Delta t_t = (t_{te} - t_{ts}) \leq t_{AD}, \quad \Delta t_r = (t_{re} - t_{rs}) \leq t_{AD} \tag{10}$$

where t_{ts} and t_{te} are, respectively, the starting and ending times of the translation acceleration period, while t_{rs} and t_{re} are the equivalents for the rotation acceleration period. Only the first half of $p(\tau)$ as defined in Eq. (8) is required, i.e., for $\tau \in [0, 0.5]$, so that both acceleration periods start with zero velocity and finish at maximum velocity, with no discontinuity in the acceleration. As $p(0.5) = 0.5$, the scales, in displacement and time, have to be doubled so that the scaled τ and $p(\tau)$ values correspond to the desired time and displacements values. Then, to describe the translation displacement, velocity and acceleration with respect to time during the acceleration period, one obtains

$$s_a(t) = 2\Delta s_a\, p(\tau) \tag{11a}$$

$$\dot{s}_a(t) = (\Delta s_a/\Delta t_t)(\mathrm{d}p(\tau)/\mathrm{d}\tau) \tag{11b}$$

$$\ddot{s}_a(t) = (\Delta s_a/2(\Delta t_t)^2)(\mathrm{d}^2 p(\tau)/\mathrm{d}\tau^2) \tag{11c}$$

where

$$\tau = \tau(t) = t/2\Delta t_t, \quad 0 \leq t \leq \Delta t_t \tag{12}$$

Similarly, for the rotation part of the trajectory,

$$\phi_a(t) = 2\Delta\phi_a\, p(\sigma) \tag{13a}$$

$$\dot{\phi}_a(t) = (\Delta\phi_a/\Delta t_r)(\mathrm{d}p(\sigma)/\mathrm{d}\sigma) \tag{13b}$$

$$\ddot{\phi}_a(t) = (\Delta\phi_a/2(\Delta t_r)^2)(\mathrm{d}^2 p(\sigma)/\mathrm{d}\sigma^2) \tag{13c}$$

where

$$\sigma = \sigma(t) = (t - t_{rs})/\Delta t_r, \quad 0 \leq t_{rs} \leq t \leq \Delta t_r + t_{rs} \tag{14}$$

The variable σ is used in Eq. (13), instead of τ, to emphasize that the translation and rotation acceleration periods are independent. In Eq. (14), the only case for which $t_{rs} = 0$ is when the design variable $h = 0$, making point B coincide with point A in Figure 2.

Using the properties of the 4-5-6-7 polynomial, the expressions for the maximum velocity and angular velocity are

$$\dot{s}_{\max} = (35/16)(\Delta s_a/\Delta t_t) \tag{15a}$$

$$\dot{\phi}_{\max} = (35/16)(\Delta \phi_a/\Delta t_r) \tag{15b}$$

Eqs. (11) to (14) are used to determine specific velocity profiles depending on when the payload accelerates and when it coasts or decelerates. The velocity profile proposed in this work, has an acceleration in translation from point A to point C, followed by a displacement at maximum velocity $\dot{s}_{\max}$ up to point D, while still accelerating in rotation from point B to point D. This means that $\Delta t_t = t_{AC} = t_{AD} - t_{CD}$ and $\Delta s_a = h + l_c$. Since the displacement at constant maximum velocity $\dot{s}_{\max}$ is from point C to point D, an expression for t_{CD} can readily be found, namely,

$$t_{CD} = c/\dot{s}_{\max} \tag{16}$$

If Eq. (15a) is applied here, one obtains

$$\dot{s}_{\max} = (35/16)(h + l_c)/(t_{AD} - t_{CD}) \tag{17}$$

Upon inserting Eq. (16) into Eq. (17), $\dot{s}_{\max}$ can be found in terms of the trajectory parameters c and h, namely,

$$\dot{s}_{\max} = (1/t_{AD})\left[(35/16)\,(h + l_c) + c\right] \tag{18}$$

It is now possible to define the translation velocity profile in terms of time and the trajectory parameters c and h, namely,

$$s(t) = 2(h + l_c)\,p(\tau) \tag{19a}$$

$$\dot{s}(t) = ((h + l_c)/t_{AC})(\mathrm{d}p(\tau)/\mathrm{d}\tau) \tag{19b}$$

$$\ddot{s}(t) = ((h + l_c)/2{t_{AC}}^2)(\mathrm{d}^2 p(\tau)/\mathrm{d}\tau^2) \tag{19c}$$

for $t \in [0, t_{AC}]$, with $\tau = t/2t_{AC}$, and

$$s(t) = h + l_c + c(t - t_{AC})/t_{CD} \tag{20a}$$

$$\dot{s}(t) = \dot{s}_{\max} \tag{20b}$$

$$\ddot{s}(t) = 0 \tag{20c}$$

Now, referring to Eqs. (13a), (13b) and (13c), the rotation velocity profile for $\phi(t)$ can be prescribed as

$$\phi(t) = \dot{\phi}(t) = \ddot{\phi}(t) = 0 \tag{21}$$

for $t \in [0, t_{AB}]$, and as

$$\phi(t) = \pi\, p(\sigma) \tag{22a}$$

$$\dot{\phi}(t) = (\pi/2t_{BD})(\mathrm{d}p(\sigma)/\mathrm{d}\sigma) \tag{22b}$$

$$\ddot{\phi}(t) = (\pi/(2t_{BD})^2)(\mathrm{d}^2 p(\sigma)/\mathrm{d}\sigma^2) \tag{22c}$$

for $t \in [t_{AB}, t_{AD}]$, with $\sigma = (t - t_{AB})/2t_{BD}$. Since point B is reached somewhere during the acceleration period, it is difficult to express t_{AB} and $t_{BD} = t_{AD} - t_{AB}$. What is known is

$$s(t_{AB}) = 2(h + l_c)p(\tau_{AB}) = h \tag{23}$$

where τ_{AB} is the value of Eq. (12) for $t = t_{AB}(c, h)$. This means that to solve Eq. (23) for t_{AB}, one needs to find the seven roots of the polynomial, which has to be done numerically, and keep the one that is real and whose value lies between $t = 0$ and $t = t_{AC}$. With the value of t_{AB} thus found, the value of t_{BD} can be computed, which allows the results of Eqs. (22a–22c) to be computed for the angular velocity profile.

4 Trajectory Optimization

A SCARA system produces motions with four-dof, three for positioning, one for orienting of the MP. If vector $\mathbf{v}$ denotes the translation velocity of the MP and ω its angular velocity around the z vertical axis of rotation with respect to the ground, then the kinetic energy of the MP is defined as

$$T = \frac{1}{2}m\|\mathbf{v}\|^2 + \frac{1}{2}I_c\omega^2 = \frac{1}{2}m\dot{s}^2 + \frac{1}{2}I_c\dot{\phi}^2 \tag{24}$$

where m denotes the mass of the payload, including that of the MP, and I_c denotes the moment of inertia of the same about a vertical axis passing through its mass center. The moment of inertia can also be expressed as $I_c = mr_c^2$, where r_c denotes the radius of gyration of the payload. Moreover, since the objective is to minimize the variation in kinetic energy over the trajectory, the derivative of T with respect to time is required, namely

$$\dot{T} = m\left(\dot{s}\ddot{s} + r_c^2\dot{\phi}\ddot{\phi}\right) \tag{25}$$

where $s = s(t, c, h)$ and $\phi = \phi(t, c, h)$ are, respectively, the displacement in translation along the trajectory from point A to point D and the rotation of the end-effector. For the given velocity profile, the rms value of time-derivative of the kinetic energy is obtained, which is minimized in order to have the lowest possible variation in kinetic energy over the trajectory. Therefore, using the rms^2 value of the foregoing time-derivative over the test trajectory, the optimization problem is formulated as

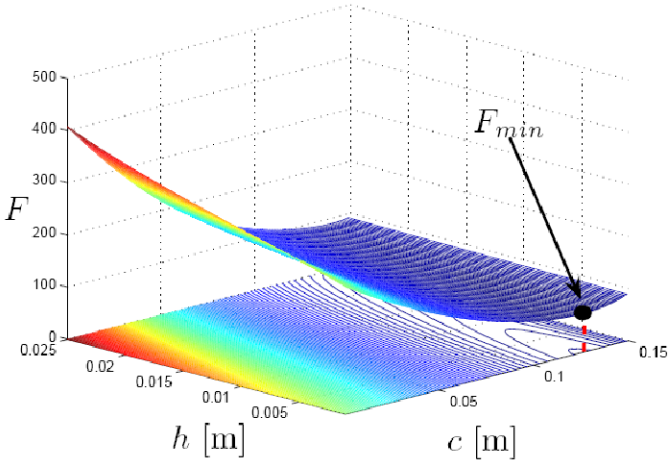


Fig. 4 The objective function as a function of c and h.

$$F(\mathbf{x}) = \frac{1}{m^2}\int_0^{t_{AD}} \dot{T}(t,\mathbf{x})^2 \mathrm{d}t = \int_0^{t_{AD}} \left(\dot{s}\ddot{s} + r_c^2\dot{\phi}\ddot{\phi}\right)^2 \mathrm{d}t \quad \rightarrow \quad \min_{\mathbf{x}} \tag{26}$$

subject to the inequality constraints $c \geq 0$ and $h \geq 0$. Notice that (a) because of the definition of the objective function $F(\mathbf{x})$, this function is independent of the mass and (b) the radius of gyration acts as a weighting factor between the translational and the rotational parts of the objective function.

Referring to Figure 2, t_{AD} is the time required to move from point A to point D and is fixed, while $\mathbf{x} = [\,c\ h\,]^T$ is the design vector containing the design variables which will be optimized. The objective function in Eq. (26) is independent of the mass of the payload, but it does depend on the radius of gyration of the payload, which is actually defined by the properties of the product that has to be manipulated through the trajectory by the SCARA system. In this work, the payload is modeled as a circular cylinder, that has a mass of 2 kg and a diameter of 65 mm, for which $r_c = 23$ mm.

For the velocity profile, plotting the objective function as a function of c and h (see Figure 4) shows that the objective function is smooth and has a well defined minimum value. Using the Orthogonal Decomposition Algorithm [2] along with slack variables, to transform the inequality constraints into equality constraints, the optimum values were determined to be $\mathbf{x} = [126.1\ \ 0.0000]^T$ mm. Since $h = 0$, the MP starts to translate in the x direction as soon as the PPO motion starts.

5 Conclusions

A general formulation of a smooth curve to implement the test trajectory of SCARA systems was derived using Lamé curves. The parameters of the Lamé curves were optimized by minimizing the time-derivative of the root-mean-square (rms) value of the kinetic energy using the Orthogonal Decomposition Algorithm to achieve an optimal test trajectory for the proposed velocity profile.

References

1. Angeles, J., Alivizatos, A. and Zsombor-Murray, P., The synthesis of smooth trajectories for pick-&-place operations. *IEEE Transactions on Systems, Man, and Cybernetics* **18**(1), 173–178 (1988).
2. Angeles, J., Anderson, K. and Gosselin, C., Constrained design optimization using orthogonal decomposition. *ASME Journal of Mechanisms, Transmissions, and Automation in Design* **112**(2), 255–256 (1990).
3. Angeles, J., Caro, S., Khan, W. and Morozov, A., Kinetostatic design of an innovative Schönflies motion generator. *Journal of Mechanical Engineering Science* **220**, 935–943 (2006).
4. Chang, Y.-H., Lee, T. and Liu, C.-H., On-line approximate Cartesian trajectory planning for robotic manipulator. *IEEE Transactions on Systems, Man, and Cybernetics* **22**(3), 542–547 (1992).
5. Codourey, A., Contribution à la commande des robots rapides et précis, PhD Thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne (1991).
6. Constantinescu, D. and Croft, E., Smooth and time-optimal trajectory planning for industrial manipulators along specified paths. *Journal of Robotic Systems* **17**(5), 233–249 (2000).
7. Gauthier, J., Angeles, J. and Nokleby, S., Optimization of a test trajectory for SCARA systems, Technical Report TR-CIM-07-01, McGill University, Montreal (2007).
8. Khan, W., The conceptual design of robotic architectures using complexity criteria, PhD Thesis, Department of Mechanical Engineering, McGill University, Montreal (2007).
9. Lin, C.-S., Chang, P.-R. and Luh, J., Formulation and optimization of cubic polynomial joint trajectories for industrial robots. *IEEE Transactions on Automatic Control* **28(12)**, 1066–1073 (1983).
10. Luh, J. and Lin, C., Approximate joint trajectories for control of industrial robots along Cartesian paths. *IEEE Transactions on Systems, Man, and Cybernetics* **14**, 444–450 (1984).
11. Paul, R., Manipulator Cartesian path control. *IEEE Transactions on Systems, Man, and Cybernetics* **9**, 702–711 (1979).
12. Shin, K. and McKay, N., A dynamic programming approach to trajectory planning for robotic manipulator. *IEEE Transactions on Automatic Control* **31**(6), 491–500 (1986).
13. Tan, H. and Potts, H., Minimum time trajectory planner for the discrete dynamic robot model with dynamic constraints. *IEEE Transactions on Robotics and Automation* **4**(2), 174–185 (1988).
14. Taylor, R., Planification and execution of straight line manipulator trajectories. *IBM Journal of Research and Development* **23(4)**, 424–436 (1979).
15. Wang, L. and Kuo, M., Time-optimal constant speed motion program for multiple cooperating manipulators. *Journal of Robotic Systems* **16**(3), 185–194 (1999).

Singularity Free Path Planning for Parallel Robots

Samir Lahouar[1], Saïd Zeghloul[1] and Lotfi Romdhane[2]

[1]*Laboratoire de Mécanique de Solides, Université de Poitiers, 86962 Futuroscope Chasseneuil Cedex, France, e-mail: { samir.lahouar, said}@lms.univ-poitiers.fr*
[2]*Laboratoire de Génie Mécanique, Ecole Nationale d'Ingénieurs de Sousse, Sousse 4000, Tunisie, e-mail: lotfi.romdhane@enim.rnu.tn*

Abstract. In this paper, we present a procedure to automatically generate the kinematic model of parallel mechanisms as well as an algorithm for their singularity free path planning. Singular positions are considered as obstacles that have to be bypassed while moving toward the goal. The 3-RPR planar parallel robot was taken as an example to illustrate the effectiveness of the procedure. This proposed method can be easily extended to other similar parallel mechanisms.

Key words: singularity, path planning, parallel robots.

1 Introduction

Since the 1960s, serial robots have taken a great place in industry. They are used in many industrial applications and in many domains replacing men in hard tasks and hostile environments. However, in the last 20 years, researchers have been interested in another type of robots: parallel robots. Different structures of parallel manipulators have been studied (see, for example Clavel, 1988; Goudali et al., 1995; Hesselbach et al., 1998). Currently, more than 100 different architectures have been proposed and probably not all of them have been discovered. A review of most known architectures is given by Merlet (2000).

Singularities may appear within the workspace. Singularities are dangerous and thus should be detected and avoided while moving a parallel manipulator. This problem is very important and is widely studied in literature; see for instance Sen et al. (1998) and Guanfeng et al. (2003).

Path planning in the case of parallel manipulators needs to include singularity avoidance. Singularities are configurations where the robot becomes uncontrollable. Many works considered the problem of detecting singularities in the case of parallel manipulators (see Angeles et al., 2003; Li et al., 2006). Merlet (2001) proposed a method to verify that a given trajectory is valid which means that it does not intersect any singularity. Some researchers were interested in generating a singularity free path (Dasgupta et al., 1998; Sen et al., 2003; Dash et al., 2005) Whereas Cortès (2003) studied the problem of collision free path planning for parallel manipulators

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 235–242.

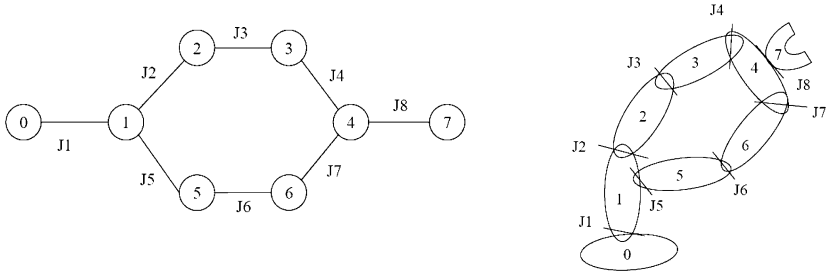


Fig. 1 A mechanism and its corresponding graph.

and closed kinematic chains, he tried to generalize methods used with serial manipulators but he did not consider the problem of singularity. However there are few works that deal with both problems singularity and collision avoidance.

In Section 2, we present a method to derive the kinematic model of parallel mechanisms. This model is then used to detect singularity and to generate a desired path. In Section 3, the 3-RPR planar parallel robot is used as an example to illustrate the proposed method. Some concluding remarks are then presented at the end of this paper.

2 Problem Formulation

2.1 Kinematic Model

A multibody mechanism is made of links and joints. The structure of a mechanism can be presented by a graph as shown in Figure 1.

Each link has a position and an orientation in the 3D space. In order to determine the position and the orientation of the link we can use either absolute coordinates or joint coordinates.

We define the base link as the first link of a robot. Its position and orientation are given with respect to the environment using absolute coordinates. In order to place the robot we have only to move the base link. The base link is unique for each robot.

A serial link is held to another link (called father link) according to the joint between both links.

Supported joints are revolute joints, linear joints, spherical joints and universal joints.

A mobile link is positioned using absolute coordinates relative to the base link. Constraints are used to restrict motion between two different links.

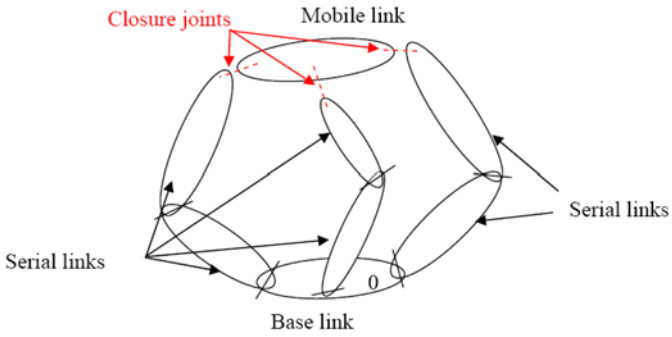


Fig. 2 Modeling method.

2.2 Generating Closure Equations

As shown in Figure 2, a parallel manipulator is made of a base platform, a mobile platform and serial legs. The idea is to replace a non actuated joint in each leg by the corresponding constraints. These joints are called closure constraints.

The base platform is modeled using a base link while the mobile platform is modeled by a mobile link. Legs are modeled using serial links.

By replacing only non actuated joints by constraints, we keep the actuated joint coordinates in the generated equations.

Constraints between mobile links and some serial links give equations on operational variables and joint coordinates.

2.3 Extracting the Kinematic Model

The number of equations depends on the constraints used to model the mechanism. A constraint of coincidence between two points gives three equations, which is equivalent to a constraint of parallelism between two vectors. While a constraint of coincidence between two links gives six equations.

The variables used to describe the motion of the mechanism depend on the links used to build the model. A mobile link has six variables. The number of variables given by a serial link depends on the nature of its corresponding joint. For a revolute and a linear joint there is only one variable while for spherical joint there are three variables and for universal joint there are two variables.

We call $\mathbf{X}$ the vector of variables. The equations can be written as

$$F(\mathbf{X}) = 0, \tag{1}$$

where F is a vector function.

In order to obtain the kinematic model we only have to derive the function F with respect to time, which yields:

$$\mathbf{E}_k\dot{\mathbf{X}} = 0. \tag{2}$$

$\dot{\mathbf{X}}$ is the derivative of vector $\mathbf{X}$ with respect to time.

The mobility of the system is given by

$$m = N - rg(\mathbf{E}_k), \tag{3}$$

where N is the dimension of the vector $\mathbf{X}$.

By rearranging Equation (2) we can write

$$\mathbf{M}\dot{\mathbf{q}}_a + \mathbf{N}\dot{\mathbf{q}}_p + \mathbf{K}\dot{\mathbf{X}}_{op} = 0, \tag{4}$$

where $\dot{\mathbf{q}}_a$ is the vector of actuated variables, $\dot{\mathbf{X}}_{op}$ the vector of output variables while $\dot{\mathbf{q}}_p$ is the vector of the remaining variables, we call them passive variables.

$\mathbf{M}$ and $\mathbf{K}$ should be full rank matrices otherwise an actuated variable could be changed without moving the mobile platform or an output variable could be changed while actuators are locked, which means that the model does not completely describe the behavior of the mechanism.

$\mathbf{M}$, $\mathbf{N}$ and $\mathbf{K}$ are evaluated at each step while moving the multibody mechanism.

In order to obtain the classical form of the kinematic model, it is necessary to eliminate the matrix $\mathbf{N}$ from Equation (5). To do so, a matrix $\mathbf{O}$ is computed. The columns of the matrix $\mathbf{O}$ are generated in the null space of matrix $\mathbf{N}$ using Gram–Schmidt orthonormalization. We have:

$$\mathbf{O}^T\mathbf{N} = 0 \quad \text{and} \quad \text{rank}(\mathbf{O}^T\mathbf{K}) = \text{rank}(\mathbf{K}), \tag{5}$$

$$\mathbf{B}\dot{\mathbf{q}}_a + \mathbf{A}\dot{\mathbf{X}}_{op} = 0, \tag{6}$$

$$\mathbf{B} = \mathbf{O}^T\mathbf{M} \tag{7}$$

$$\mathbf{A} = \mathbf{O}^T\mathbf{K}. \tag{8}$$

Once $\dot{\mathbf{q}}_a$ and $\dot{\mathbf{X}}_{op}$ are known, $\dot{\mathbf{q}}_p$ are computed by solving Equation (4).

While moving the mechanism, singularities could be detected. They correspond to a loss of rank of one of the matrices $\mathbf{N}$, $\mathbf{A}$ or $\mathbf{B}$.

The rank of these matrices is continuously checked while moving the multibody mechanism.

2.4 Path Planning

We propose to adapt path planning concept and procedures that are used for serial manipulators, as outlined in Lahouar et al. (2006). This method is based on the alternation of two searching modes. The first one is a depth search mode active

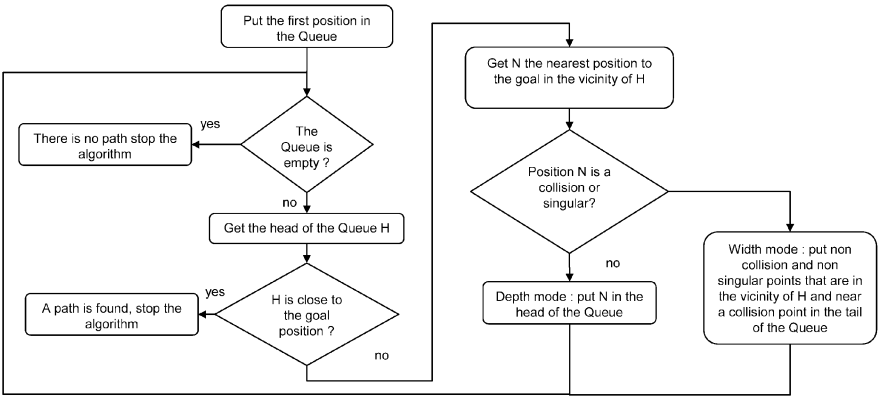


Fig. 3 Path planning algorithm.

when the robot is far from obstacles, so it evolves toward its goal. The second one is a width search mode when the robot is near an obstacle it permits to find the best way to avoid the obstacle. Although a grid is needed, it is not necessary to construct it before beginning the path planning and the collision test is checked only for some points which are selected by the algorithm. In the case of serial manipulators the configuration space is used with this method as it is easier than planning in the workspace. This is no longer justified in the case of parallel robots since forward kinematics are more difficult than inverse kinematics. Figure 3 depicts the algorithm used in generating obstacle and singularity free paths.

3 Application

The robot (Figure 4) is modeled by using a mobile link and two serial links for each leg. There are three constraints between each leg and the mobile link which gives the following geometrical equations:

$$\begin{cases} q_1 \cos q_{p1} - x = 0, \\ q_1 \sin q_{p1} - y = 0, \\ q_2 \cos q_{p2} + L - x - l\cos\theta = 0, \\ q_2 \sin q_{p2} - y - l\sin\theta = 0, \\ q_3 \cos q_{p3} + \frac{L}{2} - x - l\cos(\theta + \pi/3) = 0, \\ q_3 \sin q_{p3} + L - y - l\sin(\theta + \pi/3) = 0. \end{cases} \tag{9}$$

By deriving these equations kinematic equations are obtained.

$$\mathbf{M}\dot{\mathbf{q}}_a + \mathbf{N}\dot{\mathbf{q}}_p + \mathbf{K}\dot{\mathbf{X}}_{op} = 0. \tag{10}$$

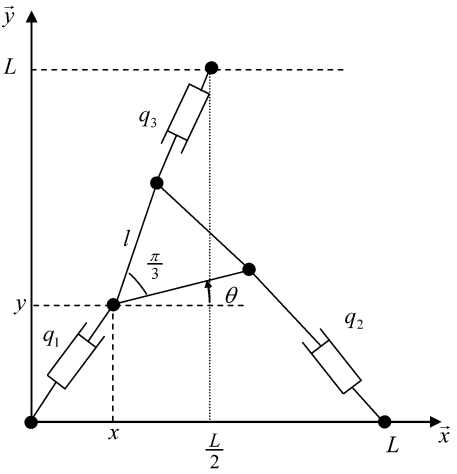


Fig. 4 The 3-RPR parallel robot.

In order to eliminate the passive variables we use the Gramm–Schmidt orthogonalization process in order to find a matrix $\mathbf{O}$ verifying the following constraints:

$$\mathbf{O}^T\mathbf{N} = 0 \quad \text{and} \quad \text{rank}(\mathbf{O}^T\mathbf{K}) = \text{rank}(\mathbf{K}). \tag{11}$$

Such matrix exists if the robot is not in a singular position. In this case, we can find easily a matrix $\mathbf{O}$ verifying these constraints and without using the orthogonalization process, which yields:

$$\dot{\mathbf{q}}_a = -\mathbf{B}\dot{\mathbf{X}}_{op}. \tag{12}$$

3.1 Singularities

For this case the matrix $\mathbf{B}$ is written as follows:

$$\mathbf{B} = \begin{pmatrix} x & y & 0 \\ x + l\cos\theta - L & y + l\sin\theta & l(L-x)\sin\theta + ly\cos\theta \\ x + l\cos\left(\theta + \frac{\pi}{3}\right) - \frac{L}{2} & y + l\sin\left(\theta + \frac{\pi}{3}\right) - L & -l\sin\left(\theta + \frac{\pi}{3}\right)\left(x - \frac{L}{2}\right) + \\ & & + l\cos\left(\theta + \frac{\pi}{3}\right)(y - L) \end{pmatrix}. \tag{13}$$

Singular positions are given by the following equation:

$$\det(\mathbf{B}) = 0. \tag{14}$$

These positions divide the workspace into two disjoint regions as shown in Figure 5.

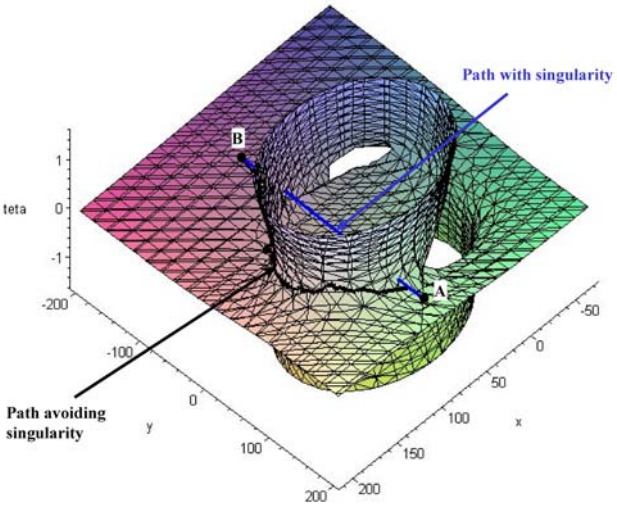


Fig. 5 Path planning results.

3.2 Path Planning

Figure 5 shows a complex surface representing all positions and orientations (x, y, θ) of the platform corresponding to a singularity. The path starts at point A $(100, 180, 0.3)$ and ends in point B $(100, -100, 0.4)$. When the singularity is ignored during the path planning, we can notice that the obtained path is a straight line, which is the shortest path between A and B (Figure 5). However, when the singularity is taken into account, we can notice that the end effector avoids the singularity as it was an obstacle yielding the second path shown in Figure 5.

4 Conclusions

We presented a new tool for modeling parallel mechanisms with complex geometry. The closure equations are automatically generated. The obtained model was used to generate a singular free path. Singular positions are considered as obstacles that have to be bypassed while moving toward the goal. The 3-RPR planar parallel robot was taken as an example to illustrate the effectiveness of the procedure. This proposed method can be easily extended to other similar parallel mechanisms.

References

Angeles, J., Yang, G., Chen, I-M. (2003), Singularity analysis of three-legged, six-dof platform manipulators with URS legs. *IEEE/ASME Transactions on Mechatronics* **8**(4), 469–475.

Clavel, R. (1988), DELTA, A fast robot with parallel geometry. In *Proceedings 18th International Symposium on Industrial Robot*, Lausanne, pp. 91–100.

Cortés, J. (2003), *Motion Planning Algorithms for General Closed-Chain Mechanisms*. PhD Thesis, INP, Toulouse.

Dasgupta, B., Mruthyunjaya, T. (1998), Singularity-free path-planning for the Stewart platform manipulator. *Mechanism and Machine Theory* **33**(6), 711–725.

Dash, A.K., Chen, I-M., Yeo, S.H., Yang, G. (2005), Workspace generation and planning singularity-free path for parallel manipulators. *Mechanism and Machine Theory* **40**, 776–805.

Goudali, A., Lallemand, J-P., Zeghloul, S. (1995), Espace de travail de la nouvelle structure 2-Delta. *Revue d'Automatique et de Productique Appliquée* **8**(2/3), 205–210.

Guanfeng, L., Yunjiang, L., Zexiang, L. (2003), Singularities of parallel manipulators: a geometric treatment. *IEEE Transactions on Robotics and Automation* **19**(4), 579–594.

Hesselbach, S., Plitea, N., Frindt, M., Kusiek, A. (1998), A new parallel mechanism to use for cutting convex glass panels. In *Advances in Robot Kinematics: Analysis and Control*, J. Lenarčič, M.L. Husty (Eds.), Kluwer Academic Publishers, pp. 165–174.

Lahouar, S., Zeghloul, S., Romdhane, L. (2006), Collision free path planning for multi-DoF manipulators. In *Industrial Robotics: Theory, Modeling and Control*, S. Cubero (Ed.), Pro Literatur Verlag, pp. 349–378.

Li, H., Gosselin, C., Richard, M. (2006), Determination of maximal singularity-free zones in workspace of planar three-degree-of-freedom parallel mechanisms. *Mechanism and Machine Theory* **41**(10), 1157–1167.

Merlet, J.-P. (2000), *Parallel Robots*, Kluwer Academic Publishers.

Merlet, J.-P. (2001), A generic trajectory verifier for the motion planning of parallel robots. *Transactions of the ASME* **123**(4), 510–515.

Sen, D., Mruthyunjaya, T.S. (1998), A centro-based characterization of singularities in the workspace of planar closed-loop manipulators. *Mechanism and Machine Theory* **33**(8), 1091–1104.

Sen, T.S., Drasgupta, B., Mallik, A.K. (2003), Variational approach for singularity-free path-planning of parallel manipulators. *Mechanism and Machine Theory* **38**, 1165–1183.

A Comparison between Two Motion Planning Strategies for Kinematically Redundant Parallel Manipulators

J.A. Carretero[1], I. Ebrahimi[1] and R. Boudreau[2]

[1]*Department of Mechanical Engineering, University of New Brunswick, Fredericton, New Brunswick, Canada E3B 5A3, e-mail: {juan.carretero, iman.ebrahimi}@unb.ca*
[2]*Département de Génie Mécanique, Université de Moncton, Moncton, New Brunswick, Canada E1A 3E9, e-mail: roger.a.boudreau@umoncton.ca*

Abstract. In this work, a new approach for motion planning of kinematically redundant parallel manipulators is proposed and compared with a method previously proposed by the authors. First, motion planning is defined and thereafter overall motion planning (OMP) is introduced. OMP consists of determining actuation schemes that optimise the manipulator's performance while considering the entire given trajectory of the end-effector at once. Next, the OMP method is compared to point-to-point motion planning. Two examples are given to compare the results of both methods. It is shown that the proposed OMP method can generate actuation schemes for given trajectories such that the manipulator avoids singular configurations.

Key words: path planning, actuation schemes, parallel manipulators, kinematic redundancy.

1 Introduction

In order for a manipulator to perform a task inside its workspace, its motion has to be planned and actuation schemes for the actuators must be determined. Motion planning and actuation schemes are important problems in robotics requiring good knowledge of the manipulator's kinematics. Motion planning has been addressed extensively for serial manipulators [1–3].

In motion planning, obstacle avoidance and self-collision are critical issues for both serial and parallel manipulators. In addition to that, for parallel manipulators, motion planning strictly depends on satisfying kinematic constraints [4] rendering motion planning more difficult. One of the motion planning methods for closed-loop mechanisms is called *probabilistic roadmap*. This method does not require any structural information about the manipulator's configuration space; the only primary requirement is the ability to determine whether a given configuration is collision-free or not. The roadmap can be used for many kinds of robots with arbitrary degrees of freedom. In order to choose the nodes for the roadmap, the method uses a probabilistic approach. It generates random configurations, examines if they are collision-free and if so, integrates them into the roadmap [5–7].

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 243–252.

A local motion planner was proposed in [4]. Its application to non-redundant manipulators yielded safe trajectories between two points which do not trespass voids inside the manipulator's workspace.

Motion planning for kinematically redundant planar parallel manipulators was studied in [8, 9]. There, an objective function based on the proximity to singular configurations was defined. It was then used to plan actuation schemes that avoid singular configurations that cannot be avoided by non-redundant manipulators. The approach was called *point-to-point motion planning* (PPMP) whereby every actuation move is decided by maximising the distance to singular configurations based on the current pose of the manipulator.

Redundancy in parallel manipulators was proposed in [10–12]. For parallel manipulators, redundancy can be separated into actuation redundancy and kinematic redundancy. *Actuation redundancy* is defined as replacing existing passive joints by active ones. Therefore, actuation redundancy does not change the mobility or force workspace of a manipulator but may result in singularity reduction [13, 14]. *Kinematic redundancy* takes place when extra active joints and links are added. Kinematic redundancy increases the mobility and results in an infinite number of solutions for the *inverse displacement problem* (IDP). Adding kinematic redundancy has various advantages such as avoiding most kinematic singularities, enlarging workspace, as well as improving dexterity [15].

In serial manipulators, singular configurations solely depend on the relative locations of the joint axes [16]. On the other hand, for parallel manipulators, singular configurations not only depend on the relative locations of the joints' axes of one branch, they also depend on the relative configurations of the other branches.

In this work, a new *overall motion planning* (OMP) is proposed for kinematically redundant parallel manipulators. Considering a defined objective function, PPMP can optimise the actuators' next displacement values based on their current values. Note that the solutions found at a current time-step may not result in optimal values for the future time-steps. On the other hand, OMP considers the entire trajectory as an objective function. This way, OMP also considers what is ahead on a trajectory to find an optimal set of actuation schemes. Here, the OMP approach is first introduced and qualitatively compared to the PPMP method. Results obtained for two different trajectories are also used to compare the PPMP and OMP methods.

2 Motion Planning

Here, motion planning is defined as finding the actuation schemes of the active joints of the manipulator such that the defined motion constraints for the manipulator will be satisfied for a given path of the end-effector. Kinematic redundancy for parallel manipulators results in having an infinite number of solutions for the IDP [15]. Having a locus or loci of solutions for the IDP, motion planning can be converted to finding the best possible set of solutions of the IDP.

2.1 Overall Motion Planning (OMP)

The OMP and PPMP methods take advantage of the existence of loci of IDP solutions for the actuators of kinematically redundant parallel manipulators. Knowing the trajectory of a manipulator's end-effector, it is possible to obtain the extreme values (i.e., their upper and lower bounds) of the active joints for the manipulator from the solution of the IDP. Next, based on the possible range of action of the actuators for every point of the trajectory, it is possible to estimate the actuation scheme of each actuator with n adjacent line segments (a.k.a. polyline) whose $n + 1$ endpoints will be called *Control-Points* or simply CPs. The following algorithm describes the OMP method for a kinematically redundant parallel manipulator with one degree of kinematic redundancy per limb (in most cases, this would mean two actuators per limb). The algorithm can be expanded to more than two actuators per limb but it is more complicated and also more computationally intensive.

1. Find the locus of IDP solutions for the given path for each actuated joint of a Kinematically Redundant Limb (KRL).
2. From the locus of IDP solutions for each end effector pose, determine the range of motion of all actuators of each KRL.
3. Select one of the actuators of each KRL as the optimisation variable.
4. Choose an initial set of values for the $n + 1$ CPs for each KRL.
5. Connect the $n + 1$ CPs with n line segments to create the polyline approximation of the actuator trajectory.
6. Verify if any point along the polyline violates the actuator boundaries. If so, replace infeasible actuator values by the boundary values.
7. Discretise each line segment of the polyline into N points and calculate the profile for the optimised actuator of each KRL, then, calculate the other actuators' profiles.
8. Compute an objective function, such as proximity to singular configurations, energy consumption or time, at each discretised point along the path.
9. Using an optimisation algorithm, find the optimal location of the CPs. Steps 5 to 8 are repeated every time the objective function is calculated.

The number of CPs can greatly affect the final results. For example, if the number of CPs is too low, proper results may not be obtained. On the other hand, a very high number of CPs is very computationally intensive. Therefore, the number of CPs of the actuation schemes for a given trajectory should be selected with care.

To reduce the computational burden during the search for optimal actuation schemes, these are estimated by polylines which connect the CPs with straight line segments. Once the optimisation process is finalised, the polyline is replaced by a high order polynomial or spline passing through all $n + 1$ control points. This ensures continuity along the path in terms of joint velocities and accelerations.

2.2 Comparison of PPMP and OMP Methods

As explained in [9], the PPMP method can avoid singularities and improve the actuation schemes for a given trajectory. However, when using PPMP, solutions found at a current time-step may not be optimal in future time-steps, thus producing configurations that are not optimal when the complete path is considered. PPMP is more sensitive to the initial pose of the manipulator than OMP since once the motion is initiated, it is not feasible to change the previous steps for PPMP. On the other hand, the OMP considers the quality of the trajectory as a whole as the objective function and after completion of the calculations, the motion commences. It can change the initial pose of the manipulator such that it suits the whole trajectory.

Note that PPMP can be a real-time motion planner while OMP is an off-line one. OMP generally requires more computation than PPMP. There are cases in which PPMP and OMP have very similar results.

3 Kinematically Redundant 3-RPRR Manipulator

The 3-RPRR kinematically redundant planar parallel manipulator was proposed in [9] where it was fully analysed from the point of view of kinematics, singularities and workspace. Starting at the base, each of the 3-RPRR limbs has an active revolute joint at point A_i followed by a prismatic actuator with length ρ_i (where i is the branch index and $i = 1, 2, 3$). At the end of the prismatic joint there is a revolute joint at D_i followed by the distal link joining branch i to the end effector at B_i. The 3-RPRR manipulator is illustrated in Figure 1 where the solid circles represent active revolute joints and the empty ones represent passive joints.

The 3-RPRR parallel manipulator (Figure 1) stems from the non-redundant 3-PRR planar parallel manipulator proposed in [17] whereby an active revolute joint is added at the base allowing to actively vary the direction of the prismatic joint. This change greatly improves the size and quality of its workspace [9].

The 3-RPRR has six *actuated-joint degrees of freedom* of which three are considered to be kinematically redundant. As mentioned earlier, kinematically redundant manipulators provide an infinite choice for the solution of the inverse displacement problem. As a result, an objective function may be defined in order to ensure the path with the best characteristics is selected. As in [9], an objective function based on proximity to singular configurations is selected. Unlike the PPMP objective function in [9], the objective function for the OMP proposed here must consider proximity to singular configurations for the entire trajectory.

To define the objective function, it is necessary to briefly explain the kinematic equations of the 3-RPRR manipulator. The vector loop equation for branch i of the manipulator can be stated as:

$$\overline{D_i B_i} = \overline{D_i A_i} + \overline{A_i O} + \overline{O P} + \overline{P B_i} \tag{1}$$

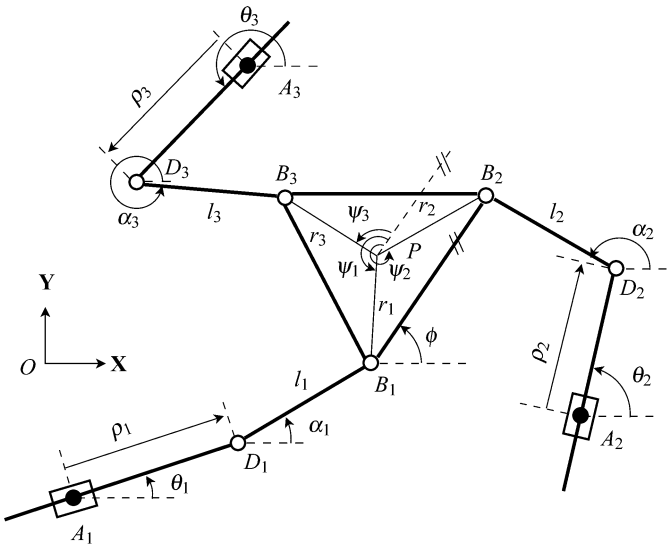


Fig. 1 3-RPRR planar 6-DOF kinematically redundant parallel manipulator (if θ_is were fixed, the manipulator would be a 3-PRR).

$$l_i{}^2 = \left(-\rho_i c_{\theta_i} - x_{A_i} + x_p + r_i c_{(\phi+\psi_i)}\right)^2 + \left(-\rho_i s_{\theta_i} - y_{A_i} + y_p + r_i s_{(\phi+\psi_i)}\right)^2 \quad (2)$$

$$l_i{}^2 = x_{l_i}{}^2 + y_{l_i}{}^2 = (l_i c_{\alpha_i})^2 + (l_i s_{\alpha_i})^2 \quad (3)$$

where c_* and s_* represent $\cos(*)$ and $\sin(*)$, respectively. Taking the derivative of Equations (1) and (2) for all the manipulator's branches with respect to time results in an equation which transforms the velocity vector of the active joints into the velocity vector of the end-effector and vice-versa. The equation can be shown as:

$$\mathbf{J_x}\dot{\mathbf{x}} = \mathbf{J_q}\dot{\mathbf{q}} \quad (4)$$

$\mathbf{J_x}$ and $\mathbf{J_q}$ are respectively referred to as the direct and inverse Jacobian matrices and can be expressed as:

$$\mathbf{J_x} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{31} & a_{33} \end{bmatrix}_{3\times 3} \quad (5)$$

$$\mathbf{J_q} = \begin{bmatrix} u_1 & v_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & u_2 & v_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & u_3 & v_3 \end{bmatrix}_{3\times 6} \quad (6)$$

where

$$\begin{aligned} a_{i1} &= x_p - x_{A_i} - \rho_i c_{\theta_1} + r_i c_{(\phi+\psi_i)} \quad u_i = c_{\theta_i} a_{i1} + s_{\theta_i} a_{i2} \\ a_{i2} &= y_p - y_{A_i} - \rho_i s_{\theta_1} + r_i s_{(\phi+\psi_i)} \quad v_i = -\rho_i s_{\theta_i} a_{i1} + \rho_i c_{\theta_i} a_{i2} \\ a_{i3} &= -r_i s_{(\phi+\psi_i)} a_{i1} + r_i c_{(\phi+\psi_i)} a_{i2} \end{aligned} \quad (7)$$

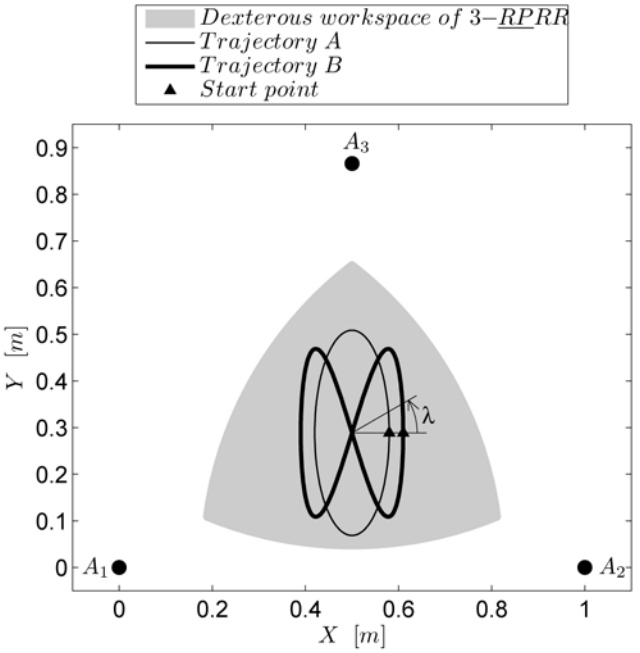


Fig. 2 Selected trajectories for the end-effector.

Here, the objective function for both the PPMP and OMP is defined based on the Normalised Scaled Incircle Radius (NSIR) proposed in [8, 9]. The NSIR is an approach for measuring the proximity to direct and inverse kinematic singularities, using the combination of a geometrical approach and the determinant of the inverse Jacobian matrix over its maximum amount. The measure is normalised as it accounts for the maximum possible values of its different parts resulting on a value between zero and one where one indicates the best possible value and zero the worse (i.e., a singular configuration).

Since for the OMP method the objective function must cover the whole trajectory's NSIR values, the average of the NSIR values of the entire trajectory is used. Also, to ensure no point within the trajectory is too close to a singular configuration, the average NSIR value is multiplied by the minimum NSIR value within the trajectory. As in [8, 9], for the PPMP, the objective function considered here simply consists of the NSIR at each point of the given trajectory.

4 Case Study

To show how the OMP works and to compare it to the PPMP, two numerical examples are given. Figure 2 illustrates the two arbitrary trajectories of the centre of

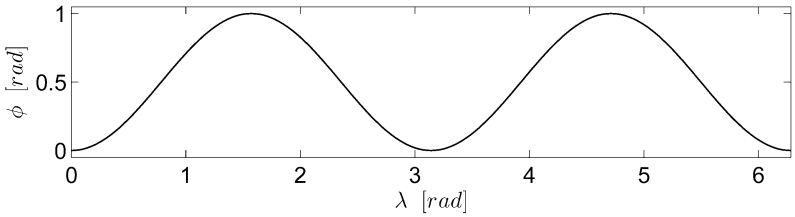


Fig. 3 Orientation of the end-effector throughout trajectory A.

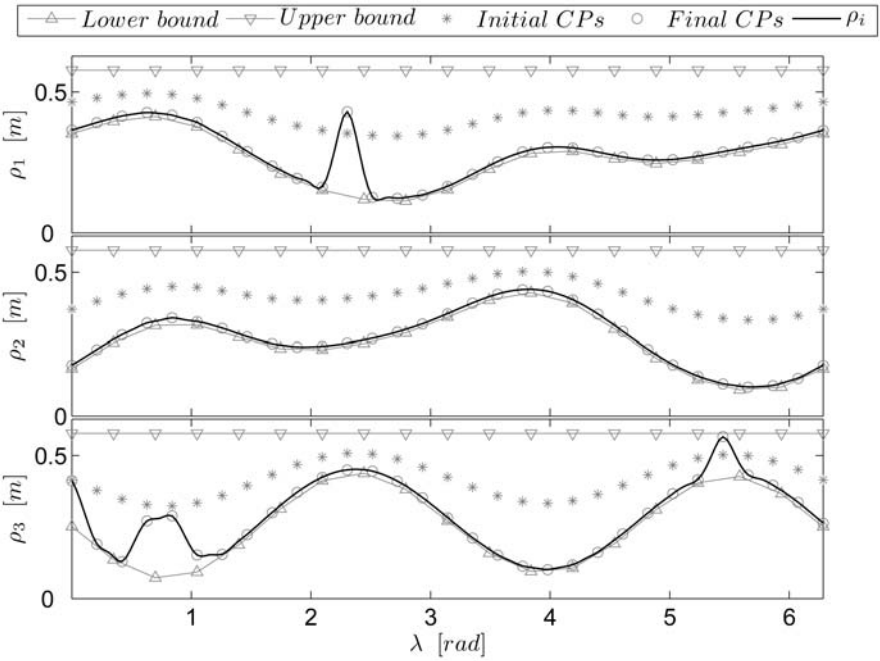


Fig. 4 Actuation schemes for prismatic actuators using OMP (trajectory B).

the 3-RPRR manipulator's end-effector. Also, Figure 3 shows the orientations of the end-effector while tracking trajectory A. For trajectory B, the end-effector orientation is fixed and equal to $\phi = \pi/3$.

For the current numerical examples, the geometric parameters of the 3-RPRR manipulators are (Figure 1): $A_1A_2 = A_2A_3 = A_3A_1 = 1.0$ m, $B_1B_2 = B_2B_3 = B_3B_1 = 0.10$ m, $r_i = 0.0577$ m, $l_i = 0.25$ m, $\psi_1 = 7\pi/6$, $\psi_2 = 11\pi/6$, $\psi_3 = \pi/2$, $\rho_{\max_i} = 0.577$ m and $\rho_{\min_i} = 0$ m.

With the end-effector trajectory defined, the ranges of motion for the prismatic and revolute actuators at each point along the trajectory are obtained from the IDP solution. In this work, the actuation scheme for all three prismatic actuators are used as the search parameters in the optimisation. Figure 4 depicts the ranges of motion of the prismatic actuators for trajectory B. Once the ranges of motion of the

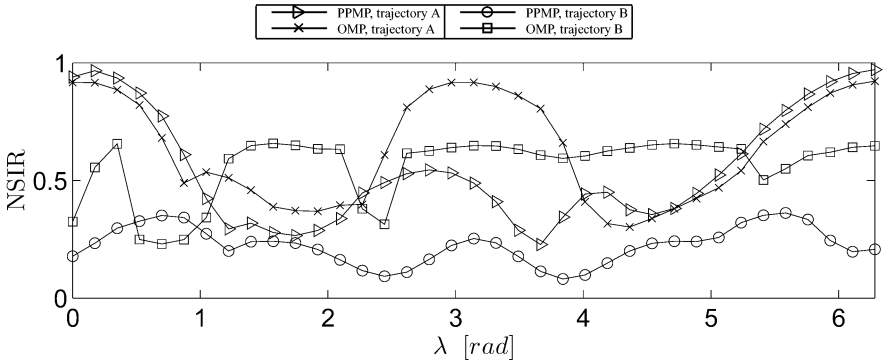


Fig. 5 NSIR values of both PPMP and OMP for the test trajectories.

Table 1 NSIR comparison for trajectories A and B.

	Trajectory A		Trajectory B	
	Ave.	Min.	Ave.	Min.
NSIR values for PPMP	0.543	0.221	0.225	0.080
NSIR values for OMP	0.633	0.302	0.565	0.229

actuators are obtained, it is possible to define the $n+1$ CPs within these bounds. For the OMP, the initial CPs can be selected arbitrarily or can be provided from other motion planning approaches such as the PPMP.

Here, the number of CPs is equal to 31 for both test trajectories thus resulting in 30 straight line segments creating the polyline. All $n+1$ CPs are evenly distributed in λ and their ρ_i values chosen within the range of motion for the corresponding prismatic joint. The arbitrary initial actuation scheme for trajectory B is shown with asterisks in Figure 4. To ensure that the polyline for each actuator is entirely inside the actuator's range of motion, all values throughout the polyline are examined. All values that violate the boundary are forced to the boundary values.

The results are obtained using the NSIR as the objective function within a constrained nonlinear multi-variable optimisation algorithm native to Matlab's Optimisation Toolbox. As an example, Figure 4 illustrates the initial (asterisks) and final results (circles) of the prismatic actuators using the OMP method on trajectory B. Note that, as mentioned earlier, the actuation scheme for the revolute actuators (θ_i) are calculated for every actuation scheme of the prismatic actuators (ρ_i).

Optimal NSIR values for both methods are illustrated in Figure 5. This figure shows that the PPMP results in larger NSIR values than the OMP at some poses. However, as shown in Table 1, the average and the minimum values of the NSIR, when using OMP are significantly larger for both trajectories.

5 Conclusions

A new motion planning approach for kinematically redundant parallel manipulators was proposed. The overall motion planning (OMP) can be used when the whole trajectory is known. The method considers the quality of the trajectory as a whole in the objective function, allowing the manipulator to obtain good characteristics throughout, in this case avoiding singular configurations. The OMP can also take advantage of other approaches' final results such as point-to-point motion planning (PPMP) as the initial values for the OMP. From the point of computational expense, OMP is significantly more demanding than PPMP but results show that the OMP can yield higher quality actuation schemes than the PPMP. Here, a local optimisation algorithm was used for determining the optimal actuation schemes of both PPMP and OMP. In future work, the authors plan to use global optimisation methods as they are likely to improve the results even further.

Acknowledgement

The authors acknowledge the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. Canny, J.F., *The Complexity of Robot Motion Planning*. MIT Press, Cambridge, 1988.
2. Latombe, J.-C., *Robot Motion Planning*. Kluwer Academic Publishers, Norwell, MA, 1991.
3. LaValle, S.M., *Planning Algorithms*. Cambridge University Press, Cambridge, 2006.
4. Merlet, J.-P., A local motion planner for closed-loop robots. In *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, USA, November, pp. 3088–3093, 2007.
5. Yakey, J.H., LaValle, S.M. and Kavraki, L.E., Randomized path planning for linkages with closed kinematic chains. *IEEE Transactions on Robotics and Automation* **17**(6), 951–958, 2001.
6. Cortés, J. and Siméon, T., Probabilistic motion planning for parallel mechanisms. In *Proceedings IEEE International Conference on Robotics and Automation*, Taiwan, Vol. 3, pp. 4354–4359, 2003.
7. Siméon, T., Laumond, J.-P., Cortés, J. and Sahbani, A., Manipulation planning with probabilistic roadmaps. *International Journal of Robotics Research* **23**(7/8), 729–746, 2004.
8. Ebrahimi, I., Carretero, J.A. and Boudreau, R., Path planning for the 3-PRRR redundant planar parallel manipulator. In *Proceedings of the 2007 IFToMM World Congress*, Besançon, France, June, 2007.
9. Ebrahimi, I., Carretero, J.A. and Boudreau, R., Kinematic analysis and path planning of a new kinematically redundant planar parallel manipulator. *Robotica*, 2008 (to appear).
10. Lee, S. and Kim, S., Kinematic analysis of generalized parallel manipulator systems. In *Proceedings of the IEEE Conference on Decision and Control*, Vol. 2, pp. 1097–1102, 1993.

11. Zanganeh, K.E. and Angeles, J., Instantaneous kinematics and design of a novel redundant parallel manipulator. In *Proceedings of IEEE Conference on Robotics and Automation*, San Diego, USA, May, pp. 3043–3048, 1994.
12. Merlet, J.-P., Redundant parallel manipulators. *Laboratory Robotics and Automation* **8**(1), 17–24, 1996.
13. Wang, J. and Gosselin, C.M., Kinematic analysis and design of kinematically redundant parallel mechanisms. *Journal of Mechanical Design* **126**(1), 109–118, 2004.
14. Garg, V., Nokleby, S.B. and Carretero, J.A., Determining the force and moment workspaces of redundantly-actuated spatial parallel manipulators. In *Proceedings of the 2007 ASME DETC*, USA, September 4–7, 2007.
15. Ebrahimi, I., Carretero, J.A. and Boudreau, R., A family of kinematically redundant planar parallel manipulators. *Journal of Mechanical Design*, 2008 (to appear).
16. Tsai, L.-W., *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*. Wiley-Interscience, John Wiley and Sons, New York, 1999.
17. Gosselin, C.M., Lemieux, S. and Merlet, J.-P., A new architecture of planar three-degree-of-freedom parallel manipulator. In *Proceedings 1996 IEEE International Conference on Robotics and Automation*, Vol. 4, pp. 3738–3743, 1996.

Trajectory Planning of Parallel Manipulators for Global Performance Optimization

Ofelia G. Alba-Gómez[1], J. Alfonso Pamanes[1] and Philippe Wenger[2]

[1]*Grupo de Mecatrónica y Control, Instituto Tecnológico de la Laguna (ITLag), Torreon, Coah, CP 27000, Mexico, e-mail: {ofealba, apamanes}@itlalaguna.edu.mx*
[2]*Institut de Recherche en Communications et Cybernétique de Nantes (IRCCyN), 1 Rue de la Noe, 44321 Nantes, France, e-mail: philippe.wenger@irccyn.ec-nantes.fr*

Abstract. The problem of finding optimal configurations for parallel manipulators is investigated in this paper. The task is assumed incompletely specified, namely, the orientation of the tool about its axis is irrelevant to the task. To solve the problem, a procedure is proposed based on the feasibility maps associated with the specified path and an algorithm for trajectory planning into the maps. The obtained motion of the manipulator is such that the condition number of the homogenized parallel Jacobian matrix is optimized along the whole path. The study is illustrated on a planar 3-RRR manipulator. The effectiveness of the proposed approach is demonstrated on an experimental prototype.

Key words: parallel manipulators, trajectory planning, task accessibility, optimization.

1 Introduction

High global stiffness and the ability to achieve high accelerations are interesting features of parallel manipulators. Such properties make this kind of manipulators attractive for applications such as machining operations and material handling. However, the difficulty of planning singularity-free trajectories inside the workspace [1, 4–6] prevents a wide industrial exploitation of this kind of manipulators.

Typically, the trajectory planning problem consists in obtaining a set of joint trajectories such that the accessibility to the whole desired path is guaranteed and the performance of the manipulator is optimized. The condition number of the Jacobian matrix is an interesting index that measures the accuracy of motion of the manipulator and the uniformity of distribution of output velocities and forces in all directions [2].

To assist the trajectory planning problem, the notion of feasibility maps was proposed [3]. The feasibility maps contain all the configurations that a given parallel manipulator can reach to track a prescribed path while avoiding parallel singularities. No attempt was addressed in [3] to generate trajectories into these maps. The present paper introduces an efficient algorithm to generate trajectories into the feas-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 253–261.

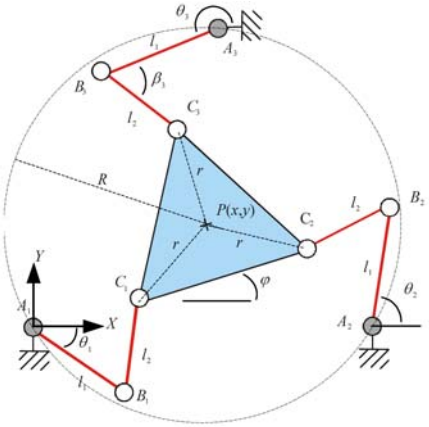


Fig. 1 Symmetric planar 3-RRR parallel manipulator.

ibility maps, which is inspired from a typical algorithm previously used for mobile robot path planning in rough terrains.

2 Preliminaries

The approach proposed in this paper pertains to situations in which a parallel manipulator must follow a prescribed path, the orientation of the tool about its axis being not specified. Such situations occur frequently in process tasks such as machining (think of a hexapod machine used for milling). Without loss of generality, a general 3-DOF planar parallel manipulator with revolute joints is used throughout this paper as illustrative example (Figure 1). This manipulator has been frequently studied [4–10]. It consists of a triangular end-effector $C_1C_2C_3$ (mobile platform) and a fixed base $A_1A_2A_3$, joined by three serial kinematic chains. The manipulator is assumed symmetric, that is, the base and the mobile platform are equilateral triangles and the link lengths are the same for the three legs. The radius of the circle that passes through the vertices of the mobile platform is referred to as r, and the radius of the circle that corresponds to the base triangle is referred to as R. The leg lengths are referred to as l_1 and l_2. The three revolute joints located on the base, are the actuated joint variables $(\theta_1, \theta_2, \theta_3)$.

The output variables are the position coordinates (x, y) of the operational point P chosen as the centroid of the platform, and the orientation φ of the platform. A reference frame is centered at A_1 with the X-axis passing through A_2. The passive and actuated joints are assumed unlimited.

2.1 Velocity Equation and Singularities

The velocity equation of this manipulator can be written in vector form as:

$$\mathbf{A}\mathbf{t} = \mathbf{B}\dot{\boldsymbol{\theta}} \quad \text{with} \quad \mathbf{t} = [\dot{\mathbf{p}} \;\; \dot{\varphi}]^T \quad \text{and} \quad \dot{\boldsymbol{\theta}} = [\dot{\theta}_1 \;\; \dot{\theta}_2 \;\; \dot{\theta}_3]^T, \tag{1}$$

where $\dot{\boldsymbol{\theta}}$ is the vector of actuated joint rates; A is defined as follows [3]:

$$\mathbf{A} = \begin{bmatrix} (\mathbf{c}_1 - \mathbf{b}_1)^T & (\mathbf{c}_1 - \mathbf{b}_1)^T \mathbf{E}(\mathbf{p} - \mathbf{c}_1) \\ (\mathbf{c}_2 - \mathbf{b}_2)^T & (\mathbf{c}_2 - \mathbf{b}_2)^T \mathbf{E}(\mathbf{p} - \mathbf{c}_2) \\ (\mathbf{c}_3 - \mathbf{b}_3)^T & (\mathbf{c}_3 - \mathbf{b}_3)^T \mathbf{E}(\mathbf{p} - \mathbf{c}_3) \end{bmatrix}, \tag{2}$$

where $\mathbf{E}$ represents a counterclockwise rotation of 90 deg.

Matrix $\mathbf{A}$ is singular whenever the axes B_1C_1, B_2C_2 and B_3C_3 intersect or are parallel. This singularity is named *parallel singularity*. When B_1C_1, B_2C_2 and B_3C_3 intersect, the torsional stiffness about the intersection point is zero and the manipulator cannot resist any torque applied at this point. When B_1C_1, B_2C_2 and B_3C_3 are parallel, the manipulator loses stiffness in a direction perpendicular to these axes and the manipulator cannot withdraw any force in this direction. Matrix $\mathbf{B}$ is singular whenever points A_i, B_i, and C_i are aligned for at least one value of i. Unlike the preceding one, this singularity, called *serial singularity*, is not dangerous.

2.2 Working Modes

The working modes are directly related with the matrix $\mathbf{B}$. A working mode is the set of postures for which the sign of elements on the diagonal of $\mathbf{B}$ does not change or some element of the diagonal does not vanish. The working modes separate inverse kinematic solutions. The 3- RRR manipulator can operate in eight distinct working modes [3], each one being associated with an inverse kinematic solution.

2.3 Condition Number

The condition number κ of a matrix $\mathbf{M}$ is defined as

$$\kappa = \|\mathbf{M}\| \|\mathbf{M}^{-1}\|, \tag{3}$$

where $\|\mathbf{M}\|$ denotes any norm of $\mathbf{M}$, such as the maximal singular value of $\mathbf{M}$, which we adopt here. With this norm, the condition number is determined as

$$\kappa = \frac{\sigma_{\mathbf{M}}}{\sigma_{\mathbf{m}}}. \tag{4}$$

When applied to matrix **B**, this definition makes sense because this matrix is dimensionally homogeneous. Matrix **A**, however, is not homogeneous. Thus, it is not possible to meaningfully order its singular values from smallest to largest. Dividing the third column of **A** by a normalizing length yields a dimensionally homogeneous matrix $\bar{\mathbf{A}}$. Daniali et al. [11] proposed to use the characteristic length L. They derived conditions under which a planar 3-DOF parallel manipulator has a configuration where matrix $\bar{\mathbf{A}}$ is isotropic: the two triangles must share a common centroid at the isotropic configuration and the angles between the leg links must be equal. However, their conditions provide $L = \sqrt{2r}|\sin(\alpha)|$, thus yielding several choices for L (α is the angle between B_iC_i and C_iP). A convenient value of the characteristic length was defined in [3] by setting $\alpha = 90$ deg. By doing so, the manipulator is as far as possible from parallel singularities at the isotropic configuration [3].

3 Feasibility Maps

A procedure to generate feasibility maps (FM) was presented in [3]. We are given a path to be followed by a parallel manipulator and this path is incompletely specified, for example only the Cartesian position (x, y) of P is required and the orientation φ of the moving platform needs not be specified. These maps describe the behavior of $\kappa^{-1}(\bar{\mathbf{A}})$ over the whole desired path for all the feasible orientations of the manipulator. An FM defines all the feasible orientations that can be used to accomplish the desired task. Moreover, an FM allows one to select the orientations of the mobile platform associated with the best values of $\kappa^{-1}(\bar{\mathbf{A}})$ during the achievement of the task. Note that an FM can be obtained for each working mode of the manipulator. Therefore, by analyzing all the maps, we can also select the best working mode to be applied.

An example of an FM of the 3-RRR manipulator is shown in Figure 2. The desired path of P (Figure 2a) must be achieved during time T. The FM associated with the first working mode (WM_1) is illustrated in Figure 2b.

4 Searching for a Trajectory

A number of methods have been developed to solve the trajectory planning problem, such as genetic algorithms [12], rapidly-exploring random trees (RRTs) [13, 14], motion roadmaps [15], and dynamic programming by a recursive process [16]. In the problem studied in this paper we have to search for a suitable continuous path into a feasibility map such that the best orientations are selected to accomplish the task.

Consider a prescribed path discretized into n regularly spaced points. We would like to determine the orientation φ of the platform at each point of the path such that $\kappa^{-1}(\bar{\mathbf{A}})$ is a maximum. But, in the same time, the rotation angle of the platform

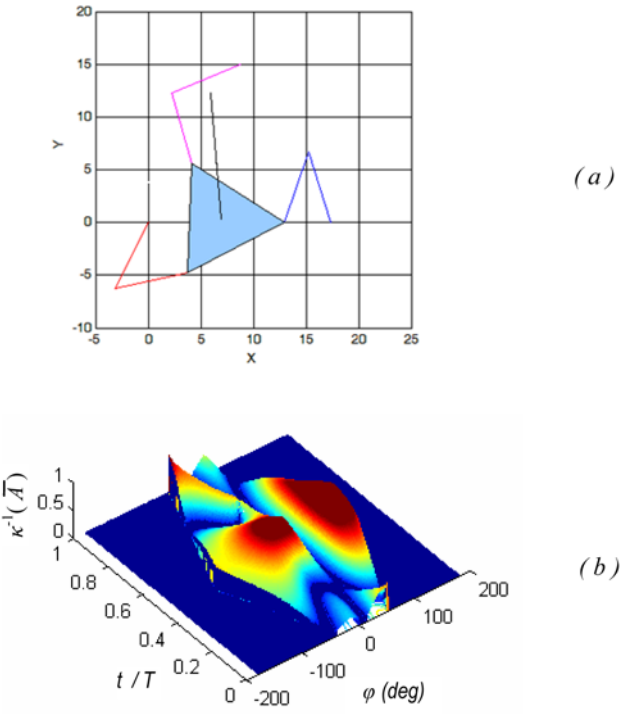


Fig. 2 (a) Manipulator and a path to be followed; (b) feasibility map.

should not exceed a maximal value between two consecutive points along the path to satisfy the actuator constraints. On the other hand, for a safe behavior of the manipulator, index $\kappa^{-1}(\bar{\mathbf{A}})$ should never drop below a given minimal value during motion. Finally, the optimal kinematic inversion along a prescribed path can be regarded as planning an optimal trajectory for a vehicle moving on a complex terrain that includes hills, valleys, and forbidden areas, the vehicle being subject to a minimal turning radius and a maximal power and hence a maximal rate of climb. The surface plot of $\kappa^{-1}(\bar{\mathbf{A}})$ versus non-dimensional time (t/T) and φ, defines a complex terrain where valleys (resp. hills) are associated with regions where $\kappa^{-1}(\bar{\mathbf{A}})$ is high (resp. low) and where forbidden areas are associated with infeasible regions (the manipulator cannot be assembled) or regions where $\kappa^{-1}(\bar{\mathbf{A}})$ is too low. To find an optimal trajectory from a starting point $P_0 = (t_0/T, \varphi_0)$ to a target point $P_f = (t_f/T, \varphi_f)$, we have adapted an algorithm [11] that was developed for a vehicle on a complex terrain with constraints (maximal slope, minimal turn radius and forbidden areas). The adapted algorithm consists in the following steps:

(a) For the first path point, P_0, find an orientation φ_0 of the platform such that $\kappa^{-1}(\bar{\mathbf{A}})$ is greater than a given minimum value $(\kappa^{-1}_{\min})$.

Fig. 3 Experimental parallel manipulator.

1. At the next point P_i search for a value of φ_i subject to the following constraints:
   (1a) $\kappa^{-1}(\bar{\mathbf{A}}_i) \geq \kappa_{\min}^{-1}$;
   (1b) $\kappa^{-1}(\bar{\mathbf{A}}_i)$ is the maximum between those corresponding to orientations of the platform in the interval $\varphi_{i-1} \pm 2°$.
2. If $\kappa^{-1}(\bar{\mathbf{A}}_i)$ at P_i does not satisfy the constraints, then returns to step (a). Otherwise, continue the step 1 with the next path point up to the last point P_f. Save the cost function (f_{cost}) defined as the mean of the values of $\kappa^{-1}(\bar{\mathbf{A}}_i)$ at all the path points.

(b) Return to step (a) until all the possible orientations of the platform at the first point P_0 have been explored. Then compare all the previously saved functions f_{cost} and choose the trajectory having the greatest f_{cost}.

5 Case Study

For validation purposes, our approach was applied on an experimental manipulator 3-RRR. This manipulator has been developed by the research groups of ITLag and IRCCyN. The values of its geometric parameters are: $l_1 = l_2 = 26$ cm, $r = 29$ cm and $R = 36$ cm. The prototype is shown in Figure 3. It was required to guide along the path shown in Figure 4 an axially symmetric tool fixed at the centroid of the mobile platform.

We applied the proposed algorithm to find the orientations to be used by the mobile platform during the task. The prototype was positioned in the first working mode (WM_1). Figure 5a displays the optimal path found in the feasibility map. Figure 5b displays the optimal values of $\kappa^{-1}(\bar{\mathbf{A}})$ along the prescribed path. Finally, Figure 6 shows a sequence of 4 configurations of the manipulator during the achievement of the task.

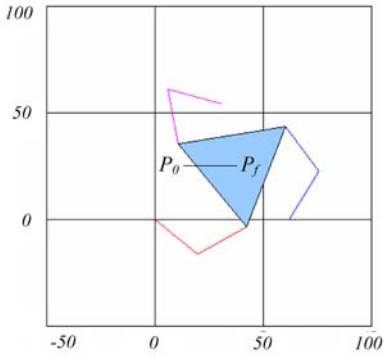


Fig. 4 Desired path, $P_0(13.5, 25)$ cm, $P_f(38, 25)$ cm, for WM_1.

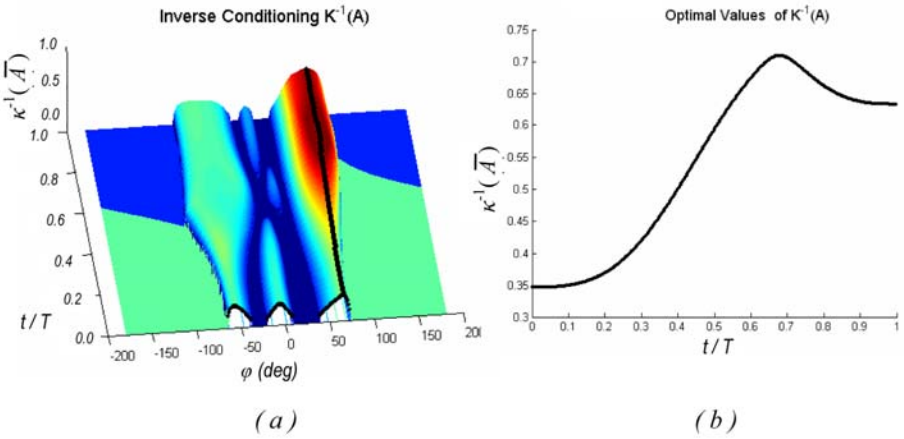


Fig. 5 (a) Optimal trajectory in the feasibility map for WM_1. (b) Optimal values of $\kappa^{-1}(\bar{\mathbf{A}})$ along the prescribed path.

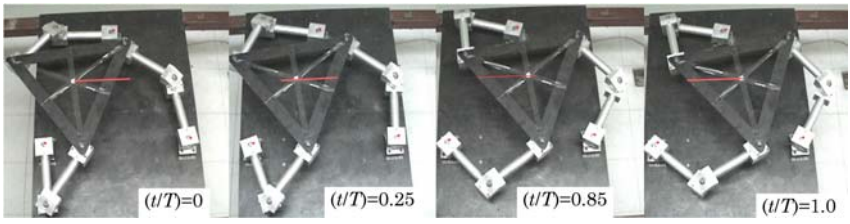


Fig. 6 Sample of manipulator configurations along the desired trajectory (WM_1).

6 Conclusion

An algorithm was presented in this paper for planning trajectories of a parallel manipulator along a prescribed path where the orientation of the tool about its axis

needs not be specified. The search of an optimal trajectory was carried out into the feasibility map of the manipulator associated with one working mode. This map makes it possible to identify the orientations of the mobile platform that define the best values of a kinetostatic index $\kappa^{-1}(\bar{\mathbf{A}})$ during the achievement of the task. To find an optimal trajectory in the feasibility map, we have adapted a path planning algorithm that was proposed in the past for a vehicle moving on a complex terrain with constraints. The resulted motion of the manipulator is such that $\kappa^{-1}(\bar{\mathbf{A}})$ is as high as possible along the specified path and parallel singularities are avoided. Consequently, the kinetostatic performance of the manipulator is globally optimized during the task. The algorithm was successfully tested on a 3-RRR manipulator prototype built at ITLag.

In the example shown in the paper, the manipulator could track the whole path in a single working mode. In some cases, however, no solutions may be found when the manipulator operates in a single working mode. This could arises when the prescribed path is long; in such a case a parallel singularity is more likely to be encountered. A possible solution is then to allow the robot to change its working mode during motion. Simulation results have shown that it is possible [17] but they need to be experimented on the ITLag's prototype for validation. To the authors' knowledge, such a global approach has not been proposed in the past.

The approach proposed in this paper is not limited to planar manipulators. It is applicable for hexapod-based parallel kinematic machines or other parallel machines. This requires, however, to redefine the kinetostatic index $\kappa^{-1}(\bar{\mathbf{A}})$.

Acknowledgments

This work has been conducted by ITLag and IRCCyN as part of a project in the Franco-Mexican Laboratory of Applied Automatics (LAFMAA), which is supported by CNRS and CONACYT. The project was also partially supported by COSNET and DGEST of Mexico.

References

1. Gosselin, C. and Angeles, J., Singularity analysis of closed-loop kinematic chains, *IEEE Trans. on Robotics and Automation* **6**(3), 281–290, 1990.
2. Salisbury, J.K. and Craig, J.J., Articulated hands: Force and kinematic issues, *Int. J. of Robotic Research* **1**(1), 4–17, 1982.
3. Alba-Gomez, O., Wenger, P. and Pamanes, G.J.A., Consistent kinetostatic indices for planar 3-DOF parallel manipulators, application to the optimal kinematic inversion, in *Proceedings of the 2005 ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Long Beach USA, September, Paper DETC2005-84326, 2005.
4. Gosselin, C. and Angeles, J., Kinematic inversion of parallel manipulators in the presence of incompletely specified tasks, *ASME J. Mechanical Design* **112**, 494–500, 1990.

5. Chablat, D., Wenger, P. and Angeles, J., Working modes and aspects in fully-parallel manipulators, in *Proceedings IEEE International Conference on Robotics and Automation*, pp. 1964–1969, 1998.
6. Arsenault, M. and Boudreau, R., The synthesis of three-degree-of-freedom planar parallel mechanisms with revolute joints (3-RRR) for an optimal, singularity-free workspace, *Journal of Robotic Systems* **21**(5), 259–274, 2004.
7. Gosselin, C. and Angeles, J., The optimum kinematic design of a planar three-degree-of-freedom parallel manipulator, *ASME J. Mech. Transmissions Automation Design* **110**, 1988.
8. Bonev, I.A., Zlatanov, D. and Gosselin, C.M., Singularity analysis of 3-DOF planar parallel mechanisms via screw theory, *ASME J. Mechanical Design* **125**(3), 573–581, 2003.
9. Pennock, G.R. and Kassner, D.J., The workspace of a general geometry planar three-degree-of-freedom platform-type manipulator, *ASME J. Mechanical Design* **115**, 269–276, 1993.
10. Gosselin, C. and Angeles, J., Singularity analysis of closed-loop kinematic chains, *IEEE Trans. on Robotics and Automation* **6**(3), 281–290, 1990.
11. Daniali, H., Zsombor-Murray, P. and Angeles, J., The isotropic design of two general classes of planar parallel manipulator, *J. Robotic Systems* **12**(12), 785–805, 1995.
12. Hu, Y., A knowledge based genetic algorithm for path planning of a mobile robot, in *Proceedings of the 2004 IEEE International Conference on Robotics & Automation*, New Orleans, LA, April, Vol. 5, pp. 4350–4355, 2004.
13. Strandberg, M., Augmenting RRT-planners with local trees, in *Proceedings of the 2004 IEEE International Conference on Robotics & Automation*, New Orleans, LA, April 2004.
14. Lindemman, S., Incrementally reducing dispersion by increasing Voronoi bias in RRTs, in *Proceedings of the 2004 IEEE International Conference on Robotics & Automation*, New Orleans, LA, April 2004.
15. Kim, J., Extracting optimal paths from roadmaps for motion planning, in *Proceedings of the 2003 IEEE International Conference on Robotics & Automation*, Taipei, Taiwan, September 2003.
16. Lamiraux, F., Kinodynamic motion planning: Connecting exploration trees using trajectory optimization methods, in *Proceedings of the 2004 IEEE International Conference on Robotics & Automation*, New Orleans, LA, April 2004.
17. Alba-Gómez, O., Pamanes, J.A. and Wenger, P., Trajectory planning of a redundant parallel manipulator changing of working mode, in *12th World Congress on Theory of Machines and Mechanisms*, Besançon, France, June 2007.

Human Motion Reconstruction by Direct Control of Marker Trajectories

Emel Demircan, Luis Sentis, Vincent De Sapio and Oussama Khatib

Artificial Intelligence Laboratory, Stanford University, Stanford, CA 94305, U.S.A., e-mail: {emeld, lsentis, vdesap, khatib}@stanford.edu

Abstract. Understanding the basis of human movement and reproducing it in robotic environments is a compelling challenge that has engaged a multidisciplinary audience. In addressing this challenge, an important initial step involves reconstructing motion from experimental motion capture data. To this end we propose a new algorithm to reconstruct human motion from motion capture data through direct control of captured marker trajectories. This algorithm is based on a task/posture decomposition and prioritized control approach. This approach ensures smooth tracking of desired marker trajectories as well as the extraction of joint angles in real-time without the need for inverse kinematics. It also provides flexibility over traditional inverse kinematic approaches. Our algorithm was validated on a sequence of tai chi motions. The results demonstrate the efficacy of the direct marker control approach for motion reconstruction from experimental marker data.

Key words: human motion synthesis, operational space formulation, task/posture decomposition, prioritization, marker space.

1 Introduction

The central nervous system (CNS) is able to elegantly coordinate the complex structure of the human body to perform movements of great agility and sophistication. An effective way of understanding human movement involves mimicking motions which are optimal in performance. Such optimal movements include those exhibited by highly skilled practitioners in sports and the martial arts. Human motor performance depends on skilled motor coordination and posture control as well as physical strength and perception. Drawing inspiration from their biological counterparts humanoid robots are being imbued with skilled dynamic behaviors. Enhancing the authenticity of synthesized human motion in robotic systems has been a continuing challenge that draws together researchers from the fields of physiology, neuroscience, biomechanics, and robotics. This challenge has been addressed by researchers through retargetting methods [1, 10]. Additionally, adapting existing motion for a human character with a given set of constraints has been explored [4, 9]. However,

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 263–272.

these techniques, which map motion-capture data to different characters and scenes, require inverse kinematic computations.

We propose a new algorithm to reconstruct human movement through direct control of optical marker trajectory data. This approach directly projects marker points onto a simulated human model and tracks the trajectory in Cartesian space. Tracking the desired trajectories is accomplished using the operational space control framework [5]. With our algorithm we can drive a simulated model of the human body to experimental marker locations in real-time. This allows smooth tracking of the desired marker trajectories and extraction of joint angles within a reasonable error boundary. Further, the task/posture decomposition used in the operational space method constitutes a natural decomposition for dealing with marker data, thus avoiding the performance of inverse kinematics.

The algorithm presented here is validated through a sequence of slow tai chi motions. Tai chi motions are light, lively, and balanced, and they constitute a rich variety of motions useful for testing purposes. With our new algorithm we show that human movement can be controlled and reconstructed in real-time. This facilitates the investigation of other high performance dynamic skills.

2 Direct Marker Control Approach

Optical motion capture constitutes a common and effective method for capturing human motion. A series of markers attached to a subject's body are imaged by a set of cameras and the spatial positions of the markers are triangulated from the image data. A number of post processing steps need to be performed to convert the raw marker positions into useful kinematic data. The most significant step is to convert the marker trajectories into joint space trajectories. This has commonly been done using inverse kinematic techniques.

As an alternative to performing inverse kinematics on marker data we propose to dynamically track the markers using a task-level control approach. We will refer to this approach as direct marker control. For the purposes of our direct marker control application, we will define task space as the space of Cartesian coordinates for the motion capture markers. However, it must be noted that marker trajectories obtained through motion capture are not independent. For example, markers on the same body link are rigidly constrained to each other and the relative motion between markers on adjacent links is limited by the freedom in the connecting joints. To accommodate for motion dependencies the markers are grouped into independent subsets, $\{m_1, \ldots, m_n\}$. Each subset, m_i, is represented by a single task vector, x_{m_i}, that is formed by concatenating the Cartesian coordinates of the individual markers contained within that subset. Using a prioritized control approach a hierarchy of marker task vectors is formed where the tasks that are lower in the hierarchy are projected into the null space of the tasks that are higher in the hierarchy. At the end of this recursive process, independent subsets of marker tasks are obtained, ensuring

the overall feasibility of the marker control. The operational space formulation [5] is then used to directly control the marker trajectories.

2.1 Prioritized Control in Marker Space

In this section, we develop the proposed framework for direct control of marker trajectories. The need to address high performance dynamic behaviors and the need for flexibility over traditional inverse kinematic approaches constitute the motivation for our approach.

A behavioral task generally involves descriptions of various parts of the multi-body mechanism, each represented by an operational point $x_{t(i)}$. The full task is represented as an $m \times 1$ vector, x_t, formed by vertically concatenating the coordinates of all operational points. The Jacobian associated with this task is denoted as J_t. The derivation of the operational space formulation begins with the joint space dynamics of the robot [8]

$$A\ddot{q} + b + g = \Gamma \tag{1}$$

where q is the the vector of n generalized coordinates of the articulated system, A is the $n \times n$ kinetic energy matrix, b is the vector of centrifugal and Coriolis generalized forces, g is the vector of gravity forces, and Γ is the vector of generalized control forces.

Task dynamic behavior is obtained by projecting (1) into the space associated with the task, which can be done with the following operation

$$\overline{J}_t^T [A\ddot{q} + b + g = \Gamma] \Longrightarrow \Lambda_t \ddot{x}_t + \mu_t + p_t = \overline{J}_t^T \Gamma \tag{2}$$

Here, $\overline{J}_t^T$ is the dynamically-consistent generalized inverse of J_t [8], Λ_t is the $m \times m$ kinetic energy matrix associated with the task and μ_t and p_t are the associated centrifugal/Coriolis and gravity force vectors.

The operational space framework [5] is used as the basis for our direct marker control algorithm. In this formulation, the task behavior is divided into a set of independent task points and the torque component for the task is determined in a manner that compensates for the dynamics in task space. For a task behavior, x_t, with decoupled dynamics and unit inertial properties $\ddot{x}_t = F_t^*$, this torque is given by the force transformation

$$\Gamma_{\text{task}} = J_t^T F_t \tag{3}$$

where J_t is the Jacobian of the task and F_t is the operational space force. This operational space force is given by

$$F_t = \Lambda_t F_t^* + \mu_t + p_t \tag{4}$$

A task/posture decomposition allows us to represent the dynamics of a simulated human subject in a relevant task space that is complemented by a posture space. The

total control torque is decomposed into two dynamically decoupled torque vectors: the torque corresponding to the commanded task behavior and the torque that only affects posture behavior in the null space of the task

$$\Gamma = \Gamma_{\text{task}} + \Gamma_{\text{posture}} = J_t^T F_t + N_t^T \Gamma_p \tag{5}$$

In this expression N_t^T is the null space projection matrix, and Γ_p is the torque projected into the null space.

The prioritized control framework [7, 11] is used to control the collection of marker task vectors. In this framework the torque decomposition is:

$$\Gamma = J_t^T F_t + N_t^T (J_p^T F_p) \tag{6}$$

where the posture torque can be rewritten as

$$\Gamma_{\text{posture}} = (J_p N_t)^T F_p = J_{p|t}^T F_{p|t} \tag{7}$$

Consequently, Equation (5) can be represented as:

$$\Gamma = J_t^T F_t + J_{p|t}^T F_{p|t} \tag{8}$$

Alternately, if an additional task is projected into the posture we express this as

$$\Gamma = J_{t_1}^T F_{t_1} + J_{t_2|t_1}^T F_{t_2|t_1} \tag{9}$$

This is generalized for an arbitrary number of additional tasks

$$\Gamma = J_{t_1}^T F_{t_1} + J_{t_2|t_1}^T F_{t_2|t_1} + \cdots + J_{t_n|t_{n-1}|\cdots|t_1}^T F_{t_n|t_{n-1}|\cdots|t_1} \tag{10}$$

2.2 Direct Marker Control Formulation

For the application to marker space we will use m_i to denote the task for a particular marker subset. Equation (10) then becomes:

$$\Gamma = J_{m_1}^T F_{m_1} + J_{m_2|m_1}^T F_{m_2|m_1} + \cdots + J_{m_n|m_{n-1}|\cdots|m_1}^T F_{m_n|m_{n-1}|\cdots|m_1} \tag{11}$$

The Jacobian and the force associated with marker space are deduced from the above equation as follows:

$$J_\otimes \triangleq \begin{bmatrix} J_{m_1} \\ J_{m_2|m_1} \\ \vdots \\ J_{m_n|m_{n-1}|\cdots|m_1} \end{bmatrix} \quad \text{and} \quad F_\otimes \triangleq \begin{bmatrix} F_{m_1} \\ F_{m_2|m_1} \\ \vdots \\ F_{m_n|m_{n-1}|\cdots|m_1} \end{bmatrix} \tag{12}$$

The overall control torque is then

$$\Gamma = J_{\otimes}^{T} F_{\otimes} \tag{13}$$

An analysis on the bounds of the joint space errors can be performed using the Jacobian associated with the marker space. We note

$$\Delta x_{\otimes} = J_{\otimes} \Delta q \tag{14}$$

The inverse of this relationship is

$$\Delta q = \bar{J}_{\otimes} \Delta x_{\otimes} \tag{15}$$

where $\bar{J}_{\otimes}$ is the dynamically-consistent generalized inverse of $J_{\otimes}$. Joint angles obtained through prioritized control in marker space deviate from the actual values but are bounded by:

$$|\Delta q| \leq |\bar{J}_{\otimes}||\Delta x_{\otimes}| \tag{16}$$

This allows us to tune the prioritized marker controller to accommodate the desired accuracy, for given configurations.

3 Simulation and Validation

A series of real-time movements performed by a tai chi master were recently captured using an optical marker system to provide validation for our new algorithm. The subject was a fifty-five year old male of average build. An eight-camera retro-reflective motion capture system was used to capture his movements at a rate of 60 Hz.

Among the motions performed, a sequence of slow movements were chosen for validation and real-time simulation. Marker data of the recorded motion were then segmented and smoothed using the EvART software (Motion Analysis Corporation).

Marker trajectories were imported into the SAI environment [6] which allows dynamic 3D simulations. Our existing human model which consists of 25 joints, was scaled to match the anthropometry of the tai chi master. The data used in this model have been derived from SIMM models [2]. The skeleton has been modeled as a multibody system within SAI and scaled based on body segment mass-center locations [3]. Figure 1 depicts the scaled human model simulated in the SAI environment.

The human motion reconstruction described in Section 2 was executed using crucial upper body joint marker trajectories (e.g. shoulder, elbow and wrist of both right and left arms). Sets of two decoupled markers were grouped into distinct marker subsets. The first subset consisted of the right shoulder and the left wrist markers and the second set consisted of the left elbow and the right wrist markers. The markers were then directly tracked through the entire movement sequence using prioritization.

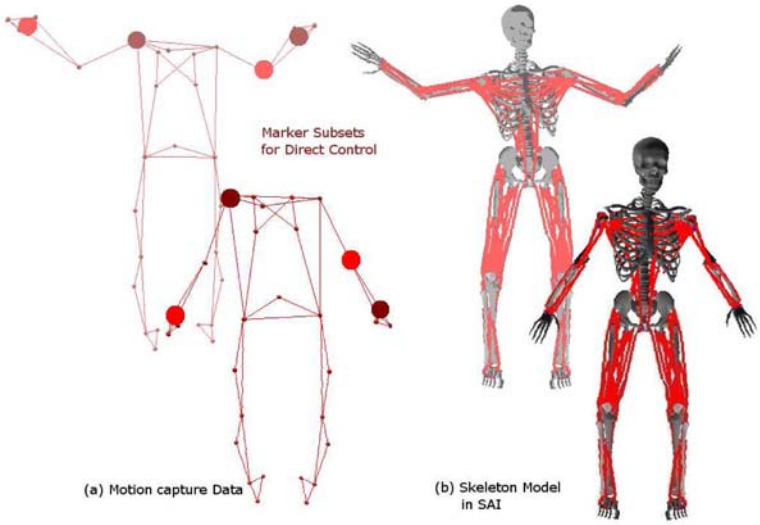


Fig. 1 The scaled human model of the tai chi master simulated in the SAI environment. Markers of the right shoulder and the left wrist are selected to form the first marker set to be controlled (dark spheres). The second subset is formed by the left elbow and the right wrist markers (light spheres).

Joint angles over the entire trajectory were directly obtained as a natural consequence of the direct marker control approach, thus avoiding any need for computing inverse kinematics. Figure 2 illustrates the joint angles obtained through direct marker tracking.

The commanded and tracked positions of the controlled markers, as well as the joint angles, were recorded during real-time simulation. Figure 3 shows the comparison between tracked and commanded positions of each tracked marker. The consistency between the two curves in each plot suggests the efficacy of the human motion reconstruction algorithm proposed.

Bounds on the joint angle errors can be addressed using Equation (15). We can compute the maximum and minimum joint angle error bounds among all the joints. Thus

$$\Delta q_{\max} = \max(\Delta q) \quad \text{and} \quad \Delta q_{\min} = \min(\Delta q) \tag{17}$$

where $\Delta q = \bar{J}_{\otimes} \Delta x_{\otimes}$. Figure 4 shows the margin of marker position errors and the margin of joint angle errors respectively. Maximum and minimum joint angle error magnitudes vary stably over the trajectory, suggesting well bounded errors on the joint angles.

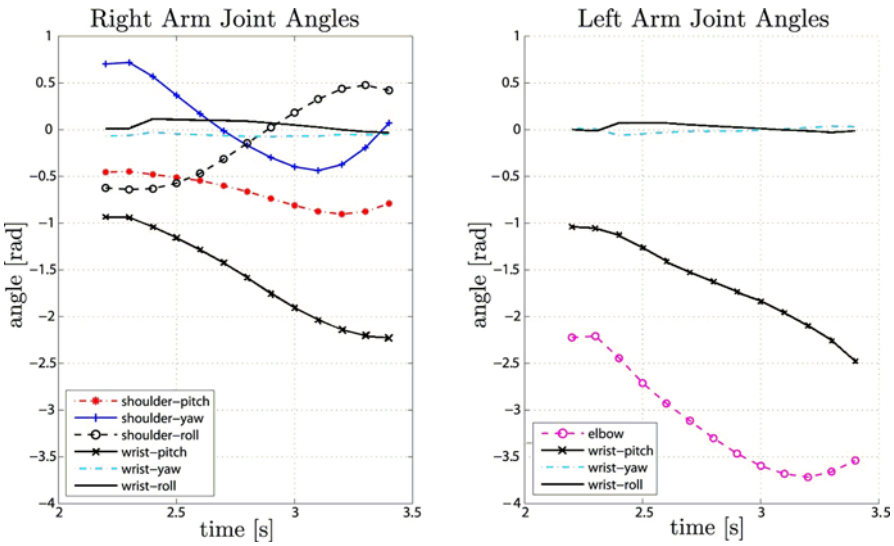


Fig. 2 Joint angles obtained through direct control of marker data. The joint angles for the shoulder, elbow and wrist segments are shown for the corresponding arm. Smooth joint space trajectories are generated without inverse kinematics.

4 Conclusions

In this paper, we presented a new algorithm to control marker trajectories based on the operational space method of Khatib [5], and task/posture decomposition using prioritization. The algorithm reconstructs human motion in real-time through smooth tracking of marker trajectories. This facilitates the extraction of joint angles without the need for inverse kinematic calculations. The algorithm also easily accommodates anthropometric scaling of the simulation model to the human subject.

We validated our new algorithm through a set of tai chi movement data. The results illustrate smooth tracking of the marker trajectories in marker space. Smooth joint angles trajectories were obtained as a natural output of the marker tracking methodology. A bound on the joint space error was obtained and the results of this analysis indicated stable error bounds over the trajectory. The errors can further be decreased with a more precise camera calibration during motion capture experiment and a more accurate model scaling of the simulation.

There are plans to extend our algorithm to address task/posture decomposition in the camera space of the optical motion capture system. This may exploit further advantages of our approach by accommodating arbitrary operational spaces. Additionally, where precise knowledge of the subject anthropometry is not known a priori, our approach can be adapted so that limb lengths can be adjusted until they optimally track the marker data. This provides a way of inferring more precise anthropometry through direct tracking of marker data. By extending our new algorithm

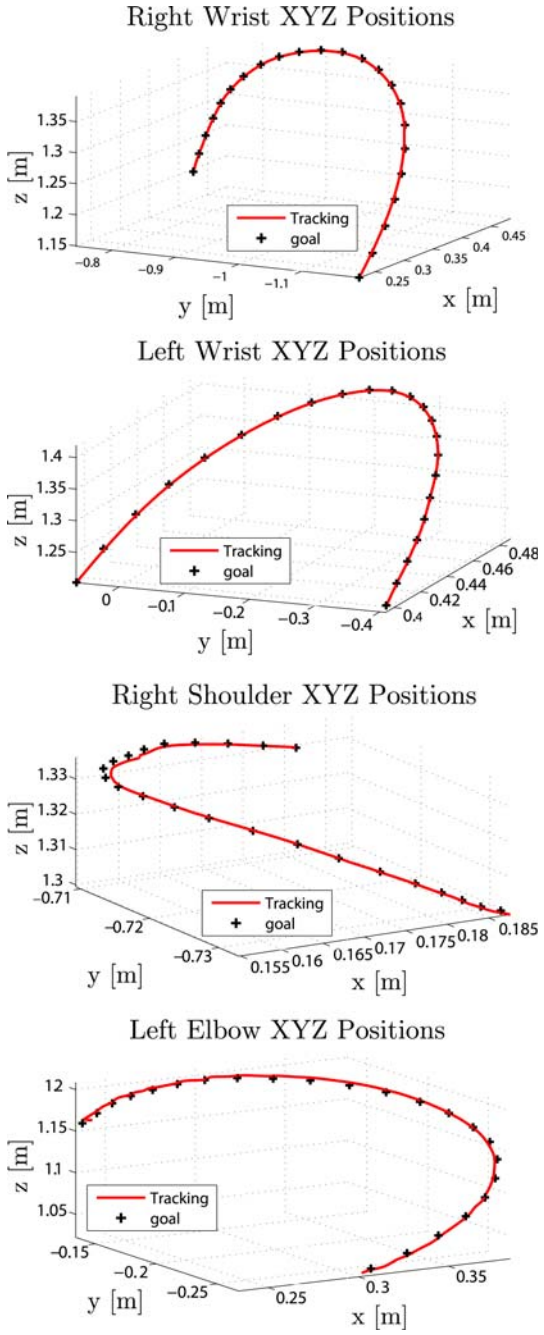


Fig. 3 Tracked and goal trajectories of markers. The tracked trajectories (solid lines) are shown for markers attached to the wrist, shoulder, and elbow segments. It can be seen that the generated trajectories closely track the corresponding goal trajectories (dotted lines).

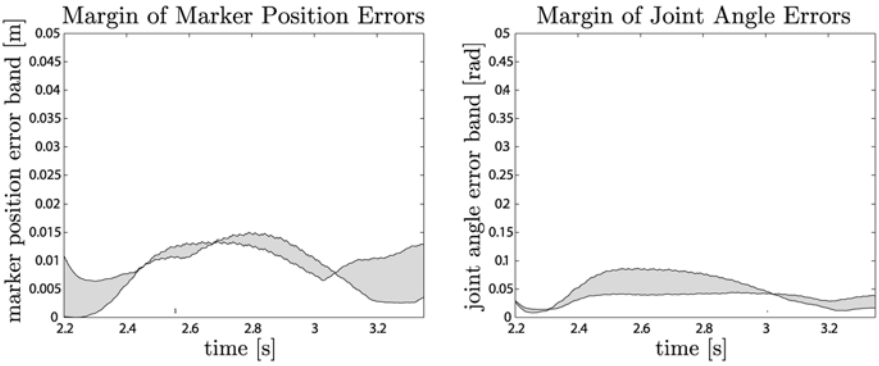


Fig. 4 Margin of marker position errors and margin of joint angle errors over the trajectory. Joint angle error magnitudes show a stable variation over the trajectory, thus ensuring well bounded errors on the joint angles.

to incorporate other operational spaces, we hope to synthesize and investigate motor control models for skilled human movement using a task-level framework.

Acknowledgements

The contributions of Dr. Jessica Rose, Director of Lucile Packard Children's Hospital Motion and Gait Analysis Laboratory, and the financial support of the Simbios National Center for Biomedical Computing Grant (http://simbios.stanford.edu/, NIH GM072970) are gratefully acknowledged.

References

1. Choi, K. and Ko, H. (1999), On-line motion retargetting, in *Proceedings Seventh Pacific Conference on Computer Graphics and Applications*, IEEE Computer Society, pp. 32–42.
2. Delp, S. and Loan, P. (1995), A software system to develop and analyse models of musculoskeletal structures, *Computers in Biology and Medicine* **25**, 21–34.
3. Dempster, W.T., (1955), Space requirements of the seated operator, WADC Technical Report 55-159.
4. Grochow, K., Martin, S.L., Hertzmann, A. and Popovic, Z. (2004), Style-based inverse kinematics, in *ACM Transactions on Graphics (TOG), Proceedings of the 2004 SIGGRAPH Conference* **23**(3), 522–531.
5. Khatib, O. (1987), A unified approach for motion and force control of robot manipulators: The operational space formulation, *International Journal of Robotics Research* **3**(1), 43–53.
6. Khatib, O., Brock, O., Chang, K., Conti, F., Ruspini, D. and Sentis, L. (2002), Robotics and interactive simulation, *Communications of the ACM* **45**(3), 46–51.
7. Khatib, O., Sentis, L., Park, J. and Warren, J. (2004), Whole-body dynamic behavior and control of human-like robots, *International Journal of Humanoid Robotics* **1**(1), 29–43.

8. Khatib, O. (2004), *Advanced Robotic Manipulation*, Class Notes, Stanford University, Stanford, USA.
9. Lee, J. and Shin, S.Y. (1999), A hierarchical approach to interactive motion editing for human-like figures, in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, ACM Press/Addison Wesley Publishing Co., pp. 39–48.
10. Savenko, A. and Clapworthy, G. (2002), Using motion analysis techniques for motion retargetting, in *Proceedings Sixth International Conference on Information Visualization (IV'02)*, IEEE Computer Society Press, p. 110.
11. Sentis, L. and Khatib, O. (2005), Synthesis of whole-body behaviors through hierarchical control of behavioral primitives, *International Journal of Humanoid Robotics* II-v4.

PERFORMANCE AND PROPERTIES OF MECHANISMS

New Self-Motions of Parallel Manipulators

Adolf Karger

Faculty of Mathematics and Physics, Charles University Praha, Czech Republic, e-mail: Adolf.Karger@mff.cuni.cz

Abstract. In this paper we present two results on geometry of parallel manipulators with planar platform and base (Stewart–Gough type platforms). In the first one we show that for this type of manipulators there is always one quadratical equation on Euler parameters between equations describing the geometry of the platform. This equation has a strong geometrical meaning – it is identically zero iff the platform is architecturally singular. This yields a very simple system of equations for architectural singularity and proves automatically the self-mobility. In the second result we describe a large class of manipulators with self-motions, a special case of which was already discussed in Borel (1908) and Bricard (1906).

Key words: Stewart–Gough platforms, architectural singularity, self-motions.

1 Introduction

Self-motion of a Stewart–Gough platform is an important phenomena in geometry of parallel manipulators – it yields an uncontrolable motion of the manipulator. It is a motion of the manipulator with locked length of legs of the manipulator and therefore it is at the same time a Borel–Bricard motion, a space motion with six spherical trajectories. In this paper we shall concentrate at manipulators with planar platform and base. We describe a new class of self-motions for them.

To describe the geometry of such a manipulator, we shall use the parametrization of a spatial motion by using Euler and Study parameters (see Botema and Roth, 1990; Husty, 1996). Any space motion is a curve at the group of spatial motions. We use the Study representation of the displacement group of the Euclidean space in the projective space P_7 with homogeneous coordinates $x_0, \ldots, x_3, y_0, \ldots, y_3$.

Let S be the set in P_7 determined by

$$U = x_0y_0 + x_1y_1 + x_2y_2 + x_3y_3 = 0,$$

where $x_0^2 + x_1^2 + x_2^2 + x_3^2 \neq 0$, the so-called Study quadric.

We can normalize coordinates in S by

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 275–282.

$$\kappa \equiv x_0^2 + x_1^2 + x_2^2 + x_3^2 = 1.$$

In this way a space motion is represented by a curve on S.

The manipulator consists of two parts – platform and base. To describe its motion, we have to describe the relative position of these two parts. The upper part of the platform lies in the moving space, the lower is in the fixed space. The upper part of the platform will be described by using lower case letters, for the base we use capital letters. We choose coordinate systems in such a way, that the third coordinate of points of the platform and base is equal to zero and therefore it will be omitted.

The platform consists of six points $m_i = [a_i, b_i]$, $i = 1, \ldots, 6$, similarly the base is determined by points $M_i = [A_i, B_i]$, $i = 1, \ldots, 6$.

The connection between a point $m = [a, b]$ of the platform and point $M = [A, B]$ of the base is given by a homogeneous quadratic equation in Study parameters as follows:

$$\begin{aligned} h = {} & R\kappa + 4(y_0^2 + y_1^2 + y_2^2 + y_3^2) + 2(x_3^2 - x_0^2)(Aa + Bb) + 2(x_2^2 - x_1^2)(Aa - Bb) \\ & + 4[x_0x_3(Ab - Ba) - x_1x_2(Ab + Ba)(x_0y_1 - y_0x_1)(A - a) \\ & + (x_0y_2 - y_0x_2)(B - b) - (x_1y_3 - y_1x_3)(B + b) \\ & + (x_2y_3 - y_2x_3)(A + a)] = 0, \end{aligned} \tag{1}$$

where $R = A^2 + B^2 + a^2 + b^2 - r^2$, and r is the distance of points M and m. The procedure to obtain this equation is in detail described in several papers (see for instance Husty, 1996) and we leave it out. Substitution of coordinates of points M_i and m_i for $i = 1, \ldots, 6$ yields six equations $h_1 = 0, \ldots, h_6 = 0$, which completely describe the mutual position of the two parts of the manipulator.

2 Movability of Architecture Singular Platforms

To solve the direct problem for such a manipulator we have to solve equations $h_i = 0$, $i = 1, \ldots, 6$, $U = 0$, $\kappa = 1$, with given geometry of the platform and given distances of points m_i and M_i, $i = 1, \ldots, 6$. We obtain eight quadratical equations for eight unknown study parameters x_j, y_j, $j = 0, 1, 2, 3$. This problem has in general finite many solutions, self-motion appears if the problem has infinite many solutions, all hyper-quadrics $h_i = 0$, $i = 1, \ldots, 6$, $U = 0$ have a curve in common. To solve this problem looks to be very difficult and not many self-motions are known.

Equations $h_i = 0$, $i = 1, \ldots, 6$, $U = 0$ are "almost" linear in y_j, $j = 0, \ldots, 3$ and therefore there is a chance to find a linear combination of these equation in such a way that y_j will dissappear.

Let $q = k_1h_1 + k_2h_2 + \cdots + k_6h_6$ be such a linear combination with coefficients k_i, $i = 1, \ldots, 6$. We shall look for condition for k_i such that q does not contain y_j. We must have $k_1 + k_2 + \cdots + k_6 = 0$, because of quadratical terms. Simple look at

the structure of the equation $h = 0$ shows that in total we obtain equations

$$\Sigma k_i = 0, \quad \Sigma k_i A_i = 0, \quad \Sigma k_i B_i = 0, \quad \Sigma k_i a_i = 0, \quad \Sigma k_i b_i = 0, \qquad (2)$$

with summation over i from 1 to 6.

As a result we obtain five homogeneous linear equations for coefficients k_i, $i = 1, \ldots, 6$. From linear algebra we know that such a system always has a nontrivial solution. This means that there is always such a linear combination of equations $h_i = 0$ without y_j.

We can show that the following is true.

Theorem. Let the system (2) have maximal rank. Then the unique quadratical equation $q = 0$ constituted from $h_i = 0$ disappers iff the manipulator is architecturally singular.

For the definition of architectural singularity, see Ma and Angeles (1992), Merlet (1992) and Karger (2003).

Corollary. This means that architecture singular manipulators have always continuous movability.

It follows from the fact that in general the solution of equations $h_i = 0, i = 1, \ldots, 6$, $U = 0$ with respect to y_j, $j = 0, \ldots, 3$ leads to three algebraic homogeneous equations in x_j, $j = 0, \ldots, 3$ of degrees two, four and eight. If the quadratical equation disappears, the spherical image of the motion is given as the intersection of two algebraic surfaces and therefore it is a curve. This curve must of course be understood in algebraic sence, it can be imaginary or consist only of several isolated points.

Proof of the Theorem. Equation $q = 0$ is identically equal to zero iff all its coefficients are equal to zero. There are essentially five coefficients in this equation, as wee see from its form. It is easy to see that

$$\begin{aligned}&\mu_1(x_0^2 + x_1^2 + x_2^2 + x_3^2) + \mu_2(-x_0^2 + x_3^2) + \mu_3(-x_1^2 + x_2^2)\\ &\quad = (\mu_1 - \mu_2)x_0^2 + (\mu_1 - \mu_3)x_1^2 + (\mu_2 + \mu_3)x_2^2 + (\mu_1 + \mu_2)x_3^2 \equiv 0\end{aligned}$$

iff $\mu_1 = \mu_2 = \mu_3 = 0$.

For this reason we obtain five equations:

$$\Sigma k_i R_i = 0, \quad \Sigma k_i A_i a_i = 0, \quad \Sigma k_i A_i b_i = 0, \quad \Sigma k_i B_i a_i = 0, \quad \Sigma k_i B_i b_i = 0, \qquad (3)$$

with k_i given by (2). These equations are equivalent with equations for architectural singularity for platforms with planar platform and base, they are rather complicated and we do not give them here. Their equivalent version is given in Karger (2003, equations 4), including cases with four collinear points, see also Karger (2008).

Remark 1. Both mentioned systems of equations are rather complicated, in (3) after (2) is solved each of them has 720 terms in general. The equivalence is to be understood that both systems have the same set of relevant solutions, they also have trivial solutions, which can be different.

Remark 2. The procedure used above does not work for general type of parallel manipulators (with nonplanar platform or base), because it is not possible to find a linear combination of equations $h_i = 0, i = 1, \ldots, 6$ which does not contain y_j, $j = 0, 1, 2, 3$. Therefore also the structure of architecture singular non-planar parallel manipulators is entirely different from the one described here (see Karger, 2008). On the other side, the problem of movability is much easier, it is possible to prove it directly for almost all cases.

3 A Class of Self-motions

In this section we shall describe self-motions of a rather general class of Stewart–Gough type platforms. This class is a generalization of the self-motion of the original platform, proposed by Stewart, we shall similarly as in Husty and Karger (1998) look for self-motions described by a planar curve lying in the plane $x_0 = 0$ in Euler parameters.

For symmetry reasons we shall not use the quadratic equation $q = 0$ described above, but use the standard way. We denote $U_i = h_1 - h_i$, $i = 2, \ldots, 6$, to solve equations $U_2 = 0, U_3 = 0, U_4 = 0, U = 0$ for y_j, $j = 0, \ldots, 3$ and substitute the result into U_5.

We notice that U_6 is obtained from U_5 by replacing the index 5 by 6. This fact will play an important role in what follows.

The computation with general values of coordinates of points M_i and m_i leads to complicated formulas. For this reason we have to specify systems of coordinates in the moving and fixed spaces. If we do not want to loose the generality of computations, we have to admitt only such changes of frames which will preserve the equation $x_0 = 0$. This means that we can use all translations in both spaces. For this reason we can use points m_1 and M_1 as origins of coordinates in both spaces.

Now we look at the quaternionic representation of the orientation group. The number 1 as a quaternion represents the direction orthogonal to the plane $x_0 = 0$. This shows that the group preserving the plane $x_0 = 0$ is at the same time the group preserving the number 1.

As the action of the orientation group in moving and fixed spaces on quaternion α is given by the formula $\alpha_1 = g_1 \alpha \bar{g}_2$, with g_1, g_2 unit quaternions, we see that the number 1 is preserved iff $g_1 = g_2$. Therefore we can use one copy of the orientation group for coordinate changes, in fixed or moving space. This means that we can suppose for instance that the plane of the platform is the plane with third coordinate zero. Rotation around the third axis remains.

We shall consider only a special case, we suppose that the plane of the base is also the plane with third coordinate zero (it seems that the general case gives no solution anyway). This means that we can rotate coordinates in the plane of the platform. Therefore we can without loss of generality suppose $A_1 = B_1 = 0, a_1 = 0, b_1 = 0$, $b_2 = 0$. Using this simplification, the equation $U_5 = 0$ still has 762 terms, but the general computation is possible. We obtain

$$U_5 = x_1^2(s_1x_1 + s_3x_2) + x_2^2(s_4x_1 + s_2x_2) + (s_5x_1 + s_6x_2)x_3^2. \tag{4}$$

We would like to solve the equation $U_5 = 0$ identically and then by substitution of index 6 for 5 we shall have also $U_6 = 0$, a self motion given by $h_1 = 0$. To obtain $U_5 \equiv 0$, we have to solve equations $s_i = 0$, $i = 1, \ldots, 6$. Denote

$$\bar{s}_3 = s_3 - s_2, \quad \bar{s}_4 = s_4 - s_1, \quad \bar{s}_5 = s_5 - s_1, \quad \bar{s}_6 = s_6 - s_2.$$

Equations $\bar{s}_3 = 0, \ldots, \bar{s}_6 = 0$ do not contain R_i,$i = 1, \ldots, 6$ and they are linear in A_5, B_5 and in a_5, b_5.

$\bar{s}_3$ and $\bar{s}_4$ have 132 terms, $\bar{s}_5$ 42 and $\bar{s}_6$ only 24. We solve last two equations for A_5, B_5 and substitute the result into $\bar{s}_3$ and $\bar{s}_4$. Let us now suppose that points m_i, M_i for $i = 1, 2, 3, 4$ are given. Equations $\bar{s}_3$, $\bar{s}_4$ are two cubic equations in a_5, b_5. They present two cubic curves in the plane of coordinates a_5, b_5. They have in general nine points in common. Inspection shows that one point is at infinity, four points are points m_i, $i = 1, 2, 3, 4$, two of the points do not have corresponding A_5, B_5.

The remaining two points we take for points m_5, m_6 and we obtain a self-motion. Inspection of the result shows that it is a special case of an architecture singular platform, with self-motion with planar spherical image. This is interesting, but not what we are looking for.

So let us look for the special case where these two cubic curves coincide. This happens iff equations $\bar{s}_3$ and $\bar{s}_4$ are proportional. We obtain only one condition, it is a cubic expression in coordinates of points m_i, M_i. If this condition is satisfied, we have a whole cubic of points in the plane of the platform which have spherical trajectories and the manipulator has a self-motion. As it is not possible to present the general computation, we shall give two examples.

Example 1. We choose

$$M_1 = [0, 0], \quad M_2 = [3, 0], \quad M_3 = [1, 5], \quad M_4 = [0, 3],$$

$$m_1 = [0, 0], \quad m_2 = [2, 0], \quad m_3 = [3, 2], \quad m_4 = [1, 3]$$

and proceed with the computation. We obtain

$$A_5 = (-24a_5 - b_5 + 18a_5b_5 - 3b_5^2)/((3b_5 - 4)(7a_5 - 10)), \quad B_5 = 5b_5/(3b_5 - 4).$$

Cubic curves are

$$\begin{aligned}\bar{s}_3 &= 63b_5^3 - 1080b_5^2a_5 - 4044b_5a_5^2 \\ &\quad + 1479b_5^2 - 6372a_5^2 + 7818b_5a_5 - 114b_5 + 12744a_5, \\ \bar{s}_4 &= 135b_5^3 - 4584b_5^2a_5 - 1386b_5a_5^2 \\ &\quad + 5403b_5^2 - 252a_5^2 + 15636b_5a_5 - 18114b_5 + 504a_5.\end{aligned}$$

Resultant of $\bar{s}_3$ and $\bar{s}_4$ with respect to b_5 is equal to

$$a_5(a_5-1)(a_5-2)(a_5-3)(7a_5-10)^2(277a_5^2-1179a_5+820).$$

From this equation we see clearly that cubics $\bar{s}_3 = 0$ and $\bar{s}_4 = 0$ have their nine common points distributed as follows:

a) one point is at infinity,
b) points m_1, m_2, m_3, m_4,
c) two points do not have corresponding points in the base,
d) two points remain, we take them for points m_5, m_6.

The result is

$$a_{5,6} = [1179 \pm 13\sqrt{(2849)}]/554, \quad b_{5,6} = [1614 \pm 40\sqrt{(2849)}]/277,$$

$$A_{5,6} = [-1435 \pm 41\sqrt{(2849)}]/812, \quad B_{5,6} = [735 \pm 8\sqrt{(2849)}]/203,$$

where the sign plus is for fifth point, minus is for the sixth point. Computation of the Jacobian shows that it is equal to zero. This shows that we have an architecture singular.

Example 2. Now we shall describe a manipulator with self-mobility which is not architecture singular. We choose the same points $m_1, m_2, m_3, m_4, M_1, M_2, M_3$ as in Example 1. We leave the point M_4 general and specify it later to satisfy the condition of proportionality of equations $\bar{s}_3$ and $\bar{s}_4$.

Denote $T = \bar{s}_3 - \lambda\bar{s}_4$, where λ is the unknown proportionality factor. T is a cubic expression in a_5, b_5, the coefficient at b_5^3 yields $\lambda = 7(B_4 + 2A_4)/[5(2B_4 - 15)]$. Substitution of this value into T shows that T factors into two parts, one of them is a cubic expression T_1 in coordinates A_4, B_4 of the point M_4 of the form

$T_1 = 98A_4^2(8B_4 - 165) + T_2$, where T_2 is linear in A_4. (The other factor is only a combination of both cubics, it is not interesting.)

Any point of $T_1 = 0$ can be chosen for point M_4, to simplify computations we choose $B_4 = 165/8$ and T_1 becomes linear in A_4 and we have $A_4 = -21155/1872$.

Equation $\bar{s}_3$ simplifies to

$$\bar{s}_3 = 2707b_5^3 + b_5^2(3354a_5 + 42462) + b_5(-400a_5^2 + 14255a_5 - 171916) + 11250a_5(a_5 - 2).$$

This equation yields a cubic with the property that any six points of this cubic can be chosen as points $m_1, \ldots, m_6$ and together with corresponding points $M_1, \ldots, M_6$ we obtain a parallel manipulator with self-motion. The correspondence between points $m = [x, y]$ and $M = [X, Y]$ of the platform and base is given by the rule

$$X = (16842xy + 40605y - 60150x - 5414y^2)/[4(7y - 25)(301 + 50x)],$$

$$Y = -55y/[2(7y - 25)].$$

During the self motion all points of the cubic run on spherical trajectories with centers again on a cubic. The motion itself is represented by a planar curve of degree

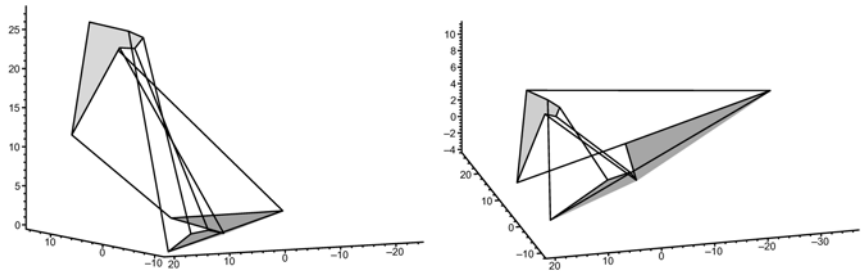


Fig. 1 Two cases of the platform from Example 2.

six in Euler parameters, it does not make sense to present its equation. It depends on three parameters R_1, R_2, R_3 and it has 93 terms. Special choice of these parameters shows that the motion can be real. In Figure 1 two pairs on the cubic have been chosen, we have obtained manipulators with self-motion.

Remarks. The general condition for geometry of the manipulator which leads to self-motions is very complicated algebraic condition for points $m_i, M_i, i = 1, \ldots, 4$ (it is the general counterpart of the equation T_1 for the special case of Example 2, it has about 1000 terms). It would be interesting to find special cases for which the condition has geometrical interpretation. An inspiring case is considered in papers of Borel (1908) and Bricard (1906). They consider so-called homological configurations.

A configuration of points $m_1, \ldots, m_4$ and $M_1, \ldots, M_4$ in plane is called homological if there is a point O such that points O, m_1, M_1, m_3, M_3 and O, m_2, M_2, m_4, M_4 are on straight lines. In our denotation we take O for origin, $m_i = [a_i, b_i]$, $M_i = [A_i, B_i]$. We suppose $b_1 = 0, B_1 = 0$. The configuration is homological if $b_3 = B_3 = 0, b_2 = ka_2, B_2 = kA_2, b_4 = ka_4, B_4 = kA_4$ for some k. If for such configuration $A_1 + a_1 = A_3 + a_3$, the necessary condition for self-motion is automatically satisfied and we have always self-motion.

Another case is the original Stewart–Gough platform. For special geometry of this manipulator we obtain a self-motion with a whole cubic of points running on spherical trajectories. One such case is given in Figure 2, where we present a Stewart–Gough platform with self-motion.

Conclusion

Beside of the proof of self-mobility of architecture singular platforms with both platform and base planar we have found a large class of self-motions of such platforms.

The condition of self-mobility is a large algebraic equation in coordinates of points m_i, M_i, $i = 1, \ldots 4$. Special solutions can be found, one of them is the

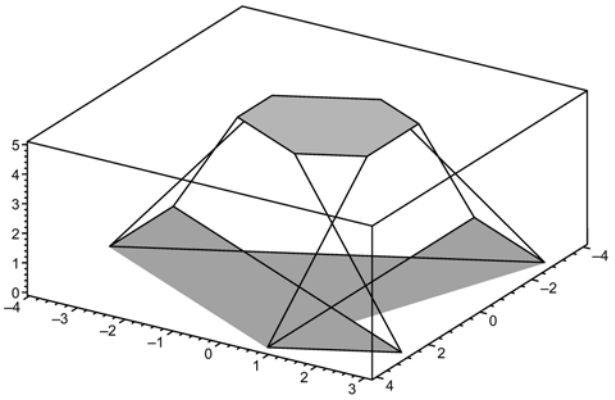


Fig. 2 Example of a Stewart–Gough platform with self-motion.

case considered by Borel and Bricard, another one is the original Stewart–Gough platform. Other special cases still have to be discovered.

Acknowledgement

The research reported here was enabled by Grant VZ 0021620839 of the Ministery of Sports and Education of the Czech Republic.

References

Borel E. (1908) Mémoire sur les déplacements à trajectoires sphérigues. *Mém. présentés par divers savants, Paris*, **2**(33), 1–128.

Bottema, O., Roth, B. (1990) *Theoretical Kinematics*, Dover Publishing, 1990.

Bricard, R. (1906) Mémoire sur les déplacements à trajectoires sphérigues. *Journ. École Polyt.* **2**(11), 1–96.

Husty, M.L. (1996) An algorithm for solving the direct kinematics of general Stewart–Gough platforms. *Mechanism and Machine Theory* **31**, 365–380.

Husty, M., Karger, A. (1998) Classification of all self-motions of the original Stewart–Gough platform. *Computer-Aided Design* **30**(3), 205–215.

Karger, A. (2003) Architecture singular planar parallel manipulators. *Mechanism and Machine Theory* **43**(3), 335–346.

Karger, A. (2008) Architecture singular non-planar parallel manipulators. *Mechanism and Machine Theory* **38**, 1149–1164.

Ma, O., Angeles, J. (1992) Architecture singularities of parallel manipulators. *International Journal of Robotics and Automation* **7**, 23–29.

Merlet, J.P. (1992) Singular configurations of parallel manipulators and Grassmann geometry. *International Journal of Robotics Research* **8**, 45–56.

Exact Envelope Computation for Moving Surfaces with Quadratic Support Functions

Margot Rabl[1], Bert Jüttler[1] and Laureano Gonzalez-Vega[2]

[1]*Institute of Applied Geometry, Johannes Kepler University, 4040 Linz, Austria, e-mail: {margot.oberneder, bert.juettler}@jku.at*
[2]*Departamenta de Matemáticas, Estadística y Computación, Universidad de Cantabria, 39005 Santander, Spain, e-mail: laureano.gonzalez@unican.es*

Abstract. We consider surfaces whose support function is obtained by restricting a quadratic polynomial to the unit sphere. If such a surface is subject to a rigid body motion, then the Gauss image of the characteristic curves is shown to be a spherical quartic curve, making them accessible to exact geometric computation. In particular we analyze the case of moving surfaces of revolution.

Key words: envelope, characteristic curve, support function, parameterization.

1 Introduction

Envelopes of moving surfaces are needed for computing the swept volume which is traced out by a moving solid. They can be obtained by collecting the characteristic curves, where the moving surface touches its envelope at a given time. In Robotics, these computations are related to the problem of collision detection. Other applications include the numerical simulation of milling processes, where the tool can be modeled as a surface of revolution.

Abdel-Malek et al. [1] gave a detailed survey about swept volume computation with many related references. Peternell et al. [5] computed the boundary of the swept volume generated by a general moving object, which is assumed to be given as a triangular mesh. Special attention is paid to the choice of the time-step for computing the characteristic curves.

Envelopes of certain specific classes of surfaces have been analyzed in more detail. Flaquer et al. [4] studied envelopes of moving quadric surfaces, in particular moving planes, spheres, cylinders and cones. In all these cases, the characteristic curves are algebraic space curves of degree 4. Xia and Ge [8] considered cylinders as an example for milling tools and generated an exact representation of the boundary surface.

We discuss the case of surfaces which are specified by their support functions. These surfaces can explicitly be parameterized by its inverse Gauss maps and we use this observation to characterize the characteristic curves. The case of surfaces with quadratic support functions is discussed in detail, since they allow the exact

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 283–290.

computation of the characteristic curves. General support functions can be approximated by surfaces with piecewise quadratic ones, which are defined over a given spherical triangulation. This should make it possible to extend the results of this paper to more general objects.

2 Support Functions

We recall the support function representation of surfaces (see e.g. [6]). Consider a given function $h \in C^\infty(\mathbb{S}^2, \mathbb{R})$, where $\mathbb{S}^2$ denotes the unit sphere in $\mathbb{R}^3$. We use this function to associate with each point $\mathbf{n} \in \mathbb{S}^2$ the plane with the unit normal $\mathbf{n}$ and oriented distance $h(\mathbf{n})$ to the origin.

The envelope of the two-parameter family of planes obtained by varying $\mathbf{n}$ in $\mathbb{S}^2$ describe a surface and the given function h is called the *support function* of this surface. For any $h \in C^\infty(\mathbb{S}^2, \mathbb{R})$, a parameterization $\mathbf{x}_h \in C^\infty(\mathbb{S}^2, \mathbb{R})$ of the surface is given by its inverse Gauss map,

$$\mathbf{x}_h(\mathbf{n}) = h(\mathbf{n})\mathbf{n} + (\nabla_{\mathbb{S}^2} h)(\mathbf{n}), \tag{1}$$

where $(\nabla_{\mathbb{S}^2} h)$ is the intrinsic gradient of the support function h with respect to the unit sphere $\mathbb{S}^2$. If the support function h is obtained by restricting a suitable function $h^0 \in C^\infty(\mathbb{R}^3, \mathbb{R})$ to the unit sphere $\mathbb{S}^2$, then

$$(\nabla_{\mathbb{S}^2} h)(\mathbf{n}) = (\nabla h^0)(\mathbf{n}) - [(\nabla h^0)(\mathbf{n}) \cdot \mathbf{n}]\mathbf{n}. \tag{2}$$

This parameterization, whose domain is the unit sphere, can now be composed with any parameterization of $\mathbb{S}^2$, e.g., by spherical coordinates.

In this paper we are particularly interested in the case where the support function is the restriction of a trivariate quadratic polynomial to $\mathbb{S}^2$. We call the corresponding envelopes *quadratically supported surfaces* (QSS). The class of QSS is closed under translations, offsetting and rotations, as these geometric operations correspond to the addition of constants and homogeneous linear polynomials, and to the composition with rotations, respectively.

Example 1. Let $\mathbf{n} = (x, y, z)^\top$ and consider the two support functions

$$h(\mathbf{n}) = x^2 + y^2 + \frac{3}{2}z^2 \quad \text{and} \quad h(\mathbf{n}) = x^2 + y^2 - z^2. \tag{3}$$

The associated QSS have the parametric representations $\mathbf{x}_h(\mathbf{n}) =$

$$\begin{pmatrix} -x^3 - xy^2 - \frac{3}{2}xz^2 + 2x \\ -yx^2 - y^3 - \frac{3}{2}yz^2 + 2y \\ -zx^2 - zy^2 - \frac{3}{2}z^3 + 3z \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} -x^3 - xy^2 + xz^2 + 2x \\ -yx^2 - y^3 + yz^2 + 2y \\ -zx^2 - zy^2 + z^3 - 2z \end{pmatrix}, \tag{4}$$

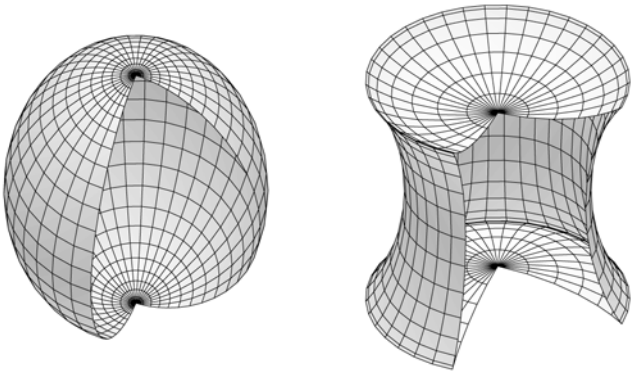


Fig. 1 Quadratically supported surfaces (QSS).

respectively. Both support functions describe surfaces of revolution, shown in Figure 1, and the profile curve of the second one is a special trochoid.

3 Motions, Velocities, Characteristics

As usual, we describe a rigid body motion by a time-dependent transformation

$$\mathbf{x}' = \mathbf{t}(\alpha) + U(\alpha)\mathbf{x} \tag{5}$$

between world coordinates $\mathbf{x}'$ and moving coordinates $\mathbf{x}$, where the parameter α represents the time, the vector $\mathbf{t}(\alpha)$ represents the translation of the origin, and the special orthogonal matrix $U(\alpha)$ specifies the rotation. For an arbitrary but constant value of α, we compute the velocity vector $\mathbf{v}'$ of a fixed point $\mathbf{x}$ in the moving system

$$\mathbf{v}' = \dot{\mathbf{t}} + \dot{U}\mathbf{x}, \tag{6}$$

where the dot indicates differentiation with respect to α and the argument α has been omitted. This velocity is transformed into the moving system

$$\mathbf{v} = U^\top \mathbf{v}' = U^\top \dot{\mathbf{t}} + U^\top \dot{U}\mathbf{x} = U^\top \dot{\mathbf{t}} + \omega \times \mathbf{x}, \tag{7}$$

where ω denotes the angular velocity.

We consider a surface in the moving space, which is assumed to be given by its support function representation. The surface touches the envelope along the *characteristic curve*. Let

$$\mathbf{v}_h(\mathbf{n}) = U^\top \dot{\mathbf{t}} + \omega \times \mathbf{x}_h(\mathbf{n}) \tag{8}$$

be the velocity of the point $\mathbf{x}_h(\mathbf{n})$, then the characteristic curve consists of all points where the inner product of the velocity and the surface normal $\mathbf{n}$ vanishes,

$$\mathbf{v}_h(\mathbf{n}) \cdot \mathbf{n} = 0. \tag{9}$$

In the case of a moving QSS, the Gauss image of the characteristic curve (i.e., the spherical curve obtained by collecting the surface normals along it) is particularly simple:

Theorem 1. *The Gauss image of the non-degenerate characteristic curve of a moving QSS is a spherical quartic.*

Proof. After substituting Eqns. (1), (2) and (8) into Eq. (9), one gets after a short computation

$$\Big(U^\top \dot{\mathbf{t}} + \omega' \times (\nabla h)(\mathbf{n})\Big) \cdot \mathbf{n} + (h(\mathbf{n}) - (\nabla h)(\mathbf{n}) \cdot \mathbf{n}) \underbrace{(\omega \times \mathbf{n}) \cdot \mathbf{n}}_{=0} = 0. \tag{10}$$

If h is a trivariate polynomial of degree 2, then this equation is of degree 2. The Gauss image of the characteristic curve consists of all surface normals that satisfy Eq. (10) and $\mathbf{n} \cdot \mathbf{n} = 1$. Consequently, it is either the intersection of two quadric surfaces, or it degenerates into the entire unit sphere. □

Summing up, the Gauss image of the characteristic curve is the zero set of the two quadratic polynomials

$$f(\mathbf{n}) = \Big(U^\top \dot{\mathbf{v}} + \omega' \times (\nabla h)(\mathbf{n})\Big) \cdot \mathbf{n} \quad \text{and} \quad g(\mathbf{n}) = \mathbf{n}^\top \cdot \mathbf{n} - 1. \tag{11}$$

The envelope surface can be generated by collecting all characteristic curves obtained for different values of α and transforming them into world coordinates.

4 Characteristic Curves for QSS of Revolution

In this section we compute a parameterization of the characteristic curve for a fixed time α. Its Gauss image is the intersection curve of the two quadrics defined by the quadratic polynomials in Eq. (11).

Dupont et al. [3] described a sophisticated algorithm for the computation of a near-optimal parameterization of the intersection curve of two quadric surfaces. This algorithm assumes that the coefficients of the two quadratic equations belong to the field $\mathbb{Q}$ of rational numbers. The intersection curve is parameterized with the help of square-root functions of certain polynomials, which belong to the ring of polynomials over a special field extension of $\mathbb{Q}$. In the most general case, the computation of the exact parameterization requires the solution of a quartic equation, and the corresponding field extension.

In the remainder of this paper we restrict ourselves to QSS which are surfaces of revolution with respect to the z-axis. The support function then takes the form

$$h_R(\mathbf{n}) = a(x^2 + y^2) + bz^2 + cx + dy + ez + f. \tag{12}$$

We assume that the coefficients a, b, c, d, e, f are in the field $\mathbb{Q}$ of rational numbers, and that the components of the angular velocity ω and the translational velocity $\dot{\mathbf{t}}$ are also from this field.

Lemma 1. *There exists a point* $\mathbf{P}$ *on the Gauss image of the characteristic curve with coordinates in the field extension* $\mathbb{Q}(\sqrt{r})$*, where* r *is an integer.*

Proof. We consider an arbitrary but fixed rational number z_0. If we substitute $z = z_0$ in Eq. (11), then this equation becomes linear in the remaining variables x and y. Indeed, quadratic terms in Eq. (11) may be present only in $[\omega' \times (\nabla h_R)(\mathbf{n})] \cdot \mathbf{n} = [(\nabla h_R)(\mathbf{n}) \times \mathbf{n}] \cdot \omega'$, and a short computation confirms that

$$(\nabla h_R)(\mathbf{n}) \times \mathbf{n} = \begin{pmatrix} 2(a-b)yz_0 - ey + dz_0 \\ 2(b-a)xz_0 + ex - cz_0 \\ cy - dx \end{pmatrix}. \tag{13}$$

Consequently, the points on the Gauss image of the characteristic curve with $z = z_0$ can be found by solving a single quadratic equation. Since the Gauss image of the characteristic curve is always non-empty, it is possible to choose z_0 such that real solutions exist. □

Based on this result we compute a parameterization of the Gauss image of the characteristic curve by the Enhanced Levin's method of [7], which is summarized below:

1. Find a real point $\mathbf{P}$ on the intersection curve $f(\mathbf{n}) = g(\mathbf{n}) = 0$. This point serves as the center of the stereographic projection into the xy-plane (another plane may be used).
2. Let $\mathbf{Q} = (\xi, \eta, 0)^\top$. The image of the spherical quartic under the stereographic projection is the planar cubic curve defined by the cubic polynomial

$$c(\xi, \eta) = \text{Resultant}\left(\frac{1}{t} f(t\mathbf{Q} + (1-t)\mathbf{P}), \frac{1}{t} g(t\mathbf{Q} + (1-t)\mathbf{P}\,;\; t\right). \tag{14}$$

3. Find a square-root parameterization of $c(\xi, \eta) = 0$ and project it back onto the unit sphere.

The first part of the last step will be explained in more detail. First we find a point $\mathbf{R}$ on the cubic curve. For instance, it can be chosen as the intersection point of the curve tangent at $\mathbf{P}$ with the xy-plane. In this case, the coordinates of $\mathbf{R}$ are again in $\mathbb{Q}(\sqrt{r})$. We then consider the pencil of lines

$$\begin{pmatrix} \xi(s,t) \\ \eta(s,t) \end{pmatrix} = t \begin{pmatrix} 1 \\ s \end{pmatrix} + (1-t)\mathbf{R},\ (s,t) \in \mathbb{R}, \tag{15}$$

and substitute into the cubic polynomial $c(\xi, \eta) = 0$. After factoring out the trivial solution $t = 0$, we solve the resulting equation, which is quadratic in t, for t and get a solution of the form

$$t(s) = \frac{k(s) \pm \sqrt{\ell(s)}}{m(s)}, \tag{16}$$

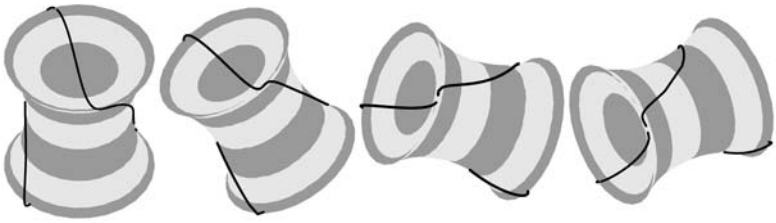


Fig. 2 Characteristic curves on a moving non-convex QSS of revolution.

where the three polynomials $k(s)$, $\ell(s)$ and $m(s)$, which possess the degrees 2, 4 and 3, respectively, belong to the ring of polynomials over the field extension $\mathbb{Q}(\sqrt{r})$.

The projection of the cubic back onto the sphere, and the computation of the characteristic curve by substituting the result into Eq. (1) involves only rational operations. We summarize the results of this section.

Theorem 2. *The characteristic curve of a QSS of revolution, where the coefficients of the support functions and the components of velocity and angular velocity are rational numbers, possesses a parameterization*

$$\left(r_1(s, \sqrt{\ell(s)}), r_2(s, \sqrt{\ell(s)}), r_3(s, \sqrt{\ell(s)})\right)^\top. \tag{17}$$

The rational functions $r_i : \mathbb{R}^2 \to \mathbb{R}$, $i = 1, 2, 3$, and the quartic polynomial $\ell(s)$ have coefficients in the field extension $\mathbb{Q}(\sqrt{r})$, where r is an integer.

5 Examples

Example 2. We consider a motion with $\dot{\mathbf{t}}(\alpha) = (1, 1, 0)^\top$ and

$$U = \frac{1}{9}\begin{pmatrix} 5\cos\alpha + 4 & 2 - 2\cos\alpha - 6\sin\alpha & 4 - 4\cos\alpha + 3\sin\alpha \\ 2 - 2\cos\alpha + 6\sin\alpha & 8\cos\alpha + 1 & 2 - 2\cos\alpha - 6\sin\alpha \\ 4 - 4\cos\alpha - 3\sin\alpha & 2 - 2\cos\alpha + 6\sin\alpha & 5\cos\alpha + 4 \end{pmatrix} \tag{18}$$

and apply it to the non-convex QSS of revolution from the second section. Figure 2 shows several positions of the moving non-convex surface and the corresponding characteristic curves.

In order to obtain expressions with rational coefficients, we substitute both $\sin\alpha = 2\beta/1 + \beta^2$ and $\cos\alpha = 1 - \beta^2/1 + \beta^2$ in $U^\top$, while $U^\top\dot{U}$ is a constant matrix, as U describes a uniform rotation with axis direction $(2, 1, 2)^\top$. As an example, we consider $\beta = 2$, i.e., $\alpha \approx 0.927$, where the two quadratic equations, which define the Gauss image, are

$$f = 15x - 20xz - 9y + 40yz + 12z \quad \text{and} \quad g = x^2 + y^2 + z^2 - 1. \tag{19}$$

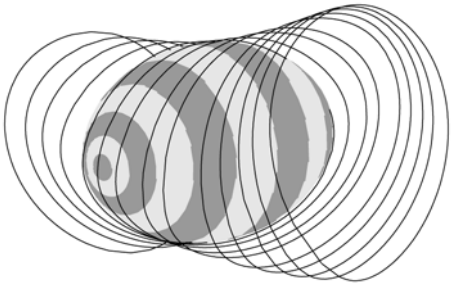


Fig. 3 Several characteristic curves of a moving convex QSS of revolution, forming the envelope surface.

A possible center of projection is found after choosing $z_0 = \frac{1}{2}$,

$$\mathbf{P} = \left(-\frac{15}{73} + \frac{77}{292}\sqrt{6},\ -\frac{33}{73} - \frac{35}{292}\sqrt{6},\ \frac{1}{2} \right)^{\top}. \tag{20}$$

After projecting the spherical quartic into a planar cubic we obtain a parameterization of the form $\mathbf{c}(s) = (c_1(s), c_2(s), 0)$, where

$$c_i(s) = \frac{A_i(s) + B_i(s)\sqrt{C(s)}}{438D(s)} \tag{21}$$

with certain polynomials A_i, B_i, C and D with coefficients in $\mathbb{Q}(\sqrt{6})$. For instance,

$$\begin{aligned} D(s) = {} & 5032323912s^3 + (7828867480\sqrt{6} + 7806587880)s^2 \\ & + (4537170456\sqrt{6} + 46260368554)s + 10293044615\sqrt{6} + 24121304410. \end{aligned} \tag{22}$$

Finally, the coordinates of the characteristic curve in the moving system are given by expressions of the form

$$\frac{\left(p_7(s)\sqrt{C(s)} + p_9(s)\right)\left(p_{16}(s)\sqrt{C(s)} + p_{18}(s)\right)}{\left(p_4(s)\sqrt{C(s)} + p_6(s)\right)^3 \left(p_3(s)\right)^3}, \tag{23}$$

where p_i represents a polynomial of degree i with coefficients in $\mathbb{Q}(\sqrt{6})$.

Example 3. We performed a similar computation for a screw motion of the convex surface. The resulting characteristic curves are shown in Figure 3.

Example 4. We modeled a robot-like structure by composing three spheres and two non-convex QSS. This structure performs a motion which is generated by two uniform rotations of the arms. Figure 4 shows some characteristic curves which are created during this motion. A collision detection can now be done by checking intersections between the characteristic curves and the environment. This type of robot-like structures could be used as bounding volumes for real mechanical devices.

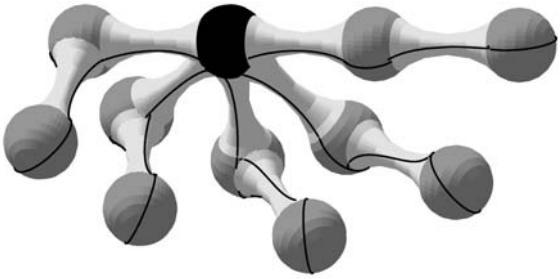


Fig. 4 Characteristic curves of several positions of a moving robot-like structure, forming the envelope surface.

Acknowledgement

The authors were supported by the WTZ programme of the Austrian Exchange Service and by the Austrian Science Fund through the research network "Industrial Geometry", subproject 9202.

References

1. Abdel-Malek, K., Yang, J., Blackmore, D. and Joy, K. (2006), Swept volumes: Fundation, perspectives, and applications, *International Journal of Shape Modeling* **12**(1), 87–127.
2. Bottema, A. and Roth, B. (1979), *Theoretical Kinematics*, North-Holland, Amsterdam.
3. Dupont, L., Lazard, D., Lazard S. and Petitjean, S. (2008), Near-optimal parameterization of the intersection of quadrics: {I. The generic algorithm, II. A classification of pencils and III. Parameterizing singular intersections}, *Journal of Symbolic Computation* **43**(3), 168–232.
4. Flaquer, J., Gárate, G. and Pargada, M. (1992), Envelopes of moving quadric surfaces, *Computer Aided Geometric Design* **9**(4), 299–312.
5. Peternell, M., Pottmann, H., Steiner, T. and Zhao, H. (2005), Swept volumes, *Computer-Aided Design and Applications* **2**, 599–608.
6. Sĭr, Z., Gravesen, J. and Jüttler, B. (2008), Curves and surfaces represented by polynomial support functions, *Theoretical Computer Science* **392**, 141–157.
7. Wang, W., Goldman, R. and Tu, C. (2003), Enhancing Levin's method for computing quadric-surface intersections, *Computer Aided Geometric Design* **20**(7), 401–422.
8. Xia, J. and Ge, Q. J. (2001), On the exact representation of the boundary surfaces of the swept volume of a cylinder undergoing rational Bézier and B-spline motions, *Journal of Mechanical Design* **123**(2), 261–265.

A Compound-Structure Frame for Improving the Performance of a Dielectric Elastomer Actuator

Giovanni Berselli, Rocco Vertechy, Gabriele Vassura and Vincenzo Parenti Castelli

DIEM, Mechanical Engineering Department, University of Bologna, 40136 Bologna, Italy, e-mail: {giovanni.berselli, rocco.vertechy, gabriele.vassura, vincenzo.parenti}@unibo.it

Abstract. Actuators based on Dielectric Elastomer films have been rising as a promising technology in robotic and mechatronic applications. Their overall behavior is highly influenced by the geometry and structural properties of the frame that is coupled with the active film. In this paper, a compound-structure compliant frame is proposed with the aim of obtaining an actuator capable of providing a quasi-constant force over a certain range of motion. A "diamond" shaped frame, designed to support the pre-stretched active film, is coupled with a "delta" shaped element, designed to modify the overall actuator stiffness and improve its performance. Both the diamond and the delta element are fully compliant mechanisms whose main structural parameters are calculated using pseudo-rigid-body models and then verified through finite element analysis. Simulations show promising performance of the proposed actuator.

Key words: dielectric elastomer actuators, frame design, compliant mechanisms.

1 Introduction

Dielectric Elastomers (DE) belong to the class of smart materials widely known as electroactive polymers (EAPs) and can be used for linear actuation in robotics and mechatronics. DE are used as films that need to be coupled with a supporting frame whose structural characteristic and geometry highly influence the overall actuator performances. Objective of the present paper is to design a compound-structure flexible frame which enables obtaining an actuator capable of providing a quasi-constant force over a certain range of motion. In particular, a compliant "diamond", similar to the one adopted in Plante (2006), is coupled with a compliant "delta" mechanism (Figures 1(a) and 1(b)), in order to modify the DE actuator output and obtain a constant force over a given displacement. A pseudo-rigid-body model (PRBM) (Howell, 2001) of the system is then derived and used for a first attempt sizing of the flexible hinges.

Finally, some finite element analysis (FEA) results are reported, that confirm the expected behavior of the proposed frame.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 291–299.

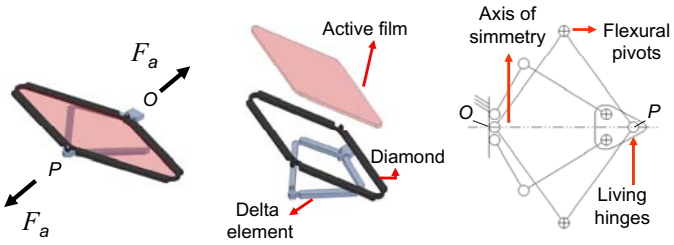


Fig. 1 Concept behind the proposed solution. 3D model (a), assembly exploded view (b), and compound-structure frame schematic (c).

2 General Aspects of Frame Design

In general, the film deformation produces a variation of the actuator length $l = |(P - O)|$, where P and O are, for instance, two points of the actuator lying on its axis of symmetry (Figure 1(c)), and a force having the same direction of vector $\overline{OP}$ that can be supplied to an external user. This force, called the actuator available thrust, F_a, is the resultant of two internal forces:

- The frame reaction force, F_s, due to the frame own stiffness, that is a function of the actual actuator configuration. The flexible frame behaves, in general, like a non-linear compression spring coupled in parallel with the DE.
- The DE film force F_f which represents the resultant force in the direction of actuation defined by vector $\overline{OP}$ due to the stress field arising in the DE. This field depends on the amount of given pre-stretch, on the applied electric potential (voltage), and on the actuator configuration. If material viscoelasticity and hysteresis are neglected, the film behaves, in general, as a non-linear tension spring. Figure 2 plots the theoretical film force against the actuator length $l = l_0 + x$ for a diamond shaped DE, where l_0 is the actuator initial length under no load and no voltage and x is the actual actuator stroke. It can be seen that a voltage application (DE film activation) causes the force F_f to drop (Plante, 2006).

The force F_a is therefore given by the difference between the film and the frame force ($F_a = F_f - F_s$). Conventionally, F_a is the force that an external user supplies to the actuator.

Assuming the film electromechanical characteristics as given, the elastic frame stiffness can be designed in order to obtain an actuator capable of providing a quasi-constant force over a given range of motion. Let us suppose to couple the DE with a compliant mechanism whose elastic reaction force increases as the distance $|P - O|$ increases during the actuator outstroke. The operating principle is illustrated in Figure 3 where the moduli of the forces are shown. An actuator working cycle (Vogan, 2004) is represented by the path $\overline{ABB''A''}$. Three different curves for F_s are represented assuming that the actuator stroke x and the initial length l_0 are invariant. Let us consider first, a frame force-position (FP) curve depicted by

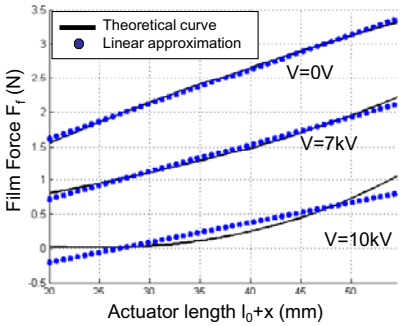


Fig. 2 Diamond shaped DE film force under different actuation voltages, Material = VHB4905, $r_2 = 55$ mm (Figure 5), DE film thickness $t = 1.5$ mm, pre-stretch ratio in the thrust direction $\lambda_1 = 2.2$, pre-stretch ratio in the direction perpendicular to the trust direction $\lambda_2 = 5$, stress field based on a first order Ogden model (Ogden, 1972).

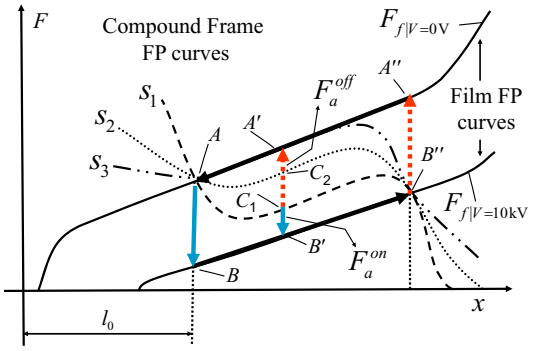


Fig. 3 FP curves qualitatively showing the moduli of F_f and F_s.

the curve s_1, where the frame configuration is defined as the length $l = |P - O|$. It can be seen that, for a large part of the stroke, F_a maintains a constant value, F_a^{on}, equal to the distance $\overline{B'C_1}$ if the DE film is activated, whereas it maintains a constant value, F_a^{off}, equal to the distance $\overline{A'C_1}$ if the DE film is inactivated. If the actuator is required to supply a larger thrust when the DE film is active (actuator on ON-state mode), a frame FP profile alike curve s_2 can be chosen, so as to increase F_a^{on} from $\overline{B'C_1}$ to $\overline{B'C_2}$ (and consequently decrease F_a^{off}). The actuator thrust on ON-state mode F_a^{on} is maximized by designing a frame that provides a FP profile alike curve s_3. In such a case, however, no restoring force can pull back the actuator to its initial position when the voltage is switched off (actuator on OFF-state mode) and a returning device has to be provided. Alternatively, the frame can be designed to maintain a restoring force, F_a^{off}, to a minimum sufficient to overcome the actuator internal dissipative forces during the backstroke; this last solution will be sought for hereafter.

One way to achieve the wanted result is to couple a given frame geometry (e.g. a diamond, see Figure 1(b)) that optimizes the DE film behavior under activation with an additional mechanism capable of modifying the overall actuator FP profile. In the case of coupled mechanisms, the resultant frame force F_s is the sum of the force F_{sd}, generated by the frame that supports the DE film, and the force F_{sc}, generated by the additional mechanism.

With regard to preferable geometries of DE supporting frame, previous work by Plante (2006) and Vogan (2004), demonstrated that diamond shaped compliant mechanisms exhibit good characteristics in terms of DE properties exploitation. For such geometries, since the obtainable F_{sd} monotonically increases during the actuator outstroke, the additional mechanism should provide a force F_{sc} whose value decreases along a relevant part of the actuator outstroke. In general, this behavior is exhibited by mechanisms characterized by unstable equilibrium positions at certain configurations. Plante (2006) and Vogan (2004) proposed the use of a bistable "snap through buckled beam" element or the use of an "over the center" mechanism. The authors themselves remarked both the advantages and the critical aspects of such designs. Our proposal is to couple a compliant "diamond" with a double slider-crank mechanism, which reaches an unstable configuration when the crank is perpendicular to the slider direction of motion. Both the diamond and the delta are fully compliant mechanisms that need to be suitably sized for the specific application.

3 Analytical Model Development

As previously stated, the actuator available thrust is given by:

$$F_a(V, x) = F_f(V, x) - F_s(x) \tag{1}$$

where x is the actuator stroke and V is the voltage applied to the DE film. F_s is also given by:

$$F_s(x) = F_{sd}(x) + F_{sc}(x) \tag{2}$$

F_{sd} and F_{sc} being forces supplied by the diamond frame and the delta mechanism respectively. The engineering drawing of the diamond frame is depicted in Figure 4; it features rigid links connected by elastic joints (small-length flexural pivots). In order to allow a comparison with previously published results, the diamond mechanism size, initial length, and stroke are the same as those described in Plante (2006). Due to the symmetry of both mechanism and load, the diamond frame of Figure 4 can be schematized as the mechanism depicted in Figure 5(a) where $\bigoplus$ means flexural hinge. In addition, half frame can be modeled as a slider-crank mechanism (Figure 5(b)).

A PRBM is used to derive the FP relationship as a function of the flexure compliances. The torques T_i due to the deflection of the small-length flexural pivots are given by:

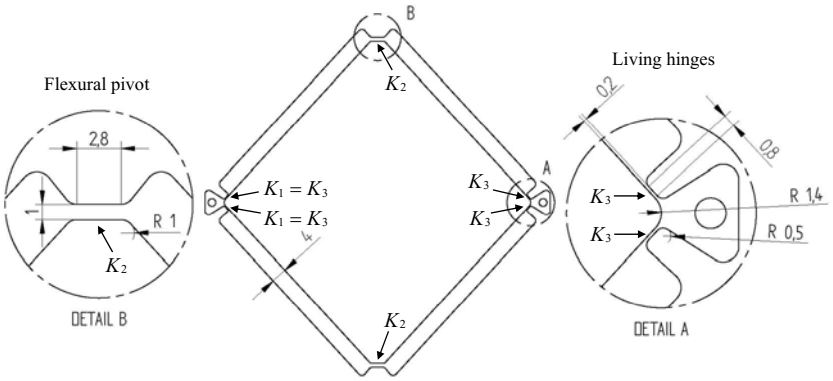


Fig. 4 Engineering drawing of the diamond frame.

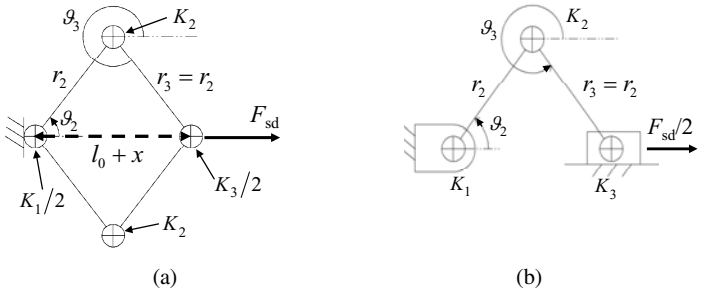


Fig. 5 Diamond frame schematic (a) and slider-crank mechanism half diamond model (b).

$$T_i = -K_i \Psi_i, \quad i = 1, \ldots, 3, \tag{3}$$

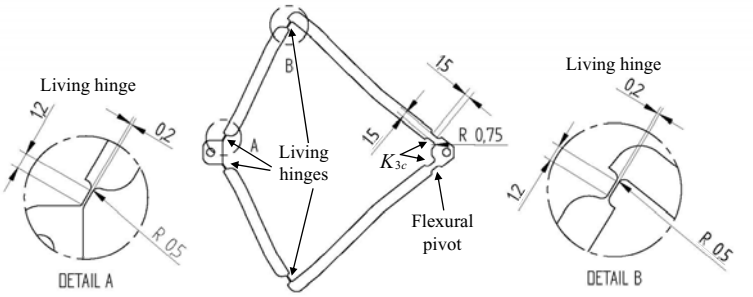
where, with reference to Figure 5, ϑ_2 is the crank angle, ϑ_3 is the connecting rod angular position, r_2 and r_3 are crank and connecting rod lengths respectively (Figure 5(a)), ϑ_{20} and ϑ_{30} are the uninflected positions of the elastic joints, K_1, K_2, K_3 are the torsional stiffnesses of the pivots to be designed, and $\Psi_1 = \vartheta_2 - \vartheta_{20}$, $\Psi_2 = (\vartheta_2 - \vartheta_{20}) - (\vartheta_3 - \vartheta_{30})$, $\Psi_3 = \vartheta_3 - \vartheta_{30}$.

The diamond frame is designed such that $r_3 = r_2$. From the mechanism position analysis the following relationships are found:

$$\vartheta_3 = 2\pi - \vartheta_2, \quad l_0 + x = 2r_2 \cos(\vartheta_2) \tag{4}$$

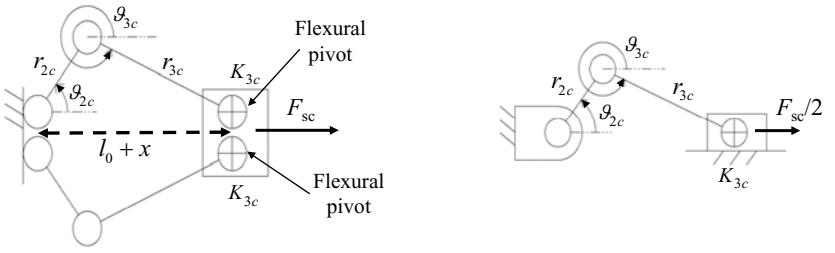
If the compliant diamond frame is formed by a monolithic piece, then $\vartheta_{30} = 2\pi - \vartheta_{20}$. The FP relationship can be found using the principle of virtual work (Howell, 2001), and solving for the output force as a function of the crank angle:

$$F_{sd} = -\frac{K_{\text{eq}}}{r_2} \frac{(\vartheta_2 - \vartheta_{20})}{\sin(\vartheta_2)} \tag{5a}$$



(a)

Fig. 6 Engineering drawing of the delta element.



(a) (b)

Fig. 7 Delta element schematic (a), slider crank mechanism half model (b).

where

$$\vartheta_2 = acos\left(\frac{x + l_0}{2r_2}\right), \quad K_{\text{eq}} = K_1 + K_3 + 4K_2, \tag{5b}$$

and l_0 is the initial actuator length under no applied load and no voltage. Equation (5a) shows that the frame force, in the output direction, is highly affected by K_2, the stiffness of the external pivots (see Figure 4). Supposing $K_1 = K_3 = 0$ (living hinges) then $K_2 = K_{\text{eq}}/4$. From the value of K_2, the dimensions of the small-length flexural pivots can be derived. Supposing the flexures being straight beam hinges with rectangular cross section then $K_i = EI_a/L$, where E is the frame material Young modulus, L is the length of the small-length flexural pivot, and $I_a = h^3 w/12$ is the moment of inertia of the pivot cross sectional area with respect to the axis a (h and w being pivot thickness and width respectively, whereas a is the barycentric axis parallel to the width).

With regard to the "delta" element, its engineering drawing is depicted in Figure 6. It features four beams, considered as perfectly rigid, connected by six elastic joints. The delta frame can be modeled as a double slider-crank mechanism with rotational springs on the slider pivots (Figure 7(a)). Due to symmetry, half frame can be modeled as a single slider-crank mechanism (Figure 7(b)). The torque due to the deflection of the flexural pivot is given by:

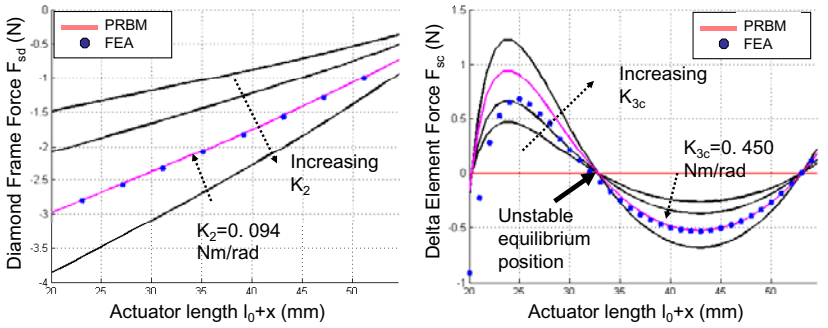


Fig. 8 Diamond frame FP curves, $\vartheta_{20} = 55°$, $r_2 = 55$ mm (a) and delta element FP curves, $\vartheta_{30c} = -38°$, $r_{2c} = 33$ mm, $r_{3c} = 46.5$ mm.

$$T_{3c} = -K_{3c}\Psi_{3c} \tag{6}$$

where, with reference to Figure 7, ϑ_{2c} is the crank angle, ϑ_{3c} is the connecting rod angular position, r_{2c} and r_{3c} are the crank and the connecting rod lengths respectively (Figure 7(a)), ϑ_{30c} is the uninflected position of the flexural pivot, K_{3c} is the pivot torsional stiffness to be designed, and $\Psi_{3c} = \vartheta_{3c} - \vartheta_{30c}$. From the position analysis of the mechanism, the following relationships are found:

$$\vartheta_{3c} = asin\left(\frac{-r_{2c}\sin(\vartheta_{2c})}{r_{3c}}\right), \quad \vartheta_{2c} = acos\left(\frac{(x+l_0)^2 + {r_{2c}}^2 - {r_{3c}}^2}{2xr_{2c}}\right) \tag{7}$$

The delta length and the diamond length are constrained to be equal by design (Figure 1(c)). Solution of the slider-crank mechanism via the principle of virtual work leads to the following FP relationship for the delta element:

$$F_{sc} = \frac{2K_{3c}}{r_{3c}}\frac{(\vartheta_{3c} - \vartheta_{30c})\cos(\vartheta_{2c})}{\sin(\vartheta_{2c} - \vartheta_{3c})} \tag{8}$$

Figures 8(a) and 8(b) show the FP curves of the compliant mechanisms as a function of the parameters K_2 and K_{3c} respectively. In particular Figure 8(a) shows that the diamond is capable of supplying negative forces in virtue of its own stiffness ($K_2 \neq 0$) whereas Figure 8(b) shows that the delta FP curves are close to a straight line with a negative slope around the unstable equilibrium point (USP). From Figures 2, 8(a), and 8(b), it can be seen that a proper choice of K_2 and K_{3c} makes it possible to conceive a quasi constant force actuator capable of sustaining compressive forces. In particular K_2 is chosen such that the film force under no voltage and the diamond reaction force have the same moduli around the USP of the delta element whereas K_{3c} is chosen so as to flatten the "film + diamond" FP curve. The characteristic dimensions of the final frame design are found by a trial and error procedure based on the analytical model developed. Final design data (including

Table 1 Frame structural parameters.

Parameter	Diamond frame	Delta element
Characteristic dimensions	$K_2 = 0.094$ Nm/rad $\vartheta_{20} = 55°$ $r_2 = 55$ mm	$K_{3c} = 0.450$ Nm/rad $\vartheta_{30c} = -38°$ $r_{2c} = 33$ mm, $r_{3c} = 46.5$ mm
Material	Delrin	Polypropylene
Flexural pivots (w × h × l)	1 × 1 × 2.8 (mm)	1.5 × 1.5 × 1.5 (mm)
Living hinges (w × h × l)	1 × 0.2 × 0.8 (mm)	1.5 × 0.2 × 1.2 (mm)
Stiffness ratio between flexural pivot and living hinge	35	337

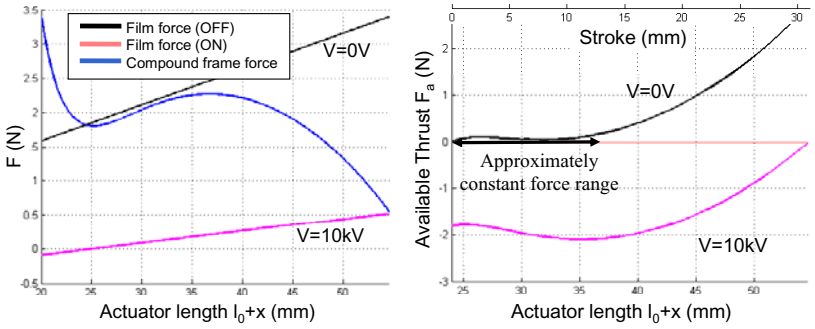


Fig. 9 Analytical FP curves showing DE film force F_f and overall frame force modulus $|F_s|$ (final design) (a), DE actuator FP curves when coupled with the delta element (final design) (b).

materials) are reported in Table 1. Figures 4 and 6 show the diamond and the delta element in the rest configuration.

In order to validate the above mentioned analytical results, neo-Hookean material constitutive models of both diamond frame and delta element final designs were analyzed through the finite element method. As depicted in Figures 8(a) and 8(b), the FEA results (dotted line) show good agreement with the behavior predicted by the PRBM for most of the usable actuator stroke. Note that the discrepancies between FEA and PRBM results, which can be seen in Figure 8(b) for delta lengths below 30 mm, are due to the large deflections, and thus moments, arising in the living hinges when the delta is mostly folded. Indeed, such bending moments are not accounted for in the PRBM and are overestimated by the Neo-Hookean model used in the FEA. The FP relationship of the overall frame force F_s is plotted in modulus against the theoretical DE film force F_f in Figure 9(a). The frame behavior is as expected (similar to curve the s_3 in Figure 3). Figure 9(b) shows the overall actuator FP curve (compound-structure frame coupled with the DE film). The backstroke available thrust keeps a value close to zero over the range 24–36 mm (in this range

a maximal deviation by 0.15 N is admitted). In order to prevent the actuator from working in the non-linear range, mechanical stops can be provided.

4 Conclusions

A novel design for a dielectric elastomer actuator has been proposed. The well known "diamond" actuator has been coupled with a "delta" compliant mechanism in order to obtain a constant available thrust over a certain range of motion. Both the coupled compliant mechanisms were sized on the basis of pseudo-rigid-body models and the chosen designs were verified using finite element analysis. The overall behavior of the actuator model confirms that compound-structure frame can be a efficient tool as long as the additional mechanism can be tailored to achieve specific response without any need to modify the basic frame geometry. Further important issues still deserve an in-depth investigation, such as, for instance, the achievable fatigue life and the actuator weight to power ratio obtainable using such a compound-structure frame.

References

Howell, L.L. (2001), *Compliant Mechanisms*, John Wiley and Sons, New York.

Jensen, B.D. and Howell, L.L. (2004), Bistable configurations of compliant mechanisms modeled using four links and translational joints, *ASME Journal of Mechanical Design* **126**, 657–666.

Kornbluh, R., Pelrine, R. and Joseph J. (1995), Elastomeric dielectric artificial muscle actuators for small robots, in *Proceedings of the Third IASTED International Conference on Robotics and Manufacturing*, Cancun, Mexico.

Ogden, R.W. (1972), Large deformation isotropic elasticity: on the correlation of theory and experiment for incompressible rubber-like solids, *Proc. Roy. Soc. London A* **326**, 565–584.

Plante, J.S. (2006), Dielectric elastomer actuators for binary robotics and mechatronics, Ph.D. Thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA.

Vogan, J. (2004), Development of dielectric elastomer actuators for MRI devices, M.S. Thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA.

Wingert, A., Lichter M.D. and Dubowsky, S. (2006), On the design of large degree-of-freedom digital mechatronic devices based on bistable dielectric elastomer actuators, *IEEE/ASME Transactions on Mechatronics* **11**(4), 448–456.

Transitions between Multiple Solutions of the Direct Kinematic Problem

E. Macho, O. Altuzarra, C. Pinto and A. Hernandez

Department of Mechanical Engineering, University of the Basque Country, 48013 Bilbao, Spain, e-mail: {erik.macho, oscar.altuzarra, charles.pinto, a.hernandez}@ehu.es

Abstract. The direct kinematic problem in parallel manipulators has multiple solutions that are traditionally called assembly modes. Non-singular transitions between some of these solutions have been detected and shown in the past. Cusp points have been defined as special points on the projection of the singularity curve onto the joint space that have the property of allowing such a non-singular transitions when encircling them. In this paper the authors will show that the condition for such a transition is more general. Authors also argue about the need for a differentiation between the concept of assembly mode and solution of the direct kinematic problem.

Key words: planar parallel manipulators, singularities, assembly modes, configuration space, cusp points.

1 Introduction

In general, parallel manipulators have multiple solutions of the Direct Kinematic Problem (DKP) that are called assembly modes. Singularities of the DKP are found by analysis of the conditions that make the determinant of the DKP Jacobian, $|J_{DKP}|$, vanish. These singularities are very important because the platform becomes uncontrollable. The DKP singularity locus divides the configuration space into parts with a different sign of $|J_{DKP}|$.

Therefore, a DKP singular configuration is reached when several DKP solutions coalesce at the same posture, i.e. the solution has a multiplicity equal to two or higher [5]. The hypothesis that two assembly modes were separated by one of these singularities was proposed also in [5]. Hence, any path joining two distinct DKP solutions should cross one of these singular postures.

However, in [6] it was shown on the 3-R<u>P</u>R planar parallel manipulator that this assumption was not always verified. The use of the configuration space was proposed in [9] to show these transitions by plotting one output variable against two inputs (with the third input constant). Some special points on the projection of the singularity locus onto the joint space, where three DKP solutions simultaneously coalesce, were found and called cusp points. A non-singular transition of DKP solu-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 301–310.

tion was possible by encircling them, and a procedure to check them was also given in [9].

Concepts such as aspects, i.e. the maximal singularity free domains where more than one DKP solution can be found, and uniqueness domains, where only one DKP solution exists, were introduced in [2, 3, 10]. Further explanations on how it is possible to make a non-singular change of assembly mode in this kind of robots, called cuspidal, have been provided in [7, 12, 13]. The first example of a planar parallel manipulator performing a non-singular transition between two DKP solutions without encircling a cusp point in the singularity curves of the joint space was pointed out in [1].

In the present paper, we propose a more general condition for such transitions, and argue about the convenience of the current definition of assembly mode. To illustrate this new type of transition between DKP solutions, the 3-PRR planar platform will be used. The main difference between this manipulator and the widely referenced 3-RPR platform is the existence of multiple Inverse Kinematic Problem (IKP) solutions or working modes (in the 3-PRR). The only other cuspidal planar platform with this latter property analyzed up to date is the 3-RRR robot [4].

IKP singularities, where the determinant $|J_{\mathrm{IKP}}|$ vanishes, are workspace boundaries. At these postures, a dependence among the output variables of the platform is verified, so they may be considered inappropriate at an operational level due to manipulability restrictions. However, they do not imply a loss of control of the robot.

Each working mode has its own DKP singularities which divide the workspace into a set of singularity free regions associated with positions with different sign of $|J_{\mathrm{DKP}}|$. A common practice in the use of manipulators is to keep them at all time in the same singularity free region, so the practical workspace remains limited. However, in [8] it was explained how several singularity free regions associated to different working modes can be joined to achieve a larger practical workspace. IKP singularities are considered as gates which allow the transition among singularity free regions associated to different working modes without risk of uncontrollability.

2 Showing Transitions on a Reduced Configuration Space

The configuration space is the locus of postures of a mechanism in a mixed domain which includes input and output variables simultaneously. Its projections onto the input or output variables spaces are the joint space and workspace respectively. The aforementioned projections can be used to plot the singularity locus and locate multiple solutions of the DKP. For the 3-RPR planar parallel manipulator used in [6] these plots are shown in Figures 1 and 2.

In this manipulator, the singularity locus is a surface that splits the workspace into two unconnected subspaces of postures with different sign in the $|J_{\mathrm{DKP}}|$. The number of different solutions of the DKP in this manipulator is six at most (S_i in Figure 1b), and are equally distributed at both sides of that singular surface. There-

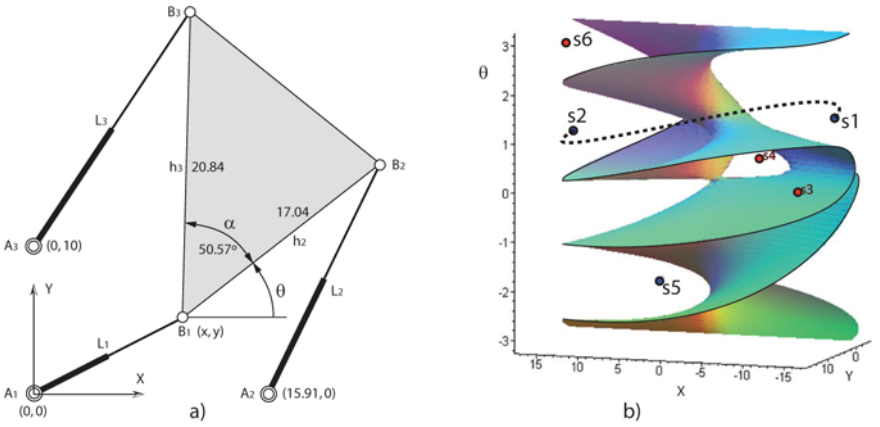


Fig. 1 (a) 3-RPR planar parallel manipulator. (b) Singularity locus and multiple DKP solutions S_i in the workspace (for $L_1 = L_2 = L_3 = 15$).

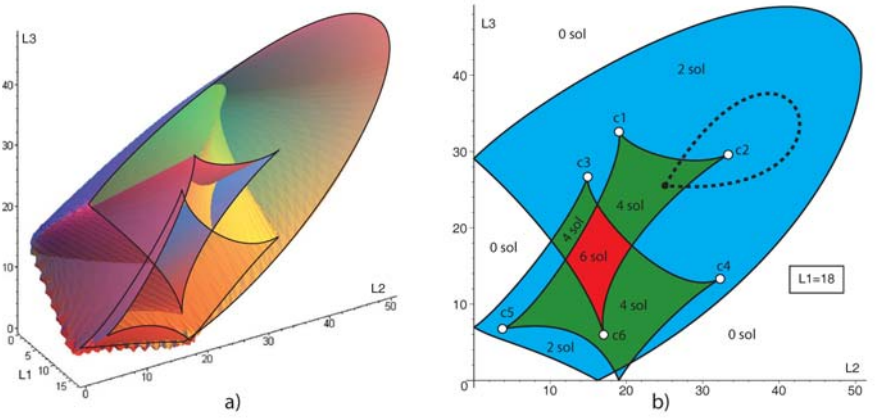


Fig. 2 (a) Singularity locus in the joint space. (b) Section at a constant value of input L_1.

fore, two DKP solutions of the same sign could be joined by non-singular paths [7] (as from S_1 to S_2 in Figure 1b).

In the joint space (Figure 2a), if an input L_1 is taken constant, a plane section is obtained with some singular curves (Figure 2b). A cusp point C_i corresponds to the coalescence of three assembly modes. Encircling it produces a non-singular transition between two solutions of the DKP with the same sign of the $|J_{\mathrm{DKP}}|$ [9].

However, McAree and Daniel [9] and Zein et al. [13] showed that if an input variable, e.g. L_1, is kept constant and one output variable, e.g. θ, is plotted against inputs L_2 and L_3, it is possible to get a reduced configuration space that is more suitable to show non-singular transitions of assembly mode. To obtain the reduced configuration space equation, $\zeta(L_2, L_3, \theta) = 0$, it is necessary to make an algebraic manipulation of the 3-RPR loop closure equations and provide a certain constant

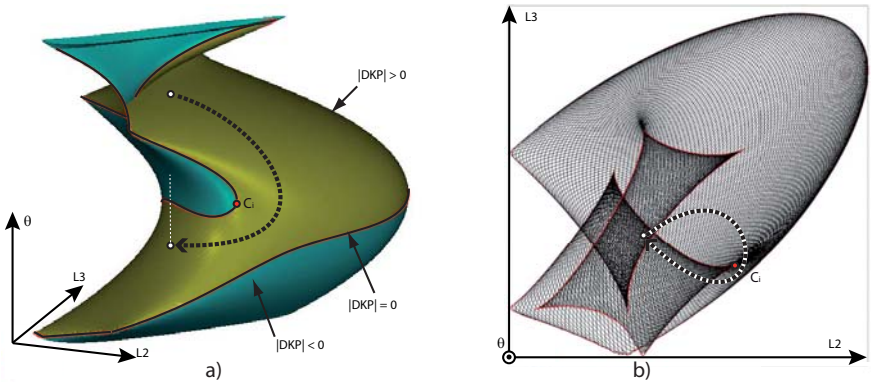


Fig. 3 (a) 3-RPR reduced configuration space. (b) Projection onto the joint space.

value for the input L_1:

$$(x - A_1^x)^2 + (y - A_1^y)^2 = {L_1}^2, \tag{1}$$

$$(x + h_2\cos(\theta) - A_2^x)^2 + (y + h_2\sin(\theta) - A_2^y)^2 = {L_2}^2, \tag{2}$$

$$(x + h_3\cos(\theta + \alpha) - A_3^x)^2 + (y + h_3\sin(\theta + \alpha) - A_3^y)^2 = {L_3}^2. \tag{3}$$

This subspace is a surface as in Figure 3a. Multiple solutions of the DKP are the intersections with a vertical line (L_2, L_3), and a path between them is shown as a curve on that surface. DKP singularity curves are given by the points in that surface with horizontal normal and for this example split the surface in two by the sign of $|J_{\text{DKP}}|$ as mentioned above.

The projection of the DKP singularity curves in Figure 3a on the L_2L_3 plane is depicted in Figure 3b, that is the same plot obtained in Figure 2b as a section of the singularity locus in the joint space. One non-singular path joining two solutions of the DKP is also projected and encircles the cusp point C_i.

In many occasions these singularity curves split the surface in several unconnected sheets (four sheets in Figure 4a), each for an assembly mode, and here it comes the hypothesis in [5]. However, in [9] it was shown that when the nonlinear system of DKP equations has a threefold root, the surface is folded like in Figure 4b. Then, three solutions of the DKP coalesce in one point C_i, i.e. a DKP solution with multiplicity three. Therefore, it is possible to come up with a non-singular path that joins two assembly modes with the same sign by encircling such a point C_i.

3 A General Condition for Non-Singular Assembly Mode Changing

This representation of the reduced configuration space as a surface suggests the idea of a more general condition for non-singular transitions between multiple solutions

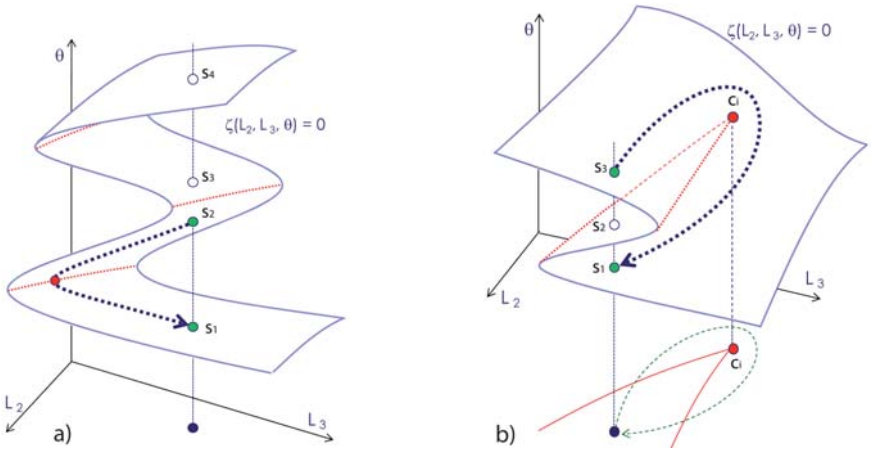


Fig. 4 Reduced configuration space. (a) Unconnected sheets. (b) Folded sheet.

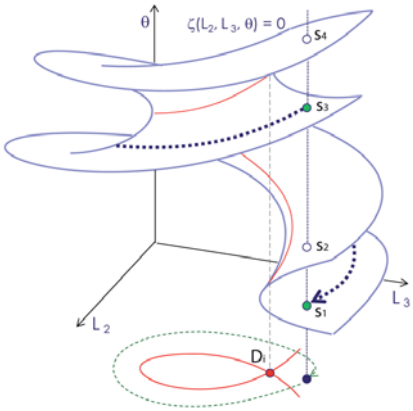


Fig. 5 Another type of folding of the configuration surface.

of the DKP. For example, an helicoidal folding of the surface as shown in Figure 5 could also be valid, and may be other possibilities exist. The projection of the singularity curve on the joint plane would not have a cusp point but another characteristic point at a loop of the projected curve, a double point, D_i. At this point there are two couples of solutions of the DKP that coalesce simultaneously, i.e. two distinct solutions of multiplicity two.

The reduced configuration space of the previous section is a part of the configuration space (L_1 constant), therefore checking the existence of a folding of the helicoidal type requires an iterative search with all values of L_1. The 3-RPR planar parallel manipulator does not have such a folding, and it can be checked in every possible plane section of the singularity locus plotted in the joint space in Figure 2a. Other planar manipulators such as the 3-RRR and the 3-PRR have it. A 3-PRR planar manipulator with a parallel layout of the sliding guides will be used (Fig-

ure 6a). This topology, firstly proposed in [1] has two advantages, it presents such helicoidal surface and the surface has the same shape for any value given to input s_1. The constraint equations for this example are:

$$(x - s_1)^2 + y^2 = {L_1}^2, \quad (4)$$

$$(x + R\sqrt{3}\cos(\theta) - s_2)^2 + (y + R\sqrt{3}\sin(\theta))^2 = {L_2}^2, \quad (5)$$

$$\left(x + R\sqrt{3}\cos\left(\theta + \frac{\pi}{3}\right) - s_3\right)^2 + \left(y + R\sqrt{3}\sin\left(\theta + \frac{\pi}{3}\right) - H\right)^2 = {L_3}^2. \quad (6)$$

Doing an algebraic manipulation of Eqs. (4), (5) and (6), the following expressions are obtained:

$$x = \lambda(s_1, s_2, s_3, \theta), \quad (7)$$

$$y = \mu(s_1, s_2, s_3, \theta). \quad (8)$$

And finally, introducing Eqs. (7) and (8) into Eq. (4), the reduced configuration space equation is achieved (the extended form of this equation is not shown due to its complexity):

$$\zeta(s_1, s_2, s_3, \theta) = 0. \quad (9)$$

Once the geometric parameters are defined, Eqs. (9), (7) and (8) can be used to solve the DKP, which is known to have multiple solutions. Given the values of the input variables s_i it is possible to find several ways of assembling the mechanism, defined by different values of the output variables x, y and θ. This kind of manipulator can have a maximum of six different DKP solutions. In a general case, parameters s_i and θ are enough to define completely the configuration of the robot. There is only one case where the procedure shown is not applicable, when a degeneracy of the DKP occurs and Eqs. (7) and (8) cannot be obtained [11]. In this case, another output parameter different from θ should be used to plot the configuration space surface.

For a null value of s_1, the resultant implicit function $\zeta(s_2, s_3, \theta) = 0$ can be depicted as the surface shown in Figure 6b. The singularity curves are given by points with horizontal normal on this surface, namely:

$$\zeta(s_2, s_3, \theta) = 0, \quad (10)$$

$$\frac{\partial \zeta}{\partial \theta} = \xi(s_2, s_3, \theta) = 0. \quad (11)$$

One of the sheets of the surface folded helicoidally is shown in more detail in Figure 7 along with a path connecting two solutions of the DKP, completely contained in such a sheet and not encircling a cusp point. In the case of a manipulator presenting multiple IKP solutions like this one, each sheet is composed by several adjacent regions associated with the different working modes of the manipulator. These regions are separated by the IKP singularity curves. When postures with $|J_{\mathrm{IKP}}| = 0$ are reached, a dependence among the output velocities is achieved, so there are manipulability restrictions, but this does not imply a risk of uncontrollability as in a

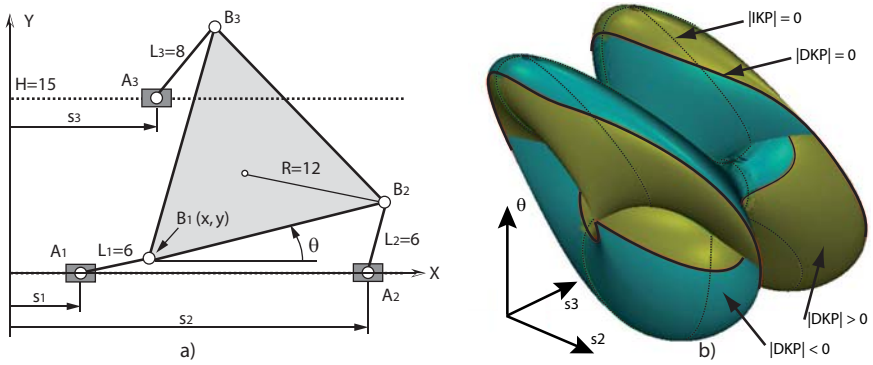


Fig. 6 (a) 3-PRR manipulator. (b) Reduced configuration space.

DKP singularity. Moreover these postures allow the transition between regions associated with the eight different working modes of this manipulator, enlarging the practical workspace. The path shown in Figure 7 crosses IKP singularity curves, so it may not be theoretically considered a non-singular transition between two DKP solutions, but since it is completely contained in a DKP singularity free sheet, the control of the robot is never lost. The sequence of input values for the elliptical path shown in Figures 7 and 8 is given by the following parametric expressions:

$$s_2 = 17.5 + 5\cos(t)\cos(9\pi/2) - 2.7\sin(t)\sin(9\pi/2), \tag{12}$$

$$s_3 = \quad 9 + 5\cos(t)\sin(9\pi/2) + 2.7\sin(t)\cos(9\pi/2). \tag{13}$$

with parameter t going from one DKP solution at $t_1 = -\pi/8$ to another solution at $t_7 = -\pi/8 - 2\pi$, corresponding to the inputs: $s_1 = 0$, $s_2 = 21.70$ and $s_3 = 11.18$.

As mentioned in the previous section, DKP singularity curves in that reduced configuration space are the edges of the sheets in which the surface is divided. The projection of the DKP singularity curves on the s_2s_3 plane can be obtained as an equation $\psi(s_2, s_3) = 0$, which comes from the elimination of the output variable θ from the system of equations given by Eqs. (10) and (11). Sometimes, those positions form a set of different closed curves ψ_i. In the example considered here, DKP singularities are composed by three different closed curves in the reduced configuration space, and therefore, also in the projection over the joint space as shown in Figure 8.

In this reduced joint space, this type of DKP solution changing imply the existence of a path encircling a loop in the singularity curve, as shown in Figure 8. In fact, it could be considered that the existence of this loop and its associated double point D_i, is actually a more general mathematical condition than the existence of a cusp point C_i, since the latter is a degeneracy of the former. Those special points should be searched in each single closed curve. This process is necessary because an analysis of the whole equation $\psi(s_2, s_3) = 0$ will provide points like e_i in Figure 8 as double points. But those points come from the crossing of different edge

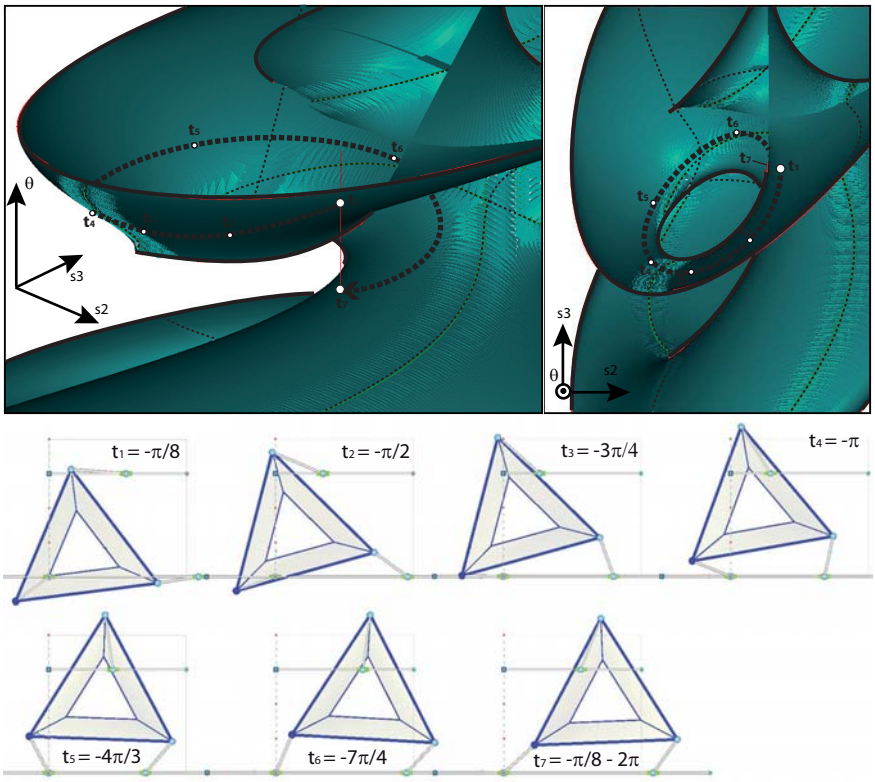


Fig. 7 Non-singular assembly mode change without encircling a cusp point.

curves and do not allow the transition between DKP solutions. Taking into account this issue, the mathematical conditions verified in double and cusp points are:

$$\partial\psi_i/\partial s_2 = \partial\psi_i/\partial s_3 = 0 \text{ (or } \infty). \tag{14}$$

It has been shown how it is possible to make a transition between certain DKP solutions, namely those which are in the same DKP singularity free regions or sheets. No way of physically distinguishing when a DKP solution becomes another has been found. The transition between both DKP solutions is completely smooth. Neither a DKP singularity is reached, nor the platform has to be disassembled. Therefore, all the DKP solutions in the same sheet of the configuration space cannot be distinguished in practice. This fact suggests that, may be the concepts of DKP solution and assembly mode should be differentiated. On the one hand, the number of DKP solutions is simply the number of different mathematical solutions of the constraint equations' system for given values of the input variables. On the other hand, the number of assembly modes should be the number of isolated subspaces into which the configuration space is divided by the DKP singularity locus.

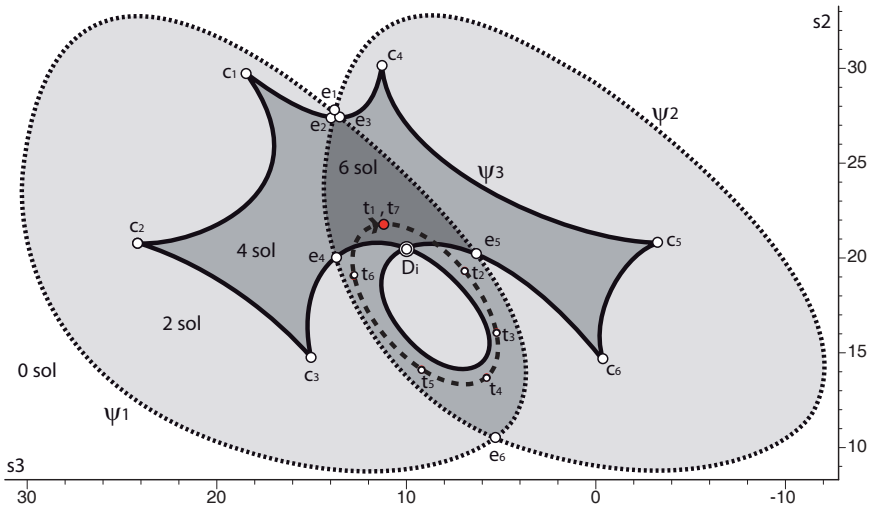


Fig. 8 Singularity locus in the joint space.

Hence, an assembly mode can be composed of more than one DKP solution, namely, those which can be joined without losing control. Three types of transitions between two different DKP solutions can be found. First, if both are located in disjoint surfaces of the configuration space, the mechanism needs to be disassembled to change between them. Second, if they are in the same surface but in different DKP singularity free sheets (Figure 4a), both solutions can be joined crossing a DKP singularity. And third, if they are in the same DKP singularity free region, the transition between them is done in a completely smooth way and there is no way of distinguishing when exactly the assembly mode is changed, no matter if the working mode (IKP solution) has changed or not (Figures 4b and 5). In the latter case both solutions could be considered as belonging to the same assembly mode.

4 Conclusions

The concepts presented in this paper provide a better understanding on how different solutions of the DKP, traditionally called assembly modes, can be joined without losing the control of the robot, and hence without disassemble the mechanism. The graphics shown have been used as an auxiliary tool to inspire the general mathematical conditions which characterize this problem. A procedure which allows to know if a mechanism has the ability of doing this has been presented. Encircling a cusp point is not the only way of doing a non-singular assembly mode change. A more general way to do this is encircling a loop in the DKP singularity curves plotted in the joint space. This loop is defined by the existence of a double point. A cusp point is a degeneracy of such loop. These singularity curves have to be separated

into simple closed curves to look for double points. It would be interesting to find a way of posing the mathematical conditions in a symbolical form and then obtain the relations among the different geometric parameters of the manipulator which ensure the capability of passing between certain DKP solutions. Moreover, it has been introduced the reason why the concept of assembly mode should be redefined.

Acknowledgements

This research work was supported in part by the Spanish Ministerio de Ciencia y Tecnologia (Project DPI2005-02207), the FEDER funds of the European Union, the Department of Education, Universities and Investigation of the Basque Government (IT-186-07) and the Grant (BFI104.148.R2) from the Basque Government.

References

1. Bamberger, H., Wolf, A., Shoham M., Assembly mode changing in parallel mechanisms. *IEEE Transactions on Robotics*, submitted (2008).
2. Chablat, D., Wenger, Ph., Working modes and aspects in fully parallel manipulator. In *Proceedings IEEE International Conference on Robotics and Automation*, pp. 1970–1976 (1998).
3. Chablat, D., Wenger, Ph., Séparation des solutions aux modèles géométriques direct et inverse pour les manipulateurs pleinement parallèles. *Mechanism and Machine Theory* **36**(6), 763–783 (2001).
4. Chablat, D., Wenger, Ph., The kinematic analysis of a symmetrical three-degree-of-freedom planar parallel manipulator. *CISM-IFToMM Symposium on Robot Design, Dynamics and Control*, Montréal (2004).
5. Hunt, K.H., Primrose, E.J.F., Assembly configurations of some in-parallel-actuated manipulators. *Mechanism and Machine Theory* **28**(1), 31–42 (1993).
6. Innocenti, C., Parenti-Castelli, V., Singularity-free evolution from one configuration to another in serial and fully-parallel manipulators. *Journal of Mechanical Design* **120**(1), 73–79 (1998).
7. Macho, E., Altuzarra, O., Pinto, C., Hernandez, A., Singularity free change change of assembly mode in parallel manipulators. Application to the 3-RPR planar platform. In *Proceedings of 12th World Congress in Mechanism and Machine Science, IFToMM 2007*, Besançon, France (2007).
8. Macho, E., Altuzarra, O., Pinto, C., Hernandez, A., Workspaces associated to assembly modes of the 5R planar parallel manipulator. *Robotica*, in press (2008).
9. McAree, P.R., Daniel, R.W., An explanation of never-special assembly changing motions for 3-3 parallel manipulators. *The International Journal of Robotics Research* **18**(6), 556–574 (1999).
10. Wenger, Ph., Chablat, D., Workspace and assembly modes in fully-parallel manipulators: A descriptive study. In *Advances in Robot Kinematics: Analysis and Control*, J. Lenarčič and M.L. Husty (Eds.), pp. 117–126. Kluwer Academic Publishers (1998).
11. Wenger, Ph., Chablat, D., Zein, M., Degeneracy study of the forward kinematics of planar 3-RPR parallel manipulators. *Journal of Mechanical Design* **129**(12) (2007).
12. Zein, M.,Wenger, Ph., Chablat, D., Singular curves and cusp points in the joint space of 3-RPR parallel manipulators. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Orlando, USA (2006).
13. Zein, M., Wenger, Ph., Chablat, D., Non-singular assembly mode changing motions for 3-RPR parallel manipulators. *Mechanism and Machine Theory* **43**(4), 480–490 (2008).

Kinetostatic Performance of a Planar Parallel Mechanism with Variable Actuation

N. Rakotomanga[1], D. Chablat[2] and S. Caro[2]

[1]*Département de Génie de la Production Automatisée, École de Technologie Supérieure, Montréal (Québec) Canada H3C 1K3, e-mail: novona.rakotomanga.1@ens.etsmtl.ca*
[2]*Institut de Recherche en Communications et Cybernétique de Nantes, UMR CNRS No. 6597, 44321 Nantes, France, e-mail: {damien.chablat, stephane.caro}@irccyn.ec-nantes.fr*

Abstract. This paper deals with a new planar parallel mechanism with variable actuation and its kinetostatic performance. A drawback of parallel mechanisms is the non-homogeneity of kinetostatic performance within their workspace. The common approach to solve this problem is the introduction of actuation redundancy, that involves force control algorithms. Another approach, highlighted in this paper, is to select the actuated joint in each limb with regard to the pose of the end-effector. First, the architecture of the mechanism and two kinetostatic performance indices are described. Then, the actuating modes of the mechanism are compared.

Key words: parallel mechanism, regular dextrous workspace, variable actuated mechanism.

1 Introduction

A drawback of serial and parallel mechanisms is the inhomogeneity of the kinetostatic performance within their workspace. For instance, dexterity, accuracy and stiffness are usually bad in the neighbourhood of singularities that can appear in the workspace of such mechanisms. As far as the parallel mechanisms are concerned, their inverse kinematics problem (IKP) has usually many solutions, which correspond to the *working modes* of the mechanism [4]. Nevertheless, it is difficult to come up with a large workspace free of singularity with a given working mode. Consequently, a trajectory planning may require a change of the working mode by means of an alternative trajectory in order to avoid singular configurations. In such a case, the initial trajectory would not be followed. The common approach to solve this problem is to introduce actuation redundancy, that involves force control algorithms [1]. Another approach is to use the concept of joint-coupling as proposed by Theingin et al. [15] or to select the actuated joint in each limb with regard to the pose of the end-effector [2], as highlighted in this paper. Therefore, we introduce a planar parallel mechanism with variable actuation, also known as *variable actuated mechanism* (VAM). First, the architecture of the mechanism and two kinetostatic performance indices are described. Then, the *actuating modes* (AMs) of the mechanism are compared based on their kinetostatic performance.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 311–320.

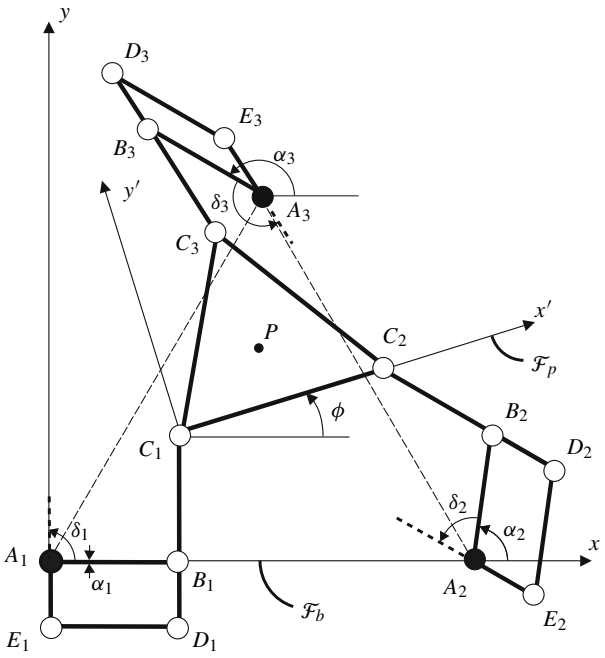


Fig. 1 3-RRR PPM with variable actuation.

2 Preliminaries

This section deals with the kinematic modeling of a new variable actuated mechanism (VAM), its singularity analysis, the presentation of some performance indices and the concept of regular dextrous workspace.

2.1 Mechanism Architecture

The concept of VAM was introduced in [2, 15]. Indeed, they derived a VAM from the architecture of the 3-RPR planar parallel manipulator (PPM) by actuating either the first revolute joint or the prismatic joint of its limbs. This paper deals with the study of a new VAM illustrated in Figure 1. This mechanism is derived from the architecture of the 3-RRR PPM. As a matter of fact, the first link of each limb of the conventional 3-RRR manipulator is replaced by parallelogram $A_i B_i D_i E_i$ to come up with the mechanism at hand. Accordingly, links $A_i B_i$ and $B_i C_i$ can be driven independently, i.e., angles α_i and δ_i are actuated and uncoupled, by means of an actuator and a double clutch mounted to the base and located in point A_i, $i = 1, 2, 3$.

Table 1 The eight actuating modes of the 3-RRR VAM.

Actuating mode number		Driven links	Active angles
1	$\underline{R}RR_1$-$\underline{R}RR_2$-$\underline{R}RR_3$	A_1B_1, A_2B_2, A_3B_3	$\alpha_1, \alpha_2, \alpha_3$
2	$\underline{R}RR_1$-$\underline{R}RR_2$-$R\underline{R}R_3$	A_1B_1, A_2B_2, A_3E_3	$\alpha_1, \alpha_2, \delta_3$
3	$\underline{R}RR_1$-$R\underline{R}R_2$-$\underline{R}RR_3$	A_1B_1, A_2E_2, A_3B_3	$\alpha_1, \delta_2, \alpha_3$
4	$R\underline{R}R_1$-$\underline{R}RR_2$-$\underline{R}RR_3$	A_1E_1, A_2B_2, A_3B_3	$\delta_1, \alpha_2, \alpha_3$
5	$\underline{R}RR_1$-$R\underline{R}R_2$-$R\underline{R}R_3$	A_1B_1, A_2E_2, A_3E_3	$\alpha_1, \delta_2, \delta_3$
6	$R\underline{R}R_1$-$R\underline{R}R_2$-$\underline{R}RR_3$	A_1E_1, A_2E_2, A_3B_3	$\delta_1, \delta_2, \alpha_3$
7	$R\underline{R}R_1$-$\underline{R}RR_2$-$R\underline{R}R_3$	A_1E_1, A_2B_2, A_3E_3	$\delta_1, \alpha_2, \delta_3$
8	$R\underline{R}R_1$-$R\underline{R}R_2$-$R\underline{R}R_3$	A_1E_1, A_2E_2, A_3E_3	$\delta_1, \delta_2, \delta_3$

It turns out that the VAM has eight *actuating modes* as shown in Table 1. Indeed, the actuating mode of the mechanism depends on its actuated joints. For instance, the first actuating mode corresponds to the 3-$\underline{\text{R}}$RR mechanism, also called $\underline{R}RR_1$-$\underline{R}RR_2$-$\underline{R}RR_3$ mechanism in the scope of this paper, as the first revolute joints (located in point A_i) of its limbs are actuated. Likewise, the eighth actuating mode corresponds to the 3-R$\underline{\text{R}}$R manipulator, also called $R\underline{R}R_1$-$R\underline{R}R_2$-$R\underline{R}R_3$ mechanism, as the second revolute joints (located in point B_i) of its limbs are actuated.

The moving platform pose of the VAM is determined by means of the Cartesian coordinates (x, y) of operation point P expressed in frame $\mathcal{F}_b$ and angle ϕ, namely, the angle between frames $\mathcal{F}_b$ and $\mathcal{F}_p$. Moreover, the passive and actuated joints do not have any stop. Points A_1, A_2 and A_3, (C_1, C_2 and C_3, respectively) lie at the corners of an equilateral triangle, of which the geometric center is point O (point P, resp.). The length of links A_iB_i and B_iC_i is equal to 3.0, $i = 1, 2, 3$. The length of segment A_1A_2 (C_1C_2, resp.) is equal to 10.0 (5.0, resp.). The unit is not specified as absolute lengths are not necessary to convey the idea.

2.2 Kinematic Modeling

The velocity $\dot{\mathbf{p}}$ of point P can be obtained in three different forms, depending on which leg is traversed, namely,

$$\dot{\mathbf{p}} = \dot{\alpha}_1\mathbf{E}(\mathbf{c}_1 - \mathbf{a}_1) + \dot{\delta}_1\mathbf{E}(\mathbf{c}_1 - \mathbf{b}_1) + \dot{\phi}\mathbf{E}(\mathbf{p} - \mathbf{c}_1) \tag{1}$$

$$\dot{\mathbf{p}} = \dot{\alpha}_2\mathbf{E}(\mathbf{c}_2 - \mathbf{a}_2) + \dot{\delta}_2\mathbf{E}(\mathbf{c}_2 - \mathbf{b}_2) + \dot{\phi}\mathbf{E}(\mathbf{p} - \mathbf{c}_2) \tag{2}$$

$$\dot{\mathbf{p}} = \dot{\alpha}_3\mathbf{E}(\mathbf{c}_3 - \mathbf{a}_3) + \dot{\delta}_3\mathbf{E}(\mathbf{c}_3 - \mathbf{b}_3) + \dot{\phi}\mathbf{E}(\mathbf{p} - \mathbf{c}_3) \tag{3}$$

with matrix $\mathbf{E}$ defined as

$$\mathbf{E} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

$\mathbf{a}_i$, $\mathbf{b}_i$ and $\mathbf{c}_i$ are the position vectors of points A_i, B_i and C_i, respectively. $\dot{\alpha}_i$, $\dot{\delta}_i$ and $\dot{\phi}$ are the rates of angles α_i, δ_i and ϕ depicted in Figure 1, $i = 1, 2, 3$.

The kinematic model of the VAM under study can be obtained from Eqs. (1)-(c) by eliminating the idle joint rates. However, the latter depend on the actuating mode of the mechanism. For instance, $\dot{\delta}_1$, $\dot{\delta}_2$ and $\dot{\delta}_3$ are idle with the first actuating mode and the corresponding kinematic model is obtained by dot-multiplying Eqs. (1)-(c) with $(\mathbf{c}_i - \mathbf{b}_i)^T$, $i = 1, 2, 3$. Likewise, $\dot{\delta}_1$, $\dot{\delta}_2$ and $\dot{\alpha}_3$ are idle with the second actuating mode and the corresponding kinematic model is obtained by dot-multiplying Eqs. (1)-(b) with $(\mathbf{c}_i - \mathbf{b}_i)^T$, $i = 1, 2$, and Eq. (3) with $(\mathbf{c}_3 - \mathbf{a}_3)^T$.

The kinematic model of the VAM can now be cast in vector form, namely,

$$\mathbf{A}\mathbf{t} = \mathbf{B}\dot{\mathbf{q}} \quad \text{with} \qquad \mathbf{t} = [\dot{\mathbf{p}}\ \dot{\phi}]^T \quad \text{and} \quad \dot{\mathbf{q}} = [\dot{q}_1\ \dot{q}_2\ \dot{q}_3]^T \tag{4}$$

with $\dot{\mathbf{q}}$ thus being the vector of actuated joint rates. $\dot{q}_i = \dot{\alpha}_i$ when link A_iB_i is driven and $\dot{q}_i = \dot{\delta}_i$ when link A_iE_i is driven, $i = 1, 2, 3$. $\mathbf{A}$ and $\mathbf{B}$ are respectively, the direct and the inverse Jacobian matrices of the mechanism, defined as

$$\mathbf{A} = \begin{bmatrix} (\mathbf{c}_1 - \mathbf{h}_1)^T & -(\mathbf{c}_1 - \mathbf{h}_1)^T\mathbf{E}(\mathbf{p} - \mathbf{c}_1) \\ (\mathbf{c}_2 - \mathbf{h}_2)^T & -(\mathbf{c}_2 - \mathbf{h}_2)^T\mathbf{E}(\mathbf{p} - \mathbf{c}_2) \\ (\mathbf{c}_3 - \mathbf{h}_3)^T & -(\mathbf{c}_3 - \mathbf{h}_3)^T\mathbf{E}(\mathbf{p} - \mathbf{c}_3) \end{bmatrix} \tag{5}$$

$$\mathbf{B} = \operatorname{diag}\big[(\mathbf{c}_i - \mathbf{b}_i)^T\mathbf{E}(\mathbf{b}_i - \mathbf{a}_i)\big], \quad i = 1, 2, 3 \tag{6}$$

where $\mathbf{h}_i = \mathbf{b}_i$ when link A_iB_i is driven and $\mathbf{h}_i = \mathbf{a}_i$ when link B_iC_i is driven, $i = 1, 2, 3$.

When $\mathbf{A}$ is non-singular, we obtain the relation

$$\mathbf{t} = \mathbf{J}\dot{\mathbf{q}} \quad \text{with} \quad \mathbf{J} = \mathbf{A}^{-1}\mathbf{B} \tag{7}$$

Likewise, we obtain

$$\dot{\mathbf{q}} = \mathbf{K}\mathbf{t} \tag{8}$$

when $\mathbf{B}$ is non-singular with $\mathbf{K}$ denoting the inverse of $\mathbf{J}$.

2.3 Singularity Analysis

The singular configurations associated with the direct-kinematic matrix of PPMs are well known [12]. For the 3-RRR PPM, such configurations are reached whenever lines (B_1C_1), (B_2C_2) and (B_3C_3) intersect (possibly at infinity). For the 3-RRR PPM, such configurations are reached whenever lines (A_1C_1), (A_2C_2) and (A_3C_3) intersect. Consequently, the singular configurations associated with the direct-kinematic matrix of the VAM are reached whenever lines (H_1C_1), (H_2C_2) and (H_3C_3) intersect where H_i stands for B_i (A_i, resp.) when link A_iB_i (B_iC_i, resp.) is driven, $i = 1, 2, 3$.

From Eq. (6), the singular configurations associated with the inverse-kinematics of the VAM are reached whenever points A_i, B_i, and C_i are aligned.

2.4 Performance Indices

We focus here on issues pertaining to manipulability or dexterity. In this regard, we understand these terms in the sense of measures of distance to singularity, which brings us to the concept of condition number in [9]. Here, we adopt the *condition number* of the underlying Jacobian matrices based on the Frobenius norm as a means to quantify distances to singularity and the *transmission angle*.

2.4.1 Condition Number

The *condition number* $\kappa_F(\mathbf{M})$ of a $m \times n$ matrix $\mathbf{M}$, with $m \leq n$, based on the Frobenius norm is defined as follows

$$\kappa_F(\mathbf{M}) = \frac{1}{m}\sqrt{\mathrm{tr}(\mathbf{M}^T\mathbf{M})\mathrm{tr}\left[(\mathbf{M}^T\mathbf{M})^{-1}\right]} \tag{9}$$

Here, the condition number is computed based on the Frobenius norm as the latter produces a condition number that is analytic in terms of the posture parameters whereas the 2-norm does not. Besides, it is much costlier to compute singular values than to compute matrix inverses.

The terms of the direct Jacobian matrix of the VAM are not homogeneous as they do not have same units. Accordingly, its condition number is meaningless. Indeed, its singular values cannot be arranged in order as they are of different nature. However, from [11] and [13], the Jacobian can be normalized by means of a *normalizing length*. Later on, the concept of *characteristic length* was introduced in [14] in order to avoid the random choice of the normalizing length. For instance, the previous concept was used in [5] to analyze the kinetostatic performance of manipulators with multiple inverse kinematic solutions, and therefore to select their best *working mode*.

2.4.2 Transmission Angle

The *transmission angle* can be used to assess the quality of force transmission in mechanisms involving passive joints. Although it is well known and easily computable for 1-DOF or single loop mechanisms [3, 8], it is not extensively used for n-DOF mechanical systems ($n > 1$) [2].

The transmission angle ψ_i is defined as an angle between vectors of force $\mathbf{Fc}_i$ and translational velocity $\mathbf{Vc}_i$ of a point to which the force is applied as illustrated in Figure 2. When link A_iB_i is driven, the direction of force $\mathbf{Fc}_i$ is the direction of link B_iC_i, namely,

$$\gamma_i = \arctan\left(\frac{y_{C_i} - y_{B_i}}{x_{C_i} - x_{B_i}}\right), \quad i = 1, 2, 3 \tag{10}$$

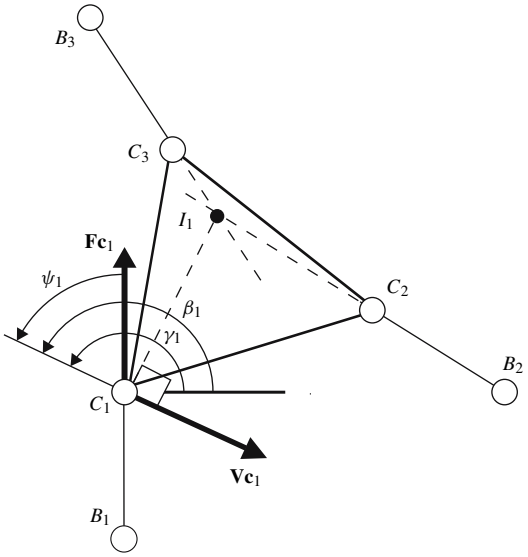


Fig. 2 Transmission angle of the 3-RRR manipulator.

Table 2 Cartesian coordinates of instantaneous centres of rotation.

	I_1	I_2	I_3
x_{I_i}	$\dfrac{b_3 - b_2}{\tan(\gamma_2) - \tan(\gamma_3)}$	$\dfrac{b_1 - b_3}{\tan(\gamma_3) - \tan(\gamma_1)}$	$\dfrac{b_2 - b_1}{\tan(\gamma_1) - \tan(\gamma_2)}$
y_{I_i}	$\dfrac{b_3 \tan(\gamma_2) - b_2 \tan(\gamma_3)}{\tan(\gamma_2) - \tan(\gamma_3)}$	$\dfrac{b_1 \tan(\gamma_3) - b_3 \tan(\gamma_1)}{\tan(\gamma_3) - \tan(\gamma_1)}$	$\dfrac{b_2 \tan(\gamma_1) - b_1 \tan(\gamma_2)}{\tan(\gamma_1) - \tan(\gamma_2)}$

Conversely, when link $A_i E_i$ is driven, the direction of force $\mathbf{Fc}_i$ is the direction of line $(A_i C_i)$, namely,

$$\gamma_i = \arctan\left(\frac{y_{C_i} - y_{A_i}}{x_{C_i} - x_{A_i}}\right), \quad i = 1, 2, 3 \tag{11}$$

The instantaneous centre of rotation depends on the leg under study. For example, instantaneous centre of rotation I_1 associated with leg 1 is the intersecting point of forces $\mathbf{Fc}_2$ anf $\mathbf{Fc}_3$. Table 2 gives the Cartesian coordinates of instantaneous centre of rotation I_i associated with the ith leg of the VAM, expressed in frame $\mathcal{F}_b$, with $b_i = y_{C_i} - x_{C_i} \tan\gamma_i, i = 1, 2, 3$. The direction of $\mathbf{Vc}_i$ is defined as,

$$\beta_i = \arctan\left(\frac{y_{C_i} - y_{I_i}}{x_{C_i} - x_{I_i}}\right) + \frac{\pi}{2}, \quad i = 1, 2, 3 \tag{12}$$

The transmission angle related to the ith leg of the VAM is defined as follows:

$$\psi_i = |\gamma_i - \beta_i|, \quad i = 1, 2, 3 \tag{13}$$

and the transmission angle ψ of the mechanism is defined as,

$$\psi = \max(\psi_i), \quad i = 1, 2, 3 \tag{14}$$

Finally, the smaller ψ, the better the force transmission of the mechanism.

2.4.3 Regular Dextrous Workspace

A manipulator had better keep good and homogeneous performance within its workspace. For that reason, the concept of *regular dextrous workspace* is introduced in [6]. In fact, the regular dextrous workspace of a manipulator is a regular-shaped workspace included in its Cartesian workspace with good and homogeneous performance. As we focus on the kinetostatic performance of the VAM in the scope of this paper, we consider only the condition number of its kinematic Jacobian matrix and its transmission angle as performance indices.

3 Actuating Modes Comparison

For the VAM under study, the inverse condition number of its kinematic Jacobian matrix, i.e., $\kappa_F^{-1}(\mathbf{J})$ with $\mathbf{J}$ defined in Eq. (7), varies from 0 to 1 within its workspace $\mathcal{W}$. Likewise, its transmission angle ψ, defined in Eq. (14), varies from 0 to 90° within $\mathcal{W}$. From [2], a mechanism has good kinetostatic performance as long as its transmission angle is smaller than 75°. Let us assume that the kinetostatic performance are good as well as long as $\kappa_F^{-1}(\mathbf{J}) > 0.15$. Therefore, we claim that the VAM and its actuating modes (AMs) have good kinetostatic performance as long as $\kappa_F^{-1}(\mathbf{J})$ is higher than 0.15 and ψ is smaller than 75°.

First, let us compare the size of the workspace corresponding to AMs of the VAM given in Table 1, based on the two previous kinetostatic performance indices. In this vein, let us consider that the orientation, ϕ, of the moving platform of the VAM is constant and the latter stays as far as possible from singular configurations, i.e., let ϕ be equal to 17.5°. From Table 3, we can notice that the size of the workspace corresponding to the 2nd, 3rd and 4th AMs is the same. Likewise, the size of the workspace corresponding to the 4th, 5th and 6th AMs is the same. This is due to the symmetric architecture of the mechanism. Moreover, the largest workspace is obtained with the 1st AM and the smallest one with the 8th AM. Finally, we can notice that the two kinematic performance indices give similar results.

In order to compare the AMs of the VAM, we also assume that its regular dextrous workspace (RDW) is a cylinder, of which the section depicts the position (x, y) of its moving platform and the height shows the rotation ϕ of the latter. Let ϕ vary between 5° and 25°. Figures 3a–e (Figures 4a–e, resp.) illustrate the kinetostatic

Table 3 Ratio of the VAM actuating modes workspace size to the VAM workspace size with $\kappa_F^{-1}(\mathbf{J}) > 0.15$ and $\psi < 75°$, $\phi = 17.5°$.

Actuating mode number	Workspace size ratio [%]	
	$\kappa_F^{-1}(\mathbf{J}) > 0.15$	$\psi < 75°$
1	88.27	83.16
2,3,4	75.33	71.93
5,6,7	62.26	70.76
8	52.15	71.86

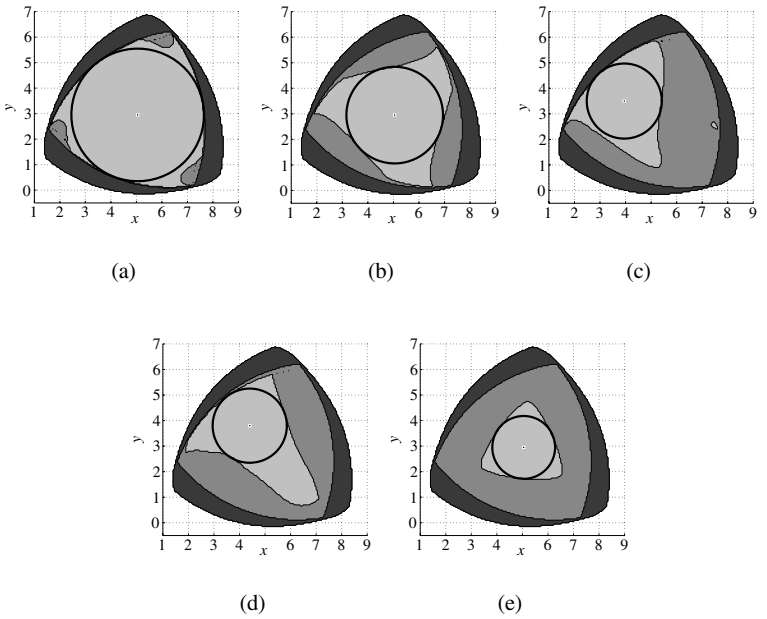


Fig. 3 RDW obtained with $\kappa_F^{-1}(\mathbf{J}) > 0.15$ of the (a) VAM; (b) 1st AM; (c) 2nd, 3rd and 4th AM; (d) 5th, 6th and 7th AM; (e) 8th AM.

performance of the VAM and its AMs within the workspace based on $\kappa_F^{-1}(\mathbf{J})$ (ψ, resp.). The dark zones depict the positions of P, in which ϕ cannot vary continuously between 5° and 25°. The dark gray zones depict the positions of P, in which ϕ can vary continuously between 5° and 25°, but $\kappa_F^{-1}(\mathbf{J})$ (ψ, resp.) is not necessarily higher (smaller, resp.) than 0.15 (75°, resp.). The light gray zones depict the positions of P, in which ϕ can vary continuously between 5° and 25° and $\kappa_F^{-1}(\mathbf{J})$ (ψ, resp.) is higher (smaller, resp.) than 0.15 (75°, resp.). Finally, the circles describe the RDW of the VAM and its AMs based on $\kappa^{-1}(\mathbf{J})$ (ψ, resp.).

Table 4 gives RDW radius of the VAM and its AMs obtained with $\kappa_F^{-1}(\mathbf{J}) > 0.15$ and $\psi < 75°$. We can notice that the results obtained with the two kinetostatic per-

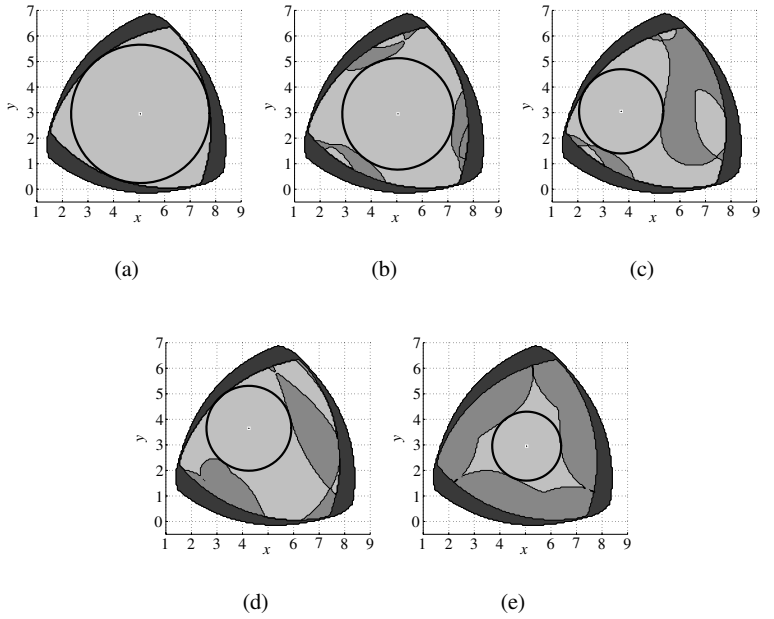


Fig. 4 RDW obtained with $\psi < 75°$ of the (a) VAM; (b) 1st AM; (c) 2nd, 3rd and 4th AM; (d) 5th, 6th and 7th AM; (e) 8th AM.

Table 4 RDW radius of the VAM and its AMs obtained with the two kinetostatic performance indices.

Actuating mode number	RDW radius	
	$\kappa_F^{-1}(\mathbf{J}) > 0.15$	$\psi < 75°$
1	1.89	2.18
2, 3, 4	1.47	1.65
5, 6, 7	1.45	1.66
8	1.23	1.35
VAM	2.60	2.71

formance indices are similar. Besides, the largest RDW is obtained with the 1st AM and the smallest one with 8th AM.

4 Conclusions

In this paper, we introduced a new planar parallel mechanism with variable actuation, which is derived from the architecture of the 3-RRR and 3-RRR PPMs. Then,

we used two indices, namely, the condition number of its kinematic Jacobian matrix and its transmission angle to compare its actuating modes. The concept of regular dextrous workspace was also used. It turns out that the mechanism with variable actuation can cover almost all its workspace with good and homogeneous kinetostatic performance as it takes advantage of the best performance of its actuating modes. Finally, for the mechanism at hand, we introduced equivalent bounds for the condition number and the transmission angle, which allow us to conclude that the two indices give similar results.

References

1. Alba-Gomez, O., Wenger, P. and Pamanes, A. (2005), Consistent kinetostatic indices for planar 3-DOF parallel manipulators, application to the optimal kinematic inversion, in *Proceedings of the ASME 2005 IDETC/CIE Conference*.
2. Arakelian, V., Briot, S. and Glazunov, V. (2007), Increase of singularity-free zones in the workspace of parallel manipulators using mechanisms of variable structure, *Mechanism and Machine Theory*, Available online at www.sciencedirect.com.
3. Balli, S. and Chand, S. (2002), Transmission angle in mechanisms, *Mechanism and Machine Theory* **37**(2), 175–195.
4. Chablat, D. and Wenger, P. (1998), Working modes and aspects in fully-parallel manipulator, in *Proceeding IEEE International Conference on Robotics and Automation*, pp. 1964–1969, May.
5. Chablat, D., Wenger, P., Caro, S. and Angeles, J. (2002), The isoconditioning loci of planar three-DOF parallel manipulators, in *Proceedings DETC ASME*, Montreal, Canada.
6. Chablat, D., Wenger, P., Majou, F. and Merlet, J.P. (2004), An interval analysis based study for the design and the comparison of 3-DOF parallel kinematic machines, *International Journal of Robotics Research* **23**(6), 615–624.
7. Chablat, D. and Wenger, P. (2004), The kinematic analysis of a symmetrical three-degree-of-freedom planar parallel manipulator, *Proceedings CISM-IFToMM Symposium on Robot Design, Dynamics and Control*, Montreal.
8. Chen, C. and Angeles, J. (2005), A generalized transmission index for spatial linkages, in *Proceedings of the ASME 2005 IDETC/CIE Conference*.
9. Golub, G.H. and Van Loan, C.F. (1989), *Matrix Computations*, The Johns Hopkins University Press, Baltimore.
10. Gosselin, C. and Angeles, J. (1991), A global performance index for the kinematic optimization of robotic manipulators, *ASME Journal of Mechanical Design* **113**, 220–226.
11. Li, Z. (1990), Geometrical consideration of robot kinematics, *The International Journal of Robotics and Automation* **5**(3), 139–145.
12. Merlet, J-P. (2006), *Parallel Robots*, Springer, Dordrecht.
13. Paden, B. and Sastry, S. (1988), Optimal kinematic design of 6R manipulator, *The International Journal of Robotics Research* **7**(2), 43–61.
14. Ranjbaran, F., Angeles, J., Gonzalez-Palacios, M.A. and Patel, R.V. (1995), The mechanical design of a seven-axes manipulator with kinematic isotropy, *ASME Journal of Intelligent and Robotic Systems* **14**(1), 21–41.
15. Theingin, Chen, I.-M., Angeles, J. and Li, C. (2007), Management of parallel-manipulator singularities using joint-coupling, *Advanced Robotics* **21**(5/6), 583–600.

Results on Planar Parallel Manipulators with Cylindrical Singularity Surface

G. Nawratil

Institute of Discrete Mathematics and Geometry, Vienna University of Technology, 1040 Vienna, Austria, e-mail: nawratil@geometrie.tuwien.ac.at

Abstract. In this article we give first results on Stewart Gough Platforms with planar base and platform, whose singularity set for any orientation of the platform is a cylindrical surface with rulings parallel to a given fixed direction p in the space of translations. In this case the singularity set can easily be visualized as curve by choosing p as projection direction. Moreover the computation of singularity free zones reduces to a 5-dimensional task. We prove that there do not exist non-architecturally singular Stewart Gough Platforms with planar base and platform and no four anchor points collinear which possess such a singularity surface.

Key words: Stewart Gough Platform, planar parallel manipulator, cylindrical singularity surface, architecture singular manipulators.

1 Introduction

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

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

the translation vector $\mathbf{t} := (t_1, t_2, t_3)^T$ and $K := e_0^2+e_1^2+e_2^2+e_3^2$. Moreover it should be noted that K is used as homogenizing factor whenever it is suitable.

It is well known (see e.g. [5]) that the set of singular configurations is given by $Q := \det(\mathbf{Q}) = 0$, where the ith row of the 6×6 matrix $\mathbf{Q}$ equals the Plücker coordinates $(\mathbf{l}_i, \widehat{\mathbf{l}}_i) := (\mathbf{R}\cdot\mathbf{m}_i + \mathbf{t} - K\mathbf{M}_i, \mathbf{M}_i \times \mathbf{l}_i)$ of the carrier line of the ith leg.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 321–330.

As we consider only manipulators with planar platform we may suppose $c_i = 0$ for $i = 1, \ldots, 6$. We set up the planar base in a more general position as

$$C_1 = 0, \quad C_i = \left[C_2(B_3A_i - A_3B_i) + A_2C_3B_i\right]/(A_2B_3) \quad \text{for} \quad i = 4, 5, 6. \tag{2}$$

Moreover it was proven by Karger in [2] that for planar parallel manipulator with no four points on a line we can assume $A_1 = B_1 = B_2 = a_1 = b_1 = b_2 = 0$ and $A_2B_3B_4B_5a_2(a_4 - a_3)coll(3, 4, 5) \neq 0$ with

$$coll(i, j, k) := a_i(b_j - b_k) + a_j(b_k - b_i) + a_k(b_i - b_j). \tag{3}$$

Note that $coll(i, j, k) = 0$ characterizes collinear platform anchor points $\mathbf{m}_i, \mathbf{m}_j$ and $\mathbf{m}_k$.

2 Preliminary Considerations

The set of Stewart Gough Platforms whose singularity set for any orientation is a cylindrical surface with rulings parallel to a given direction p also contains the set of architecture singular manipulators. This is due to the fact that the singularity surface of these manipulators equals the whole space of translations for any orientation.

It can easily be seen from the following example that the above two sets are distinct:

The non-planar manipulator determined by $\mathbf{m}_1 = \mathbf{m}_2$, $\mathbf{m}_3 = \mathbf{m}_4$, $\mathbf{m}_5 = \mathbf{m}_6$ and $\overline{\mathbf{M}_1\mathbf{M}_2} \parallel \overline{\mathbf{M}_3\mathbf{M}_4} \parallel \overline{\mathbf{M}_5\mathbf{M}_6} \parallel p$ has for any orientation of the platform a cylindrical surface with rulings parallel to the direction p without being architecturally singular (see Figure 1). This manipulator is only in a singular configuration iff the three planes $[\mathbf{M}_1, \mathbf{M}_2, \mathbf{m}_1]$, $[\mathbf{M}_3, \mathbf{M}_4, \mathbf{m}_3]$ and $[\mathbf{M}_5, \mathbf{M}_6, \mathbf{m}_5]$ have a common intersection line.

As the direct kinematics of this manipulator can be put down to that of a 3-dof RPR parallel manipulator, a rational parametrization of its singularity surface according to Husty et al. [1] can be given. The singularity surface is a quadratic cylinder due to the (singular) affine correspondence between the base and the platform (cf. [3]).

Moreover, if $\mathbf{M}_1, \ldots, \mathbf{M}_6$ are coplanar we get an example for a planar parallel manipulator with this property. Now the question arises, if there also exist non-architecturally singular planar manipulators with no four anchor points on a line possessing such a singularity surface. In the following section we prove that such manipulators do not exist.

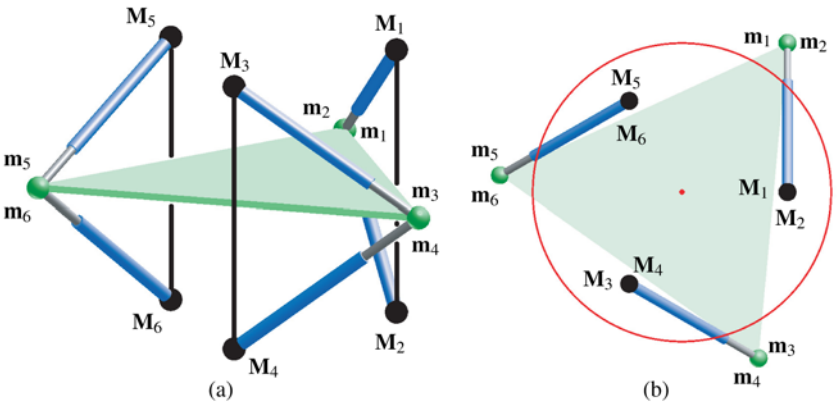


Fig. 1 Non-planar manipulator with cylindrical singularity surface: (a) Axonometric view. (b) Projection direction is p: The singularity surface (with respect to the barycenter of the platform) is displayed as conic.

3 The Main Theorem and Its Proof

Theorem. *The set of planar parallel manipulators with no four anchor points on a line which possess a cylindrical singularity surface with rulings parallel to a given fixed direction p for any orientation of the platform equals the set of planar architecture singular manipulators (with no four anchor points on a line).*

The analytical proof of this theorem is based on the following idea: We choose an Cartesian frame in the base such that one axis t_i is parallel to the given direction p. Then $Q := \det(\mathbf{Q}) = 0$ must be independent of t_i for all $e_0, \ldots, e_3, t_j, t_k$ with $j \neq k \neq i \neq j$. Our proof is based on the resulting equations and [2, theorem 1].

We have to distinguish between two cases given in the following subsections.

3.1 Base Is Not Parallel to p

The proof of the case where the base is orthogonal to p is hidden in the proof of theorem 1 given by Karger [2]. Therefore this case which corresponds to $C_2 = C_3 = 0$ by eliminating t_3 from Q does not need be discussed.

For all other directions we start analogously to Karger's proof by setting $t_1 = t_2 = 0$ and performing the same elementary operations with the matrix $\mathbf{Q}$ as described on page 1154 of the cited paper. Then the last row of $\mathbf{Q}$ is of the form

$$(r_{11}K_1 + r_{12}A_2K_2, r_{21}K_1 + r_{22}A_2K_2, r_{31}K_1 + r_{32}A_2K_2, r_{21}C_2K_3 + r_{22}C_2K_4, \\ r_{31}A_2K_3 + r_{32}A_2K_4 - r_{11}C_2K_3 - r_{12}C_2K_4, -r_{21}A_2K_3 - r_{22}A_2K_4)D^{-1} \tag{4}$$

with $D := A_2B_3B_4B_5coll(3,4,5)$ and r_{ij} of Eq. (1). It should be noted that $K_1 = K_2 = K_3 = K_4 = 0$ are the four conditions given in [2] which are satisfied iff a planar manipulator (with no four points on a line) is architecturally singular.

Now Q can be written as

$$Q = A_2^2(r_{11}r_{22} - r_{12}r_{21})Q_3t_3^3 + A_2B_3Q_2t_3^2 + Q_1t_3 + Q_0. \tag{5}$$

With the coefficients Q_1, Q_2 and Q_3 the steps (**a**) and (**b**) on p. 1155 of [2] can be done one by one. The steps (**c**) and (**d**) are different and therefore given here.

Step (c) $K_1 = 0$, $K_2 = 0$, $K_4 \neq 0$

After substituting Euler parameters e_i into Q_1 and Q_2, we can factor out K of Q_i $(i = 1, 2)$; let us call the remaining coefficient again Q_i $(i = 1, 2)$.

(A) Let $B_4b_3 - B_3b_4 \neq 0$.
From the coefficient of e_0^2 in Q_2 we express A_5. Denote the coefficients of $e_0^5e_1$ and $e_2^5e_3$ of Q_2 by v_1 and v_2, respectively, and express a_5 from v_1+v_2. Now $B_3-b_3 = 0$ or $B_4-b_4 = 0$ must be different from zero; we may suppose $B_3-b_3 \neq 0$. Therefore we can express a_4 from $v_1 = 0$. The coefficient $e_0^1e_1^5$ of Q_2 yields $a_3 = a_2A_3/A_2$. Now the coefficients of $e_0^4e_2e_3$ and $e_0e_1e_2^4$ of Q_2 can only vanish (without contradiction) if $K_3 = 0$. The coefficient $e_0^3e_3^3$ of Q_2 yields $C_3 = C_2A_3/A_2$. Finally we get as coefficient of $e_0^3e_1^2e_2$ of Q_2 the expression $A_2B_3K_4a_2coll(3,4,5)$, a contradiction.

(B) Let $B_4b_3 - B_3b_4 = 0$, i.e. $b_4 = b_3B_4/B_3$.
From $v_1 + v_2 = 0$ we get $b_5 = b_3B_5/B_3$. Let us denote the coefficients of $e_0e_1^5$, $e_1e_3^5$, $e_0^5e_2$ of Q_2 by v_3, v_4, v_5. From $v_2 - v_3 = 0$ and $v_4 + v_5 = 0$ we compute A_4 and A_5. Now Q_1 factorizes into $KA_2F_1[16]F_2[2316]/(a_2B_3)$, where the number in the square brackets denotes the number of additive factors in the expression.

• ad $F_1 = 0$: From the coefficient of e_0e_3 we express $a_3 = a_2A_3/A_2$. If we denote the coefficients of e_i^2 by q_i, the sum $q_0 + q_1 + q_2 + q_3$ yields A_2B_3, a contradiction.

• ad $F_2 = 0$: We denote the coefficients of $e_0^3e_1$, $e_0e_1^3$, $e_0^2e_1e_3$ and $e_1e_2^2e_3$ by p_1, p_2, p_3 and p_4. The equations $p_1 + p_2 = 0$ and $p_3 - p_4 = 0$ can only vanish (without contradiction) if $K_3 = 0$ or $C_2 = C_3 = 0$. As the later case can be neglected we set K_3 equal to zero. The equation $w_1 - w_2 = 0$ vanishes (without contradiction) for $C_2 = 0$ or $F_3[12] = 0$, where w_1 and w_2 are the coefficients of e_0^4 and e_1^4. If $C_2 = 0$ we obtain $C_3 = 0$ from $w_1 + w_2 = 0$.

(i) Let $n := B_3B_4a_5(a_4 - a_3) + B_3B_5a_4(a_3 - a_5) + B_4B_5a_3(a_5 - a_4) \neq 0$. Then we can express $a_2 = d/n$ from $F_3[12] = 0$ with

$$d := a_3^2B_4B_5(a_5 - a_4) + a_4^2B_3B_5(a_3 - a_5) + a_5^2B_3B_4(a_4 - a_3). \tag{6}$$

From the coefficient of $e_0^3 e_1$ we compute C_2. Plugging the obtained expression into F_2 yields $K_4 B_4 B_5 b_3 F_4[80] F_5[96]/(A_2 dn)$. Now the coefficient of $e_0 e_1$ of $F_4[80]$ as well as the one of $F_5[96]$ yields $A_2 B_3 d = 0$, a contradiction.

(ii) Let $n = 0$. For $h := B_3 a_4 (a_5 - a_3) + B_4 a_3 (a_4 - a_5) \neq 0$ we can compute B_5 from $n = 0$. Substituting this into $F_3[12]$ yields the contradiction. If $h = 0$ we can compute a_5 from this equation under the assumption $B_3 a_4 - B_4 a_3 \neq 0$. Substituting this into $n = 0$ yields

$$a_4 a_3 B_4 B_3 (a_4 - a_3)(B_3 - B_4)/(B_3 a_4 - B_4 a_3) = 0. \tag{7}$$

Now we have to distinguish between the following two cases:

- $a_3 = 0$ or $a_4 = 0$: Without loss of generality we assume $a_3 = 0$ and $a_4 \neq 0$. Now the coefficient of $e_1^2 e_2 e_3$ of F_2 can only vanish (without contradiction) if $A_2 B_3 a_4 - B_3 a_2 A_2 + B_4 a_2 A_3 = 0$ or $C_2 = 0$. For the later we obtain $C_3 = 0$ from $w_1 = 0$. Therefore we set $a_4 = a_2 (B_3 A_2 - B_4 A_3)/(A_2 B_3)$ and substitute this into $w_1 = 0$. This equation can only vanish (without contradiction) for $C_3 = 0$. The coefficient of $e_0^2 e_1^2$ of F_2 yields the contradiction.
- $B_3 - B_4 = 0$: Substitution $B_3 = B_4$ into $F_3[12]$ yields $B_4 B_5 a_3 a_4 (a_3 - a_4)$ and therefore the above case; i.e. $a_3 = 0$ or $a_4 = 0$.

The last missing case is $B_3 a_4 - B_4 a_3 = 0$. Plugging $a_3 = a_4 B_3/B_4$ into $h = 0$ yields the contradiction. This finishes step (c).

Step (d) $K_1 = 0$, $K_2 = 0$, $K_4 = 0$, $K_3 \neq 0$

(A) Let $B_3 b_5 - b_3 B_5 \neq 0$.
We compute the coefficients l_i of $e_0^5 e_2$, $e_0 e_2^5$, $e_1^5 e_3$ and $e_1 e_3^5$ of Q_2 which are of the form:

$$l_1 = A_2 B_3 K_3 (A_2 - a_2) F_6[12], \qquad l_2 = A_2 B_3 K_3 (A_2 + a_2) F_7[12], \tag{8}$$

$$l_3 = A_2 B_3 K_3 (A_2 - a_2) F_7[12], \qquad l_4 = A_2 B_3 K_3 (A_2 + a_2) F_6[12]. \tag{9}$$

The equations $A_2 - a_2 = 0$ and $A_2 + a_2 = 0$ yield a contradiction.

(i) If we assume $b_4 \neq 0$ we can compute a_4 and A_4 from $F_6[12] = 0$ and $F_7[12] = 0$. Now the sum of the coefficients of $e_0^3 e_3^3$ and $e_1^3 e_2^3$ of Q_2 yield $A_2 C_2 B_3 K_3 coll(3, 4, 5)$ which implies $C_2 = 0$. The sum of the coefficients of $e_0^4 e_3^2$ and $e_1^2 e_2^4$ of Q_2 yield $A_2 C_3 a_2 K_3 coll(3, 4, 5)$ and therefore $C_3 = 0$.

(ii) If $b_4 = 0$ we proceed similar and compute from $F_6 = 0$ and $F_7 = 0$ the unknowns a_4 and A_3. By performing the same steps as above we also obtain $C_2 = C_3 = 0$.

(B) Let $B_3 b_5 - b_3 B_5 = 0$, e.i. $b_5 = b_3 B_5/B_3$.
In this case we look at $l_1 + l_3$ and $l_1 + l_4$ which are of the form:

$$l_1 + l_3 = C_2 K_3 b_3 F_8[12] \quad \text{and} \quad l_1 + l_4 = b_3^2 K_3 (A_3 C_2 - A_2 C_3) F_9[12]/B_3^2, \quad (10)$$

respectively. Moreover the linear combinations $m_1 - m_2$ and $m_1 + m_3$ are of the form

$$m_1 + m_3 = C_2 K_3 b_3^2 F_9[12]/B_3 \quad \text{and} \quad m_1 - m_2 = b_3 K_3 (A_3 C_2 - A_2 C_3) F_8[12]/B_3, \quad (11)$$

where m_1, m_2, m_3 are the coefficients of $e_0^2 e_3^4$, $e_1^2 e_2^4$ and $e_1^4 e_2^2$ of Q_2. As $C_2 = C_3 = 0$ can be neglected we set $F_8[12]$ and $F_9[12]$ equal to zero and compute A_4 and A_5 from it. In the next step we factorize Q_1 which yields $A_2 b_3 B_4 B_5 K_3 K F_1[16] F_{10}[144] coll(3, 4, 5)/(a_2 B_3)$.

(i) In step (c) it was already shown that $F_1[16] = 0$ yields a contradiction.

(ii) Therefore we proceed by computing the sum of the coefficient of $e_0^3 e_1$ and $e_0 e_1^3$ of $F_{10}[144]$ which results in $a_2 B_3^2 C_2$. With $C_2 = 0$ the difference of the coefficient of $e_0^3 e_2$ and $e_0 e_2^3$ of $F_{10}[144]$ yield $a_2 A_2 B_3^2 C_3$ and therefore $C_3 = 0$, which finishes this part.

3.2 Base Is Parallel to p

In this case we take as translation vector $\mathbf{t} := (\cos\varphi t_1 - \sin\varphi t_2, \sin\varphi t_1 + \cos\varphi t_2, t_3)^T$ and set $C_2 = C_3 = 0$. After performing again the same elementary operations with the matrix $\mathbf{Q}$ as above and replacing the sixth row by Eq. (4), we have to distinguish between the following two cases.

3.2.1 $\mathbf{M_1 M_2}$ Is Parallel to *p*

If we set $\varphi = 0$ the t_1 axis is parallel to p ($\Rightarrow Q$ must be independent of t_1). We denote the coefficients of $t_1^i t_2^j t_3^k$ from Q by $Q^{i,j,k}$. From $Q^{1,0,1}$ we can factor out K and from $Q^{1,0,0}$ we can even factor out K^2. We denote the coefficient of $e_0^a e_1^b e_2^c e_3^d$ of $Q^{i,j,k}$ by $P^{i,j,k}_{a,b,c,d}$ and compute

$$P^{1,0,1}_{4,1,1,0} - P^{1,0,1}_{1,4,0,1} - P^{1,0,1}_{1,0,4,1} + P^{1,0,1}_{0,1,1,4} = K_1 B_3 B_4 B_5 coll(3, 4, 5) \quad (12)$$

$$P^{1,0,1}_{0,2,2,2} + P^{1,0,1}_{2,0,2,2} - P^{1,0,1}_{2,2,0,2} - P^{1,0,1}_{2,2,2,0} = K_2 A_2 B_3 B_4 B_5 coll(3, 4, 5) \quad (13)$$

which yields $K_1 = K_2 = 0$. Now we consider

$$P^{1,0,0}_{3,1,2,0} - P^{1,0,0}_{2,0,3,1} - P^{1,0,0}_{1,3,0,2} + P^{1,0,0}_{0,2,1,3} = K_3 a_2 B_3 B_4 B_5 coll(3, 4, 5) \quad (14)$$

$$P^{1,0,0}_{3,2,1,0} - P^{1,0,0}_{2,3,0,1} - P^{1,0,0}_{1,0,3,2} + P^{1,0,0}_{0,1,2,3} = K_4 a_2 B_3 B_4 B_5 coll(3, 4, 5) \quad (15)$$

which finishes this part of the proof.

3.2.2 $\mathbf{M_1M_2}$ Is Not Parallel to p

As this part of the proof is too long to be presented here in its full length we refer to the corresponding technical report [7]. In the following we only give a sketch of the proof as well as the two special solutions S_1 and S_2 which cause difficulties.

First of all we can assume $\sin\varphi \neq 0$ if we eliminate t_1 from Q. If we assume additionally $K_2 = 0$, one can show in a similar way as above that also $K_1 = K_3 = K_4 = 0$ must hold. But if we assume $K_2 \neq 0$ there exist two solutions which fulfill all equations resulting from the coefficients of t_1 of Q without contradicting

$$A_2B_3B_4B_5a_2(a_4 - a_3)coll(3,4,5)K_2\sin\varphi \neq 0. \tag{16}$$

These two solutions S_1 and S_2 are given by

$$S_1: \quad A_i = B_i\cot\varphi,\ A_j = B_j\cot\varphi,\ A_k = A_2 + B_k\cot\varphi, \tag{17}$$

$$b_k = 0,\ a_2 = a_k,\ a_i = K_1b_i/(K_2A_2),\ a_j = K_1b_j/(K_2A_2), \tag{18}$$

$$K_3 = 0 \quad \text{and} \quad K_4 = 0 \tag{19}$$

and

$$S_2: \quad A_i = A_2 + B_i\cot\varphi,\ A_j = A_2 + B_j\cot\varphi,\ A_k = B_k\cot\varphi, \tag{20}$$

$$a_i = a_2 + b_iK_3/K_4,\ a_j = a_2 + b_jK_3/K_4,\ a_k = b_k = 0, \tag{21}$$

$$A_2K_2 + K_4 = 0 \quad \text{and} \quad K_1 + K_3 = 0 \tag{22}$$

for $i, j, k \in \{3,4,5\}$ and $i \neq j \neq k \neq i$. In the following we show that these solutions also imply contradictions for the choice of $\mathbf{M}_6$ and $\mathbf{m}_6$, respectively.

First of all we can set $A_2 = 1$ due to $A_2 \neq 0$. Then we replace K_i in Eqs. (22) and (19) by the expressions given in [2, eq. (4)]. If we plug now the expressions of Eqs. (17) and (18) into the resulting equations of Eq. (19) we get

$$K_3 = (A_6 - B_6\cot\varphi)(a_k - a_6) \quad \text{and} \quad K_4 = (A_6 - B_6\cot\varphi)b_6, \tag{23}$$

respectively. The solution $a_6 = a_k$ and $b_6 = 0$ contradicts $K_2 \neq 0$. If $A_6 = B_6\cot\varphi$ the four base anchor points $\mathbf{M}_1, \mathbf{M}_i, \mathbf{M}_j$ and $\mathbf{M}_6$ are collinear.

For the second solution we proceed similarly, i.e. we plug the expressions of Eqs. (20) and (21) into the equations of (22). We end up with

$$K_1 + K_3 = (1 - A_6 + B_6\cot\varphi)a_6 \quad \text{and} \quad K_2 + K_4 = (1 - A_6 + B_6\cot\varphi)b_6. \tag{24}$$

The solution $a_6 = b_6 = 0$ again contradicts $K_2 \neq 0$. The other solution $A_6 = 1 + B_6\cot\varphi$ implies the collinearity of the four base anchor points $\mathbf{M}_2, \mathbf{M}_i, \mathbf{M}_j$ and $\mathbf{M}_6$. This finishes the proof of the given theorem. □

4 A Further Example

The two solutions S_1 and S_2 imply a further example for an planar parallel manipulator with cylindrical singularity surface beside the one given in Section 2. The computation of the corresponding manipulator can be done as follows:

S_1: If we set $A_6 = B_6 \cot\varphi$ there are two conditions left, which derive from [2, eq. (4)]. Solving these two equations for the variables K_1 and K_2 yield:

$$K_1 = a_6 s/(b_6 B_i B_j (b_i - b_j)) \quad \text{and} \quad K_2 = s/(B_i B_j (b_i - b_j)) \tag{25}$$

with

$$s := B_i B_j b_6 (b_i - b_j) + B_j B_6 b_i (b_j - b_6) + B_i B_6 b_j (b_6 - b_i). \tag{26}$$

As special case we obtain

$$a_6 = b_6 = 0 \quad \text{and} \quad K_2 = b_i b_j B_6 (B_j - B_i))/(B_i B_j (b_i - b_j)). \tag{27}$$

S_2: For $A_6 = 1 + B_6 \cot\varphi$ analogous computations yield

$$K_1 = s(a_6 - a_2)/(b_6 B_i B_j (b_i - b_j)), \quad K_2 = s/(B_i B_j (b_i - b_j)), \tag{28}$$

with s of Eq. (26). Here the special case is given by

$$a_6 = a_2,\ b_6 = 0 \quad \text{and} \quad K_2 = b_i b_j B_6 (B_j - B_i)/(B_i B_j (b_i - b_j)). \tag{29}$$

It should be noted that we can assume $B_i B_j (b_i - b_j) \neq 0$, otherwise D of Eq. (4) is equal to zero which is forbidden (division by zero).

Moreover it should be mentioned that if $s = 0$ holds the manipulator is architecturally singular due to $K_1 = K_2 = K_3 = K_4 = 0$. The condition $s = 0$ expresses that the cross ratio of the base anchor points $\mathbf{M}_x, \mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_6$ and of the corresponding platform anchor points $\mathbf{m}_x, \mathbf{m}_i, \mathbf{m}_j, \mathbf{m}_6$ is the same.

In the architecturally singular case the carrier lines of the involved four legs belong to a ruled quadric, which can also degenerate into two planes (cf. 8th entry in the list of architecture singular Stewart Gough Platforms given by Karger in [4, theorem 3]).

It follows immediately from the expressions of a_i and a_j given in Eqs. (18) and (21), respectively, that the platform anchor points $\mathbf{m}_i, \mathbf{m}_j$ and $\mathbf{m}_x$ of solution S_x ($x = 1, 2$) are collinear. If we plug now the obtained expressions for K_1 and K_2 (given in Eqs. (25) and (28), respectively) of solution S_x into a_i and a_j, we can see that also $\mathbf{m}_6$ is located on the line spanned by $\mathbf{m}_i, \mathbf{m}_j$ and $\mathbf{m}_x$. For both special cases (given in Eqs. (27) and (29), respectively) this is trivially true due to $\mathbf{m}_x = \mathbf{m}_6$.

Therefore the geometric properties of the planar parallel manipulator with cylindrical singularity surface corresponding with solution S_1 and S_2 can be summarized as follows:

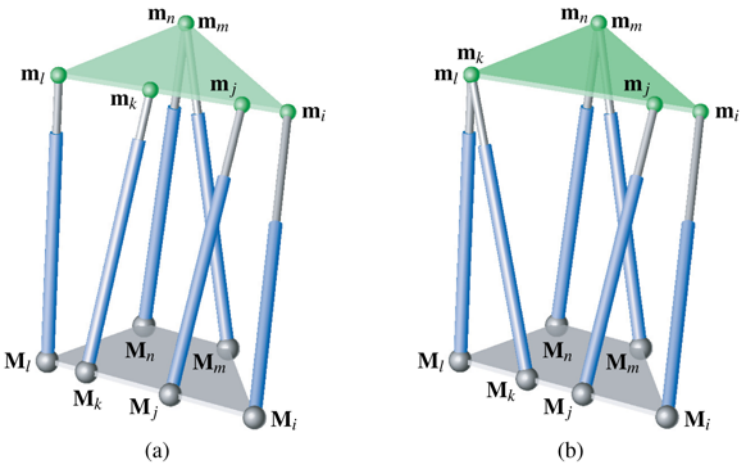


Fig. 2 Planar parallel manipulator with cylindrical singularity surface: (a) General case. (b) Special case.

(i) $\mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_k, \mathbf{M}_l$ are collinear,
(ii) $\mathbf{m}_i, \mathbf{m}_j, \mathbf{m}_k, \mathbf{m}_l$ are collinear,
(iii) $\overline{\mathbf{M}_m\mathbf{M}_n} \parallel \overline{\mathbf{M}_i\mathbf{M}_j} \parallel p$,
(iv) and $\mathbf{m}_m = \mathbf{m}_n$.

For the special cases we have the additional condition $\mathbf{m}_k = \mathbf{m}_l$. The manipulator and its special case is given in Figures 2a and 2b, respectively.

This manipulator is in a singular position iff $\mathbf{m}_m = \mathbf{m}_n$ lies in the carrier plane of the base or if the carrier lines of $\mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_k, \mathbf{M}_l$ and $\mathbf{m}_i, \mathbf{m}_j, \mathbf{m}_k, \mathbf{m}_l$ intersect each other. Therefore the quadratic singularity surface always splits into two planes (parallel to p).

5 Remarks

Remark 1. The known examples of planar parallel manipulators with a cylindrical singularity surface (given in Sections 2 and 4) raise the question if such manipulators with a cubic singularity surface exist. A complete list of planar parallel manipulators with a cylindrical singularity surface is in preparation [8].

Remark 2. It should be noted that the proof of the second direction ($\det(\mathbf{Q}) = 0 \Rightarrow K_1 = K_2 = K_3 = K_4 = 0$) of theorem 1 given by Karger [2] can be replaced just by four equations, namely by Eqs. (12–15). As the four conditions $K_1 = K_2 = K_3 = K_4 = 0$ are expressed by not more than four equations, we have found the shortest possible analytical proof of the second direction of the cited theorem.

Remark 3. Röschel and Mick proved in [6, 9] that planar parallel manipulators are architecturally singular iff $\{\mathbf{M}_i, \mathbf{m}_i\}$ for $(i = 1, \ldots, 6)$ are four-fold conjugate pairs

of points with respect to a 3-dimensional linear manifold of correlations or one of the two sets $\{\mathbf{M}_i\}$ and $\{\mathbf{m}_i\}$ is situated on a line.

It would be nice to have such a geometric proof for the given theorem too. It might be possible to prove in a similar way to [6, 9] that planar parallel manipulators with no four points on a line and a cylindrical singularity surface must consist of four-fold conjugate pairs of anchor points.

6 Conclusion

We presented first results on planar parallel manipulators whose singularity set for any orientation is a cylindrical surface with rulings parallel to a given fixed direction p in the space of translations. We proved that there do not exist non-architecturally singular Stewart Gough Platforms with planar base and platform and no four anchor points collinear which possess such a singularity surface.

As by-product of our proof we gave the shortest possible analytical proof for the second direction ($\det(\mathbf{Q}) = 0 \Rightarrow K_1 = K_2 = K_3 = K_4 = 0$) of theorem 1 given by Karger [2]. Moreover, we presented two examples of planar manipulators with cylindrical singularity surface. A complete list of such planar parallel manipulators is in preparation [8].

References

1. Husty, M.L., Hayes, M.J.D. and Loibnegger, H., The general singularity surface of planar three-legged platforms, in *Advances in Multibody Systems and Mechantronics*, A. Kecskemethy (Ed.), Duisburg, Germany, pp. 203–214 (1999).
2. Karger, A., Architecture singular planar parallel manipulators, *Mechanism and Machine Theory* **38**(11), 1149–1164 (2003).
3. Karger, A., Stewart–Gough platforms with simple singularity surface, in *Advances in Robot Kinematics: Mechanisms and Motion*, J. Lenarčič, B. Roth (Eds.), pp. 247–254, Springer (2006).
4. Karger, A., Architecturally singular non-planar parallel manipulators, *Mechanism and Machine Theory* **43**(3), 335–346 (2008).
5. Merlet, J.-P., Singular configurations of parallel manipulators and Grassmann geometry, *International Journal of Robotics Research* **8**(5), 45–56 (1992).
6. Mick, S. and Röschel, O., Geometry & architecturally shaky platforms, in *Advances in Robot Kinematics: Analysis and Control*, J. Lenarčič, M.L. Husty (Eds.), pp. 455–464, Kluwer (1998).
7. Nawratil, G., Main theorem on planar parallel manipulators with cylindrical singularity surface, Technical Report No. 187, Geometry Preprint Series, Vienna University of Technology (2008).
8. Nawratil, G., All planar parallel manipulators with cylindrical singularity surface, in preparation.
9. Röschel, O. and Mick, S., Characterisation of architecturally shaky platforms, in *Advances in Robot Kinematics: Analysis and Control*, J. Lenarčič, M.L. Husty (Eds.), pp. 465–474, Kluwer (1998).

Stiffness Matrix of Compliant Parallel Mechanisms

C. Quennouelle and C.M. Gosselin

Laboratoire de Robotique de l'Université Laval, Québec, Canada G1K 7P4,
e-mail: {cyril.quennouelle.1, gosselin}@ulaval.ca

Abstract. Starting from the definition of a stiffness matrix, the authors present the Cartesian stiffness matrix of parallel compliant mechanisms. The proposed formulation is more general than any other stiffness matrix found in the literature since it can take into account the stiffness of the passive joints and remains valid for large displacements. Then, the conservative property, the validity,and the positive definiteness of this matrix are discussed.

Key words: stiffness matrix, compliant parallel mechanisms, kinemato-static model.

1 Introduction

The stiffness matrix of a mechanism is defined as the Hessian matrix of a potential. For example, the Cartesian stiffness matrix is the square matrix of second-order partial derivatives of potential ξ_f associated with wrench $\mathbf{f}$ with respect to the vector of Cartesian coordinates, noted $\mathbf{x}$:

$$\mathbf{K}_C = \frac{\partial^2 \xi_f}{\partial \mathbf{x}^2}. \tag{1}$$

Thus by definition, a stiffness matrix is a symmetric matrix [3]. A stiffness matrix is also conservative [1]. And since the Hessian matrix of a potential is used to determine the stability of an equilibrium [5], a stiffness matrix can be either positive-definite or negative-definite.

In this paper, a stiffness matrix that considers the external loads, the changes of the geometry of the mechanism and the stiffness of any joint – even the passive ones – is presented. The kinematic model of a parallel mechanism that takes into account the passive joints is first introduced. Then, expressions for the potential energy are derived in order to obtain a general form of the Cartesian stiffness matrix of a compliant mechanism. The correctness and the properties of this matrix are then discussed and applied in a simple parallel mechanism.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 331–341.

2 Model of a Parallel Mechanism

2.1 Geometric Constraints

In a parallel mechanism, some geometrical constraints between the joint coordinates corresponding to the closure of loops formed by legs must always be satisfied. These constraints are written as $\mathcal{K}(\boldsymbol{\theta}) = \mathbf{0}$, where $\boldsymbol{\theta}$ is the joint coordinate vector of the mechanism.

Therefore, a vector of generalized coordinates $\boldsymbol{\chi}$ is defined such that $\boldsymbol{\lambda}$, the vector of the kinematically constrained coordinates and $\boldsymbol{\theta}$, the complete joint coordinate vector of the mechanism, always satisfy the geometric constraints. One has:

$$\boldsymbol{\lambda} = \boldsymbol{\lambda}(\boldsymbol{\chi}) \quad \text{and} \quad \boldsymbol{\theta} = \boldsymbol{\theta}(\boldsymbol{\chi}) = \begin{bmatrix} \boldsymbol{\chi} \\ \boldsymbol{\lambda} \end{bmatrix}, \tag{2}$$

where $\boldsymbol{\lambda} = [\lambda_1, \cdots, \lambda_c]^T - c$ being the number of constrained coordinates – and $\boldsymbol{\theta} = [\theta_1, \cdots, \theta_m]^T$ with m the number of joints in the mechanism and θ_k the coordinate of the kth joint.

2.2 Kinematic Constraints

The variation of the kinematically constrained (dependent) joint coordinates is described by a matrix $\mathbf{G}$ and a matrix $\mathbf{R}$ defined as

$$\mathbf{G} = \frac{d\boldsymbol{\lambda}}{d\boldsymbol{\chi}} \quad \text{and} \quad \mathbf{R} = \frac{d\boldsymbol{\theta}}{d\boldsymbol{\chi}} = \begin{bmatrix} \mathbf{1}_l \\ \mathbf{G} \end{bmatrix}, \tag{3}$$

where $\mathbf{1}_l$ stands for the $l \times l$ identity matrix. The relations between the variation of the joint coordinates and the variation of the generalized coordinates are expressed as $d\boldsymbol{\lambda} = \mathbf{G} d\boldsymbol{\chi}$ and $d\boldsymbol{\theta} = \mathbf{R} d\boldsymbol{\chi}$.

2.3 Kinematic Model

2.3.1 Pose of the Platform

Represented by a vector $\mathbf{x}$, it is defined as the average pose of the end-effector of all legs of the mechanism, namely

$$\mathbf{x} = \frac{1}{n} \sum_{i=a}^{n} \mathbf{x}_i, \quad i \in \{a, \cdots, n\}, \tag{4}$$

where $\mathbf{x}_i = \left[\mathbf{c}_i^T, \mathbf{q}_i^T\right]^T$ is the pose vector of the ith leg and where $\mathbf{c}_i$ is the position vector of a chosen point on the platform while $\mathbf{q}_i$ is a quaternion vector describing the orientation of the platform. All legs are indexed from a to n.

2.3.2 Jacobian Matrix $\mathbf{J}_\theta$

The Jacobian matrix $\mathbf{J}_\theta$ of a parallel mechanism in which all joints – even the passive ones – are considered is written as

$$\mathbf{J}_\theta = \frac{\partial \mathbf{x}}{\partial \boldsymbol{\theta}} = \frac{1}{n} \sum_{i=a}^{n} \frac{\partial \mathbf{x}_i}{\partial \boldsymbol{\theta}}. \tag{5}$$

2.3.3 Jacobian Matrix J

In this Jacobian matrix, only the generalized coordinates are considered. It is defined as

$$\mathbf{J} = \frac{\partial \mathbf{x}}{\partial \boldsymbol{\chi}} = \frac{\partial \mathbf{x}}{\partial \boldsymbol{\theta}} \frac{\partial \boldsymbol{\theta}}{\partial \boldsymbol{\chi}} = \mathbf{J}_\theta \mathbf{R}. \tag{6}$$

2.3.4 Kinematic Model

Thus, the kinematic model of the complete mechanism can be written in different equivalent forms, namely

$$d\mathbf{x} = \mathbf{J}_\theta d\boldsymbol{\theta} = \mathbf{J}_\theta \mathbf{R} d\boldsymbol{\chi} = \mathbf{J} d\boldsymbol{\chi}. \tag{7}$$

2.3.5 Inverse Kinematic Model

From Eq. (7), the inverse kinematic model of the mechanism is expressed as

$$d\boldsymbol{\chi} = \mathbf{J}^{-1} d\mathbf{x}, \quad d\boldsymbol{\lambda} = \mathbf{G}\mathbf{J}^{-1} d\mathbf{x} \quad \text{and} \quad d\boldsymbol{\theta} = \mathbf{R}\mathbf{J}^{-1} d\mathbf{x}. \tag{8}$$

If the number of components in $\mathbf{x}$ is larger than six, then $\mathbf{J}^{-1}$ should be replaced by the Moore–Penrose generalized inverse.

3 Cartesian Stiffness Matrix of a Compliant Mechanism

3.1 Potential Energy of a Mechanism

The potential energy stored in the compliant joints of a compliant mechanism, noted ξ_θ, is calculated as

$$\xi_\theta = \int_{\boldsymbol{\theta}_0}^{\boldsymbol{\theta}} \boldsymbol{\tau}_\theta^T d\boldsymbol{\theta} = \int_{\boldsymbol{\chi}_0}^{\boldsymbol{\chi}} \boldsymbol{\tau}_\chi^T d\boldsymbol{\chi} + \int_{\boldsymbol{\lambda}_0}^{\boldsymbol{\lambda}} \boldsymbol{\tau}_\lambda^T d\boldsymbol{\lambda}, \tag{9}$$

where $\boldsymbol{\tau}_\theta$, $\boldsymbol{\tau}_\chi$ and $\boldsymbol{\tau}_\lambda$ are the vectors of joint torques/forces respectively associated with the joints corresponding to vectors $\boldsymbol{\theta}$, $\boldsymbol{\chi}$ or $\boldsymbol{\lambda}$ and where $\boldsymbol{\theta}_0$, $\boldsymbol{\chi}_0$ and $\boldsymbol{\lambda}_0$ correspond to the unloaded configuration of the mechanism. In the particular – but frequent – case of compliant joints with constant stiffness, ξ_θ is written as

$$\xi_\theta = \frac{1}{2}(\boldsymbol{\chi} - \boldsymbol{\chi}_0)^T \mathbf{K}_\chi (\boldsymbol{\chi} - \boldsymbol{\chi}_0) + \frac{1}{2}(\boldsymbol{\lambda} - \boldsymbol{\lambda}_0)^T \mathbf{K}_\lambda (\boldsymbol{\lambda} - \boldsymbol{\lambda}_0), \tag{10}$$

where $\mathbf{K}_\chi$ and $\mathbf{K}_\lambda$ are the (diagonal) joint stiffness matrices.

The potential energy ξ_f associated to the external wrench $\mathbf{f}$ is equal to the work provided by $\mathbf{f}$ and is defined as

$$\xi_f = \int_{\mathbf{x}_0}^{\mathbf{x}} \mathbf{f}^T d\mathbf{x}, \tag{11}$$

where $\mathbf{x}_0$ corresponds to the unloaded configuration.

The potential energy due to the external wrench ξ_f is equal – apart from a constant ξ_0 – to the energy stored in the mechanism. ($\xi_f = \xi_\theta + \xi_0$). From Eq. (8), this can be written as

$$\int_{\mathbf{x}_0}^{\mathbf{x}} \mathbf{f}^T d\mathbf{x} = \int_{\mathbf{x}_0}^{\mathbf{x}} \boldsymbol{\tau}_\chi^T \mathbf{J}^{-1} d\mathbf{x} + \int_{\mathbf{x}_0}^{\mathbf{x}} \boldsymbol{\tau}_\lambda^T \mathbf{G}\mathbf{J}^{-1} d\mathbf{x} + \xi_0. \tag{12}$$

3.2 Cartesian Static Equilibrium

Differentiating Eq. (12) with respect to the pose $\mathbf{x}$ leads to the Cartesian static equilibrium of a compliant mechanism. It is written as

$$\frac{d\xi_f}{d\mathbf{x}} = \frac{d\xi_\theta}{d\mathbf{x}} + \frac{d\xi_0}{d\mathbf{x}} \Leftrightarrow \mathbf{f} = \mathbf{J}^{-T}\boldsymbol{\tau}_\chi + \mathbf{J}^{-T}\mathbf{G}^T\boldsymbol{\tau}_\lambda. \tag{13}$$

In the most general case, the stiffness of these joints is not constant and the corresponding forces/torques are defined as

$$\begin{cases} \boldsymbol{\tau}_\chi = \int_{\boldsymbol{\chi}_0}^{\boldsymbol{\chi}} \mathbf{K}_\chi d\boldsymbol{\chi} = \int_{\mathbf{x}_0}^{\mathbf{x}} \mathbf{K}_\chi \mathbf{J}^{-1} d\mathbf{x}, \\ \boldsymbol{\tau}_\lambda = \int_{\boldsymbol{\lambda}_0}^{\boldsymbol{\lambda}} \mathbf{K}_\lambda d\boldsymbol{\lambda} = \int_{\mathbf{x}_0}^{\mathbf{x}} \mathbf{K}_\lambda \mathbf{G}\mathbf{J}^{-1} d\mathbf{x}. \end{cases} \tag{14}$$

3.3 Cartesian Stiffness Matrix

The definition of the Cartesian stiffness matrix of a mechanism is given in Eq. (1) and is equivalent to $\mathbf{K}_C = d\mathbf{f}/d\mathbf{x}$. Therefore, using Eqs. (13) and (14), it is obvious that $\mathbf{K}_C$ is not constant and depends on the stiffness of the joints and the geometric configuration of the mechanism. To obtain this function, the right-hand side of Eq. (13) is differentiated with respect to $\mathbf{x}$

$$\frac{d^2\xi_\theta}{d\mathbf{x}^2} = \frac{d}{d\mathbf{x}} \left(\mathbf{J}^{-T} \int_{\mathbf{x}_0}^{\mathbf{x}} \mathbf{K}_\chi \, \mathbf{J}^{-1} d\mathbf{x} + \mathbf{J}^{-T} \mathbf{G}^T \int_{\mathbf{x}_0}^{\mathbf{x}} \mathbf{K}_\lambda \mathbf{G}\mathbf{J}^{-1} d\mathbf{x} \right), \tag{15}$$

which leads to

$$\frac{d^2\xi_\theta}{d\mathbf{x}^2} = \mathbf{A} + \mathbf{B} + \mathbf{J}^{-T}\mathbf{K}_\chi \mathbf{J}^{-1}, \tag{16}$$

where

$$\mathbf{A} = \frac{d\mathbf{J}^{-T}}{d\mathbf{x}} (\boldsymbol{\tau}_\chi + \mathbf{G}^T \boldsymbol{\tau}_\lambda) \quad \text{and} \quad \mathbf{B} = \mathbf{J}^{-T} \frac{d\mathbf{G}^T}{d\mathbf{x}} \boldsymbol{\tau}_\lambda + \mathbf{J}^{-T}\mathbf{G}^T \mathbf{K}_\lambda \mathbf{G}\mathbf{J}^{-1} \tag{17}$$

3.3.1 Matrix A

First, the derivative of the inverse of a matrix can be written as

$$\frac{d\mathbf{J}^{-T}}{d\mathbf{x}} = -\mathbf{J}^{-T} \frac{d\mathbf{J}^T}{d\mathbf{x}} \mathbf{J}^{-T}. \tag{18}$$

Thus, matrix $\mathbf{A}$ can be expressed, using Eq. (13) as

$$\mathbf{A} = -\mathbf{J}^{-T} \frac{d\mathbf{J}^T}{d\mathbf{x}} (\mathbf{J}^{-T} \boldsymbol{\tau}_\chi + \mathbf{J}^{-T}\mathbf{G}^T \boldsymbol{\tau}_\lambda) = -\mathbf{J}^{-T} \frac{d\mathbf{J}^T}{d\mathbf{x}} \mathbf{f}. \tag{19}$$

Using the chain rule, the derivative is written as

$$\frac{d\mathbf{J}^T}{d\mathbf{x}} \mathbf{f} = \left(\frac{d\mathbf{J}^T}{d\boldsymbol{\chi}} \mathbf{f} \right) \frac{d\boldsymbol{\chi}}{d\mathbf{x}} = \left(\frac{d\mathbf{J}^T}{d\boldsymbol{\chi}} \mathbf{f} \right) \mathbf{J}^{-1}. \tag{20}$$

Hence, a matrix that captures the effect of the external wrench can be defined as

$$\mathbf{K}_E = -\frac{d\mathbf{J}^T}{d\boldsymbol{\chi}} \mathbf{f} = -\left[\left(\frac{d\mathbf{J}^T}{d\chi_1} \mathbf{f} \right) \; \cdots \; \left(\frac{d\mathbf{J}^T}{d\chi_m} \mathbf{f} \right) \right], \tag{21}$$

where χ_i is the ith joint coordinate of $\boldsymbol{\chi}$ and $(d\mathbf{J}^T/d\chi_i)\mathbf{f}$ is a vector forming the ith column of $l \times l$ matrix $\mathbf{K}_E$. Indeed matrix $\mathbf{K}_E$ is equal to the opposite of the matrix noted $\mathbf{K}_G$ in [1].

Therefore from Eqs. (19) and (20), the matrix $\mathbf{A}$ introduced in Eq. (16) is equal to

$$\mathbf{A} = -\mathbf{J}^{-T}\left(\frac{d\mathbf{J}^T}{d\boldsymbol{\chi}}\mathbf{f}\right)\mathbf{J}^{-1} = \mathbf{J}^{-T}\mathbf{K}_E\mathbf{J}^{-1}. \tag{22}$$

3.3.2 Matrix B

Using the chain rule, the right-hand element of matrix $\mathbf{B}$ introduced in Eq. (16), can be differentiated as

$$\frac{d\mathbf{G}^T}{d\mathbf{x}}\boldsymbol{\tau}_\lambda = \left(\frac{d\mathbf{G}^T}{d\boldsymbol{\chi}}\boldsymbol{\tau}_\lambda\right)\frac{d\boldsymbol{\chi}}{d\mathbf{x}} = \left(\frac{d\mathbf{G}^T}{d\boldsymbol{\chi}}\boldsymbol{\tau}_\lambda\right)\mathbf{J}^{-1}. \tag{23}$$

A matrix $\mathbf{K}_{IG}$ that captures the effect of the changes of geometry of the kinematic constraints, is defined as

$$\mathbf{K}_{IG} = \frac{d\mathbf{G}^T}{d\boldsymbol{\chi}}\boldsymbol{\tau}_\lambda = \left[\left(\frac{d\mathbf{G}^T}{d\chi_1}\boldsymbol{\tau}_\lambda\right) \cdots \left(\frac{d\mathbf{G}^T}{d\chi_m}\boldsymbol{\tau}_\lambda\right)\right], \tag{24}$$

where $(d\mathbf{G}^T/d\chi_i)\boldsymbol{\tau}_\lambda$ is a vector forming the ith column of $l \times l$ matrix $\mathbf{K}_{IG}$. Moreover, another matrix noted $\mathbf{K}_{IK}$ that captures the effect of the stiffness of the kinematically constrained joints, is defined as

$$\mathbf{K}_{IK} = \mathbf{G}^T\mathbf{K}_\lambda\mathbf{G}. \tag{25}$$

Matrices $\mathbf{K}_{IG}$ and $\mathbf{K}_{IK}$ are functions of the generalized coordinates and they represent the contribution of the kinematically constrained joints to the stiffness of the mechanism. This contribution is assembled in a matrix $\mathbf{K}_I$, defined as

$$\mathbf{K}_I = \mathbf{K}_{IG} + \mathbf{K}_{IK} = \frac{d\mathbf{G}^T}{d\boldsymbol{\chi}}\boldsymbol{\tau}_\lambda + \mathbf{G}^T\mathbf{K}_\lambda\mathbf{G}. \tag{26}$$

Thus, according to Eqs. (17), (23), (25) and (26), $\mathbf{B}$ is equal to

$$\mathbf{B} = \mathbf{J}^{-T}\mathbf{K}_I\mathbf{J}^{-1}. \tag{27}$$

3.3.3 Cartesian Stiffness Matrix

Finally, combining eqs. (16), (22) and (27), the Cartesian stiffness matrix of a compliant mechanism is written as

$$\mathbf{K}_C = \mathbf{J}^{-T}\left(\mathbf{K}_\chi + \mathbf{K}_I + \mathbf{K}_E\right)\mathbf{J}^{-1}. \tag{28}$$

This matrix includes the three contributions that determine the stiffness of a mechanism according to our initial assumption (no gravity and no dynamical effects), namely: the stiffness of the kinematically unconstrained joints ($\mathbf{K}_\chi$), the stiffness due to the passive joints and the internal torques/forces ($\mathbf{K}_I$) and the stiffness due to the external loads ($\mathbf{K}_E$).

3.4 Stiffness Matrix Expressed in Generalized Coordinates

In the domain of generalized coordinates, the stiffness of the mechanism is described by a matrix $\mathbf{K}_M$ defined as

$$\mathbf{K}_M = \mathbf{K}_\chi + \mathbf{K}_I + \mathbf{K}_E. \tag{29}$$

Therefore, the relation between the stiffness in the generalized domain and in the Cartesian domain is written under a familiar form, namely

$$\mathbf{K}_C = \mathbf{J}^{-T}\mathbf{K}_M\mathbf{J}^{-1} \quad \text{or} \quad \mathbf{K}_M = \mathbf{J}^T\mathbf{K}_C\mathbf{J}. \tag{30}$$

4 Properties of the Stiffness Matrix

4.1 Conservativity of the Matrix

Since the Cartesian stiffness matrix has been calculated by differentiating three torques/forces, namely $\mathbf{f}$, $\boldsymbol{\tau}_\chi$ and $\boldsymbol{\tau}_\lambda$, which are in turn expressed as the derivative of a potential function, $\mathbf{K}_C$ is by definition a conservative matrix. Thus, $\mathbf{K}_C$ is proved symmetric and satisfying the exact differential condition [1].

4.2 A Matrix of a More General Application

The Cartesian stiffness matrices found in the literature can be easily obtained from the matrix presented here. The matrices for serial mechanisms [1, 4] in which there are no passive joints and no internal wrenches such that $\mathbf{K}_I = \mathbf{0}$. As well as the matrices in which the external wrench is not taken into account [2, 4] such that $\mathbf{K}_E = \mathbf{0}$.

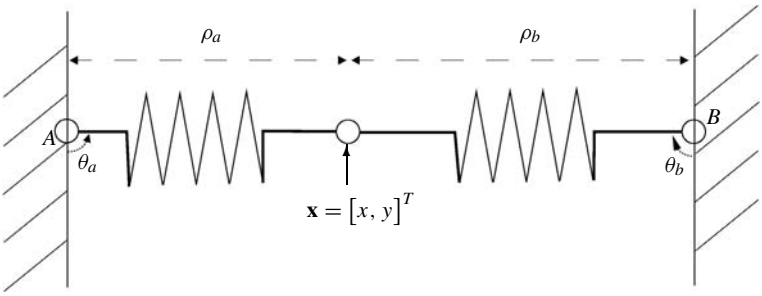


Fig. 1 2-DoF parallel mechanism in unstable static equilibrium.

4.3 Positive Definite Property

The stiffness matrix of a compliant mechanism can be positive definite, semi-positive definite or non-positive definite.

The 2-DoF mechanism presented in the configuration shown in Figure 1 is used to illustrate this property.

4.3.1 Parameters of the Mechanism

The articular coordinates of this mechanism are θ_a, θ_b, ρ_a and ρ_b, while the pose of the platform is $\mathbf{x} = [x, y]^T$. The two base points of the legs, noted A and B, are defined by vectors $\mathbf{a} = (0, 0)$ and $\mathbf{b} = (L, 0)$. The revolute joints are not compliant while both prismatic compliant joints are identical, the free length of their equivalent linear spring is noted ρ_0 and their stiffness coefficient is noted k_ρ. In the configuration presented in Figure 1, the parameters are $\theta_a = 0$, $\theta_b = \pi$, $\rho_a = \rho_b = L/2$ and the external wrench $\mathbf{f} = \mathbf{0}$.

4.3.2 Pose of the End-Effector

The coordinates of leg a are arbitrarily chosen as the generalized coordinates of the mechanism, noted $\boldsymbol{\chi}$. Then, the pose and the Jacobian matrix can be expressed as

$$\mathbf{x} = \begin{Bmatrix} \rho_a \cos\theta_a \\ \rho_a \sin\theta_a \end{Bmatrix} \quad \text{and} \quad \mathbf{J} = \begin{bmatrix} -\rho_a \sin\theta_a & \cos\theta_a \\ \rho_a \cos\theta_a & \sin\theta_a \end{bmatrix}. \tag{31}$$

4.3.3 Geometric Constraints

The two kinematically constrained joints, noted $\boldsymbol{\lambda}$, are θ_b and ρ_b. The geometric constraints that represent the condition of rigidity of the platform, are then written

as

$$\mathbf{x}_a - \mathbf{x}_b = \mathbf{0} \Leftrightarrow \begin{cases} \rho_a \cos\theta_a - (\rho_b \cos\theta_b + L) = 0, \\ \rho_a \sin\theta_a - \rho_b \sin\theta_b = 0. \end{cases} \tag{32}$$

The retained solution of the latter equation is written as

$$\rho_b = \sqrt{\rho_a^2 - 2\rho_a L\cos\theta_a + L^2}, \quad \theta_b = \arctan\left(\frac{\rho 1}{\rho_b}\sin\theta_a, \frac{\rho 1}{\rho_b}\cos\theta_a\right). \tag{33}$$

4.3.4 Kinematic Constraints

Thus, matrix $\mathbf{G}$ defined in Eq. (3) can be derived from the geometric constraints. $\mathbf{G}$ is written as

$$\mathbf{G} = \frac{d\boldsymbol{\lambda}}{d\boldsymbol{\chi}} = \mathbf{J}_b^{-1}\mathbf{J}_a. = \begin{bmatrix} \dfrac{-(\cos\theta_a L - \rho_a)\rho_a}{\rho_a^2 - 2\rho_a L\cos\theta_a + L^2} & \dfrac{-\sin\theta_a L}{\rho_a^2 - 2\rho_a L\cos\theta_a + L^2} \\ \dfrac{\rho_a \sin\theta_a L}{\sqrt{\rho_a^2 - 2\rho_a L\cos\theta_a + L^2}} & \dfrac{-(\cos\theta_a L - \rho_a)}{\sqrt{\rho_a^2 - 2\rho_a L\cos\theta_a + L^2}} \end{bmatrix}. \tag{34}$$

4.3.5 Torque/Force Vectors

The force associated to the passive compliant joint ρ_b is written as $\tau_\rho = k_\rho(\rho_b - \rho_0)$.

4.3.6 Stiffness Matrices Due to Passive Joints

The four components of $\mathbf{K}_{IK}$, defined in Eq. (25), can be analytically calculated as

$$K_{IK}(1,1) = \frac{\rho_a^2 \sin^2\theta_a L^2 k_\rho}{\rho_b^2}, \quad K_{IK}(2,2) = \frac{(\cos\theta_a L - \rho_a)^2 k_\rho}{\rho_b^2}, \tag{35}$$

$$K_{IK}(1,2) = K_{IK}(2,1) = \frac{\rho_a \sin\theta_a L k_\rho(\cos\theta_a L - \rho_a)}{\rho_b^2}. \tag{36}$$

4.3.7 Stiffness Matrices Due to Internal Wrenches

The four components of matrix $\mathbf{K}_{IG}$, defined in Eq. (24), can be analytically calculated as

$$K_{IG}(1,1) = -\frac{{\rho_a}^2 \sin^2\theta_a L^2 \tau_\rho}{\rho_b^3} + \frac{\rho_a \cos\theta_a L \tau_\rho}{\rho_b}, \tag{37}$$

$$K_{IG}(1,2) = K_{IG}(2,1) = \frac{\sin\theta_a L \tau_\rho}{\rho_b} + \frac{(\cos\theta_a L - \rho_a)\,\tau_\rho \rho_a \sin\theta_a L}{\rho_b^3}, \tag{38}$$

$$K_{IG}(2,2) = \frac{\tau_\rho}{\rho_b} + \frac{(\cos\theta_a L - \rho_a)\,\tau_\rho}{2\rho_b^2}. \tag{39}$$

4.3.8 Negative Definite Matrices

In the presented configuration, matrices $\mathbf{K}_M$ and $\mathbf{J}$ are functions of three parameters only, namely k_ρ, L and ρ_0. They are written as

$$\mathbf{K}_M = \begin{bmatrix} k_\rho L(\frac{1}{2}L - \rho_0) & 0 \\ 0 & 2k_\rho \end{bmatrix} \quad \text{and} \quad \mathbf{J} = \begin{bmatrix} 0 & 1 \\ L/2 & 0 \end{bmatrix}, \tag{40}$$

and the Cartesian stiffness matrix (Eq. (30)) is calculated as

$$\mathbf{K}_C = \begin{bmatrix} 2k_\rho & 0 \\ 0 & 2k_\rho(L - 2\rho_0)/L \end{bmatrix}. \tag{41}$$

Therefore, this formulation of the Cartesian stiffness matrix demonstrates that a stiffness matrix can be negative definite: the presented configuration is stable with respect to axis $(O\mathbf{y})$, only if $\rho_0 < L/2$, i.e., if the linear springs are in tension.

5 Conclusion

The presented formulation of the stiffness matrix is a generalization of the already existing stiffness matrices, since it can take into account non-zero external loads, non-constant Jacobian matrices and stiff passive joints, this later point being its main novelty.

Acknowledgements

The authors would like to acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada (NSERC) as well as the Canada Research Chair (CRC) Program.

References

1. Chen, S.F. and Kao, I., Conservative congruence transformation for joint and Cartesian stiffness matrices of robotic hands and fingers. *International Journal of Robotics Research* **19**(9), 835–847 (2000).
2. Gosselin, C.M., Stiffness mapping for parallel manipulators. *IEEE Transactions on Robotics and Automation* **6**(3), 377–382 (1990).
3. Kövecses, J. and Angeles, J., The stiffness matrix in elastically articulated rigid-body systems. *Multibody System Dynamics* **18**(2), 169–184 (2007).
4. Salisbury, J.K., Active stiffness control of a manipulator in Cartesian coordinates. In *Proceedings of the 19th IEEE Conference on Decision and Control*, Albuquerque, NM, USA, pp. 87–97 (1980).
5. Su, H.J. and McCarthy, J.M., A polynomial homotopy formulation of the inverse static analysis of planar compliant mechanisms. *ASME International Journal of Mechanical Design* **128**(4), 776–786 (2006).

MEASURE AND CALIBRATION

A Pair of Measures of Rotational Error for Axisymmetric Robot End-Effectors

Sébastien Briot and Ilian A. Bonev

Department of Automated Manufacturing Engineering, École de Technologie Supérieure (ÉTS), Montreal, Canada; e-mail: sebastien.briot.1@ens.etsmtl.ca, ilian.bonev@etsmtl.ca

Abstract. This paper deals with the problem of representing the rotational error of spatial robots with three orientational degrees of freedom (DOF). Typically, the errors on each of three Euler angles defining the orientation of an end-effector are analyzed separately. However, this is wrong since an accuracy measure should depend only on the "distance" between the nominal pose and the actual one, and not on the choice of reference frame in which these are represented. Several bi-invariant metrics for rotational error exist but are single-parameter and, by definition, disregard the shape of the robot end-effector. Yet, robot end-effectors are typically axisymmetric. Therefore, we propose a two-parameter measure of rotational errors that is better suited for such robot end-effectors.

Key words: accuracy analysis, rotational errors, Euler angles, distance metrics.

1 Introduction

Knowing the exact positioning accuracy is an important challenge when designing a robot. Such an analysis is useful for optimization procedures occurring in the design stage of a new robot. Many tools have been developed in order to promptly estimate the accuracy of a robot. The most common performance indices used to indirectly optimize the accuracy of a robot are the dexterity index (Gosselin, 1992), the condition numbers, and the global conditioning index (Gosselin and Angeles, 1991).

Another way to estimate the accuracy of a robot is to use the first order approximation of the direct kinematic model that maps the input error vector $\delta\mathbf{q}$ to the output error vector $\delta\mathbf{p}$ through the linear relation $\delta\mathbf{p} = \mathbf{J}\delta\mathbf{q}$, where $\mathbf{J}$ is the Jacobian matrix of the robot. However, such an approach will only give a rough estimate of the *maximal* errors occurring in the workspace. This estimate is relevant for infinitesimal errors and far from singularities, but on real robots, errors are not always small.

Furthermore, all of the above tools essentially take into account only errors in the inputs. Yet, errors are due to various factors such as manufacturing tolerances,

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 345–352.

backlash, compliance, sensor errors and control errors. Besides, once a prototype is built, it is important to be able to simply measure and quantify these errors, which is obviously not done using the Jacobian matrix of the robot.

Clearly, from the industrial point of view, the best accuracy measure would be the maximum translational and maximum rotational errors over a given portion of the workspace (Briot and Bonev, 2008; Merlet, 2006a) or at a given nominal configuration. However, while representing the accuracy for planar robots is simple to realize (Briot and Bonev, 2008), this problem is much more complicated when the number of orientational DOFs increases.

Several papers deal with the accuracy of robots with several orientational DOFs (Jelenkovic and Budin, 2002; Kim and Choi, 2000; Merlet and Daney, 2007; Ropponen and Arai, 1995; Wang and Ehmann, 2002; Zhao et al., 2002). In all of these works, the authors analyze the rotational accuracy of robots by considering the errors on three Euler angles. However, such an analysis is wrong because these errors depend on the choice of reference frame in which the nominal pose and the actual one are expressed. Yet, obviously, a rotational error should depend only on the "distance" between the nominal orientation and the actual one.

For the design of mechanisms, defining metrics for measuring the distance between one pose (position and orientation) and another is of utmost importance and is still an area of ongoing research (Larochelle et al., 2007). For representing the accuracy of a robot in industry, however, it is meaningless to combine translational and rotational errors in a single measure. Fortunately, for the specific case of body orientations, there exists a family of *bi-invariant metrics* (Park, 1995; Gupta, 1997). Certainly, the most intuitive one is the rotation angle about a unique axis that brings one reference frame to coincide with another.

Single-parameter invariant measures of rotational error are relevant only for asymmetric robot end-effectors. However, robot end-effectors usually have axial symmetry (in most industrial robots, which are sold without an end-effector, the flange on which an end-effector is mounted is axisymmetric). That is why, in this paper, we propose a pair of rotational error measures that is better suited for such end-effectors. These measures are not invariant because their *raison d'être* is to depend on an end-effector reference frame that takes into account the axial symmetry of the end-effector. However, our measures depend only on the orientation of the actual reference frame with respect to the nominal one.

The remainder of this paper is organized as follows. Section 2 deals with the definition of the proposed measure for orientational accuracy. In Section 3, the new error measure is illustrated on a well-known spherical robot called the *Agile Eye*. Conclusions are given in the last section.

2 Representation of Rotational Errors

To describe the orientation of a body, three consecutive rotations about some of the three axes of the base reference frame are generally used. The angles of these rotations are referred to as Euler angles.

As previously mentioned, the offsets between the corresponding Euler angles associated with two reference frames varies significantly with the choice of base reference frame that is used to define these Euler angels. Therefore, such an offset cannot be used for measuring rotational errors. As a matter of fact, the same is true when considering the individual x, y and z translational errors, though these errors are bounded by the distance between the origins of the first two reference frames. This distance is, in fact, used in industry to represent translational errors, which is the only accuracy measure currently used (ISO 9283 standard).

Thus, it would be great if a similar invariant metric for the "distance" between two orientations existed. Such metrics exist and the best known one is the rotation angle about a unique axis that brings one reference frame to coincide with another. This angle is not only physically meaningful but also quite easy to compute from the trace of the rotation matrix that represents the orientation of one reference frame with respect to the other.

The only problem with such single-parameter invariant metrics is that robot end-effectors are typically axisymmetric. In some cases, when axisymmetric tools are used, the rotation about the tool axis is even irrelevant. Therefore, if industry ever becomes interested in rotational errors, it will more likely look for a two-parameter metric – one measure that reflects the deviation of the axis of symmetry and, for example, another that reflects the rotation about this axis of symmetry.

Consider two different orientations of an axisymmetric robot end-effector: a nominal one denoted by the index 0, and another one denoted by the index 1. Let $\mathbf{d}_0$ and $\mathbf{d}_1$ be two unit vectors parallel to the axis of symmetry of the robot end-effector in each of the two orientations (Figure 1). The first meaningful parameter characterizing the change of orientations is clearly the angle α between vectors $\mathbf{d}_0$ and $\mathbf{d}_1$. Its expression is trivial:

$$\alpha = \cos^{-1}(\mathbf{d}_0^T \mathbf{d}_1). \tag{1}$$

Now, consider that we render the axes of symmetry of both end-effectors of Figure 1 coincident by rotating end-effector 1 about an axis $\mathbf{v}_0$ normal to $\mathbf{d}_0$ and $\mathbf{d}_1$. The smallest angle to which end-effector 1 needs to be rotated about the $\mathbf{d}_0$ axis in order to coincide with end-effector 0 will be denoted by β and will be our second error measure.

Of course, while the choice for the first measure is pretty obvious and unquestionable, there might be other alternatives for the second one. However, we believe that our choice is the most logical and intuitive one.

It can be shown that angles α and β are actually the *tilt* and *torsion* angles (to a sign difference) of the *Tilt-and-Torsion* (T&T) three-angle orientation parameterization introduced in Bonev (2002). Therefore, angle β can be found using the following equation:

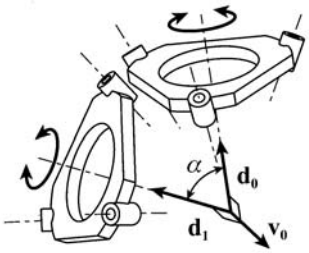


Fig. 1 Measuring the rotation errors of a robot end-effector.

$$\beta = \begin{cases} |\text{atan}\,2(r_{3,2}/\sin\alpha, -r_{3,1}/\sin\alpha) + \text{atan}\,2(r_{2,3}/\sin\alpha, r_{1,3}/\sin\alpha)|, & \text{if } \alpha \neq 0, \\ |\text{atan}\,2(r_{2,1}, r_{1,1})|, & \text{if } \alpha = 0 \end{cases} \quad (2)$$

where $r_{i,j}$ $(i, j = 1, 2, 3)$ are the elements of the rotation matrix that represents the orientation of the reference frame associated with the real pose with respect to the reference frame associated with the nominal pose. Note that β does not depend on the choice of the x and y axes.

Obviously, $0 \leq \alpha < 90°$ in the context of accuracy analysis. In this range, angles α and β are the direct equivalent of the translational and rotational errors, respectively, associated with planar 3-DOF motion.

3 Example: The Agile Eye

In this section, for simplicity, we address the problem of computing the rotational accuracy of a parallel robot in the presence of active-joint errors only. Specifically, we will analyze the rotational accuracy of the well known spherical parallel robot called the *Agile Eye* (Figure 2a). Its kinematics, singularities (Figure 2b), workspace, and working and assembly modes have been studied in detail in several papers (Bonev and Gosselin, 2005, 2006; Bonev et al., 2006). Therefore, we will recall only one important fact before focusing on the accuracy analysis.

The *Agile Eye* is a very particular parallel robot with its Type 1 (serial) and Type 2 (parallel) singularities coinciding and degenerating to six curves as shown in Figure 2b using a T&T angles coordinate system. Each curve corresponds to finite self motions of both the platform and the legs.

3.1 Accuracy Analysis

In order to compute the rotational errors of the *Agile Eye*, we use T&T angles (ϕ, θ, σ) and the following discretization method. For a fixed torsion angle σ, we

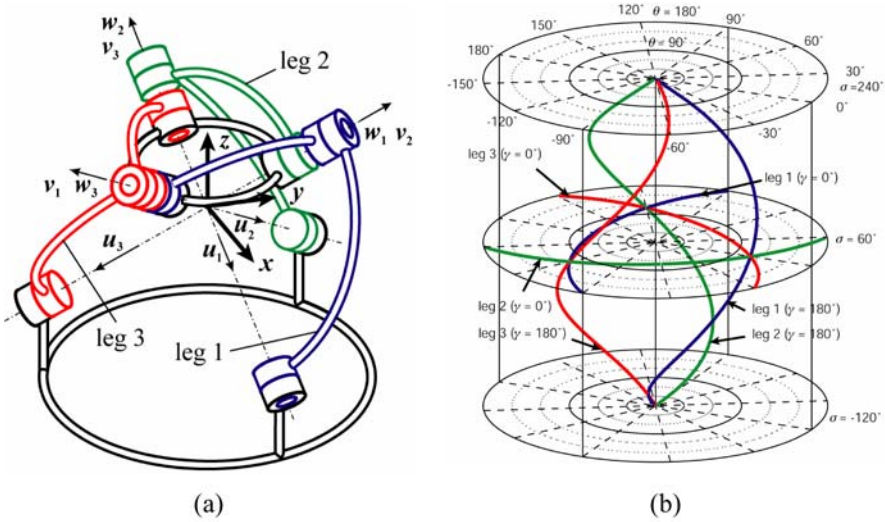


Fig. 2 (a) A CAD model of the *Agile Eye* at its reference configuration (zero active-joint variables and $\phi = 0°$, $\theta = 0°$ and $\sigma = 60°$ T&T angles) and (b) its singularity loci.

discretize the orientation space (ϕ, θ). At a given nominal orientation $(\phi_0, \theta_0, \sigma_0)$ of the mobile platform, we compute the active-joint variables (q_{10}, q_{20}, q_{30}) for one given working mode using the inverse kinematics. Clearly, we should stick to the same working mode throughout the whole workspace. Then, assuming that active-joint variable q_i can vary from its nominal value q_{i0} by an error ε (in our study, $\varepsilon = \pm 0.01°$), we discretize the active-joint interval from $q_{i0} - \varepsilon$ to $q_{i0} + \varepsilon$ and for each set of active-joint variables solve the direct kinematics. For the *Agile Eye*, the direct kinematics can be found analytically. As presented in Bonev et al. (2006), there are four non-trivial solutions which define the assembly modes of the robot. Therefore, the problem is to know which of the obtained solutions is the desired assembly mode corresponding to the nominal configuration. Fortunately, this problem has already been solved in Bonev et al. (2006).

Of course, the direct kinematics of parallel robots can rarely be solved analytically. Therefore, in such cases, a numerical method should be used, such as the Newton–Raphson algorithm. This method is much more time-consuming, but, when sufficiently far from singularities, it converges to the appropriate assembly mode of the robot.

Thus, for a given nominal configuration, it is possible to compute all possible rotational errors α and β, and to retain their maxima. These maximal errors are presented in Figure 3 using T&T polar plots for several constant torsions. We can think of these polar plots as the equivalent of constant-orientation polar plots for planar robots.

Analyzing Figure 3, it appears that our rotational errors grow considerably near singularities (represented by crosses and dotted lines in Figure 3), which is perfectly

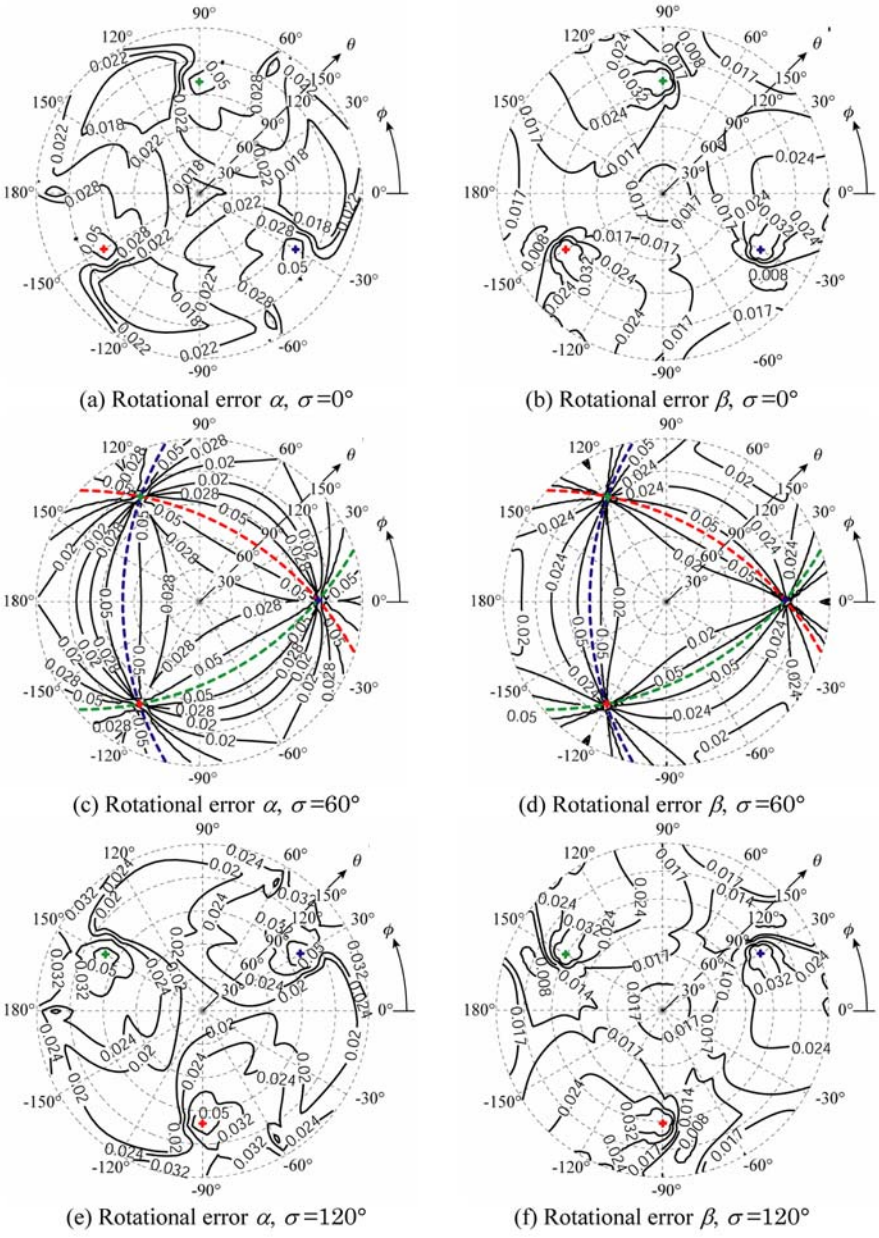


Fig. 3 Rotational errors of the *Agile Eye* for $\sigma \in [0°, 120°]$.

normal. An unexpected result is, however, the fact that, when computing the rotational errors, most of the time, the maximal output errors occur when the active-joint variables are inside their error intervals, whereas it has been demonstrated in Briot and Bonev (2008) that for planar parallel robots, the maximal output errors can only occur when at least two of the active-joint variables are at the extremes of their error intervals. This observation suggests that it should be impossible to find a manageable analytic expression for the output (rotational) errors of even a simple spherical parallel robot such as the *Agile Eye*. A simple discretization method such as ours will be sufficient but more time-consuming and less accurate. However, more efficient approaches could be used, such as interval analysis (Merlet, 2006a).

It should also be noted that the curves are periodic, but not symmetric, which is due to the fact that the output errors depend on the robot's working mode.

4 Conclusions

In this paper, we addressed the problem of representing rotational errors of spatial robots. A review of previous works has shown that, most frequently, authors analyze the errors on each Euler angle defining the orientation of a robot end-effector. However, this is generally wrong since an accuracy measure should depend only on the "distance" between the nominal pose and the actual one, and not on the choice of reference frame in which these are expressed. In contrast, several bi-invariant metrics for rotational error exist. However, they are single-parameter and, by definition, disregard the shape of the robot end-effector. Yet, most robot end-effectors are axisymmetric. Therefore, the choice of end-effector reference frame is not arbitrary and users are likely to be interested in more than one rotational error measure. To fill this gap, we propose a pair of measures of rotational errors that is better suited for such robot end-effectors.

In order to present the efficiency of our approach, we applied it to compute the output errors of a well-known spherical parallel robot called the *Agile Eye*. It was shown that its accuracy depends on the working mode and is poor near singularities.

Acknowledgements

We would like to thank the anonymous reviewers for their valuable remarks that helped us improve considerably this manuscript.

References

Bonev, I.A. (2002), *Geometric Analysis of Parallel Mechanisms*, Doctoral Thesis, Laval University, Quebec, QC, Canada

Bonev, I.A. and Gosselin, C.M. (2005), Singularity loci of spherical parallel mechanisms, in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain.

Bonev, I.A. and Gosselin, C.M. (2006), Analytical determination of the workspace of symmetrical parallel mechanisms, *IEEE Transactions on Robotics* **2**(5), 1011–1017.

Bonev, I.A., Chablat, D. and Wenger, P. (2006), Working and assembly modes of the Agile Eye, in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Orlando, Florida, USA, pp. 2317–2322.

Briot, S. and Bonev, I.A. (2008), Accuracy analysis of 3-DOF planar parallel robots, *Mechanism and Machine Theory* **32**(4).

Gosselin, C.M. (1992), The optimum design of robotic manipulators using dexterity indices, *Robotics and Autonomous Systems* **9**(4), 213–226.

Gosselin, C.M. and Angeles, J. (1991), A global performance index for the kinematic optimization of robotic manipulators, *Journal of Mechanical Design* **113**(3), 220–226.

Gupta, K.C. (1997), Measures of positional error for a rigid body, *Journal of Mechanical Design* **119**, 346–348.

Jelenkovic, L. and Budin, L. (2002), Error analysis of a Stewart platform based manipulators, in *Proceedings of the International Conference on Intelligent Engineering Systems (INES)*, Opatija, Croatia.

Kim, H.-S. and Choi, Y.-J. (2000), The kinematic error bound of the Stewart platform, *Journal of Robotic Systems* **17**(1), 63–73.

Larochelle, P.M., Murray, A.P. and Angeles, J., A distance metric for finite sets of rigid-body displacements via the polar decomposition, *Journal of Mechanical Design* **129**, 883–886.

Merlet, J.-P. (2006a), Computing the worst case accuracy of a PKM over a workspace or a trajectory, in *Proceedings of the 5th Chemnitz Parallel Kinematics Seminar*, Chemnitz, Germany, pp. 83–96.

Merlet, J.-P. (2006b), Jacobian, manipulability, condition number, and accuracy of parallel robots, *Journal of Mechanical Design* **128**, 199–205.

Merlet, J.-P. and Daney, D. (2005), Dimensional synthesis of parallel robots with a guaranteed given accuracy over a specific workspace, in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, pp. 942–947.

Park, F.C. (1995), Distance metrics on the rigid-body motions with applications to mechanical design, *Journal of Mechanical Design* **117**, 48–54.

Ropponen, T. and Arai, T. (1995), Accuracy analysis of a modified Stewart platform manipulator, in *Proceedings of the 1995 IEEE International Conference on Robotics and Automation*, Nagoya, Japan, pp. 521–525.

Wang, S.-H. and Ehmann, K.F. (2002), Error model and accuracy analysis of a six-DOF Stewart platform, *Journal of Mechanical Design* **124**, 286–295.

Zhao, J.-W., Fan, K.-C., Chang, T.-H. and Li, Z. (2002), Error analysis of a serial-parallel type machine tool, *The International Journal of Advanced Manufacturing Technology* **19**(3), 174–179.

Angular-Velocity Estimation from the Centripetal Component of the Rigid-Body Acceleration Field

Philippe Cardou[1] and Jorge Angeles[2]

[1]*Department of Mechanical Engineering, Laval University, Quebec City, QC, Canada, e-mail: pcardou@gmc.ulaval.ca*
[2]*Centre for Intelligent Machines, Department of Mechanical Engineering, McGill University, Montreal, QC, Canada, e-mail: angeles@cim.mcgill.ca*

Abstract. Angular-velocity estimation from point-acceleration measurements is now common practice in biomechanics, and is starting to gain some interest within the robotics community. In this paper we propose two methods for the estimation of the rigid-body angular velocity from the centripetal component of its acceleration field. These methods rely on the adjoint matrix of the symmetric component of the associated angular acceleration matrix. Simulation shows that the methods proposed here can be more robust than their current counterparts.

Key words: angular velocity, rigid-body acceleration field, biomechanics, accelerometer, strap-down inertial measurement unit.

1 Introduction

The fast navigation of mobile robots over rough terrain requires a robust estimation of their velocity and position variables. These vehicles are generally equipped with arrays of different contactless sensors such as accelerometers, gyroscopes, range sensors, magnetometers, and tilt sensors, which are often used *faute de mieux* [7]. In particular, mechanical gyroscopes need to be coupled with other angular sensors, as the shocks involved tend to undermine their accuracy. Due to their inherent simplicity, accelerometers are generally more robust than gyroscopes in that respect, which could make them useful at measuring translations *and* rotations. Such a concept was recently proposed by Lin et al. [4].

On the other hand, in the field of robotic manipulation, common practice is to estimate the angular velocity of the robot end-effector by taking the time-derivative of encoder readouts. This technique is inaccurate when some of the links of the kinematic chain exhibit compliance, a situation that arises in long-reach manipulators [5] and in manipulators that generate high accelerations [3]. The vibrations inherent to these manipulators may be alleviated through control, provided some feedback on the actual pose of the end-effector is available. Among the sensors that have been proposed to measure the vibrations of the end-effector, we find piezoelectric elements [9], strain gages [2], and accelerometers

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 353–360.

[5]. In the latter method, Parsa [5] used the accelerometer readouts to estimate the acceleration field of the robot end-effector, and, from there, its twist and pose parameters.

In this paper, progress is reported on just a piece of the problem addressed by [5], namely, the estimation of the rigid-body angular velocity from the centripetal component of its acceleration field. Besides classical applications in biomechanics [8, 10], the proposed estimation technique is potentially useful to both mobile robots and flexible manipulators.

2 Rigid-Body Acceleration Field

Assume that n accelerometers are attached at n points $\{P_i\}_{i=1}^n$ of a rigid body moving in space. The position of each point P_i in a frame with origin at the fixed point O is given by vector $\mathbf{p}_i \in \mathbb{R}^3$. Let us also define B, a landmark point on the rigid body, and its position $\mathbf{b} \in \mathbb{R}^3$ in the same frame. This allows the definition of body-fixed vectors $\mathbf{r}_i \equiv \mathbf{p}_i - \mathbf{b}$, $i = 1, \ldots, n$. The acceleration of point P_i may be written as

$$\ddot{\mathbf{p}}_i = \ddot{\mathbf{b}} + \mathbf{W}\mathbf{r}_i, \quad i = 1, \ldots, n, \tag{1}$$

where $\mathbf{W} \equiv \dot{\mathbf{\Omega}} + \mathbf{\Omega}^2$ is the angular-acceleration matrix [1], $\mathbf{\Omega} \equiv \mathrm{CPM}(\boldsymbol{\omega})$, $\boldsymbol{\omega}$ is the rigid-body angular velocity, and $\mathrm{CPM}(\cdot)$ returns the cross-product matrix[1] of $(\cdot)$. Apparently, from Eq. (1), one may infer $\mathbf{W}$ from the accelerations of at least three non-collinear points. The angular-acceleration matrix represents the rigid-body acceleration field.

In this paper, we are interested in the estimation of the rigid-body angular velocity from centripetal acceleration (CA) measurements, techniques which we label CA methods. The centripetal component of the acceleration field corresponds to the symmetric component of $\mathbf{W}$,

$$\mathbf{W}_S \equiv \frac{1}{2}\left(\mathbf{W} + \mathbf{W}^T\right) = \mathbf{\Omega}^2. \tag{2}$$

2.1 Existing CA Methods

The most frequent instances of the CA method found in the literature [5, 6] consist in taking the square-root of a linear combination of the diagonal entries of $\mathbf{W}_S$, which may be summarized as

[1] $\mathrm{CPM}(\mathbf{a})$ is defined as $\partial(\mathbf{a} \times \mathbf{b})/\partial\mathbf{b}$, for any $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$.

$$
\begin{aligned}
&\texttt{if } \operatorname{tr}(\hat{\mathbf{W}}_S) < 0 \texttt{ and } \hat{\boldsymbol{\omega}}_{TA} \neq \mathbf{0}_3 \\
&\quad \hat{\zeta}_i = \hat{w}_{i,i} - (1/2)\operatorname{tr}(\hat{\mathbf{W}}_S), \quad i = 1,2,3, \\
&\quad \hat{\omega}_{CAD,i} = \operatorname{sgn}(\hat{\omega}_{TA,i})\mathrm{u}(\hat{\zeta}_i)\sqrt{\hat{\zeta}_i}, \quad i = 1,2,3, \\
&\texttt{else} \\
&\quad \hat{\boldsymbol{\omega}}_{CAD} = \mathbf{0}_3, \\
&\texttt{end}
\end{aligned} \tag{3}
$$

where $\hat{w}_{i,j}$ is the (i,j) entry of $\hat{\mathbf{W}}_S$, $\mathbf{0}_3$ is the three-dimensional zero-vector, $\operatorname{sgn}(\cdot)$ is the signum function, $\mathrm{u}(\cdot)$ is the Heaviside (step) function, and the "hat" symbol ($\hat{\cdot}$) refers to an estimate of the quantity $(\cdot)$. The angular velocity estimate $\hat{\boldsymbol{\omega}}_{TA}$ obtained from tangential acceleration (TA) is used here to resolve the sign ambiguity inherent to CA methods. This estimate is computed from the preceding centripetal acceleration estimate as

$$
\hat{\boldsymbol{\omega}}_{TA}(t) = \hat{\boldsymbol{\omega}}_{CA}(t - \Delta t) + \int_{t-\Delta t}^{t} \hat{\dot{\boldsymbol{\omega}}}(\tau) d\tau. \tag{4}
$$

Indeed, the centripetal component of the acceleration field does not provide any information on the sign of $\boldsymbol{\omega}$, since

$$
\mathbf{W}_S = \boldsymbol{\Omega}^2 = \mathrm{CPM}(\boldsymbol{\omega})\mathrm{CPM}(\boldsymbol{\omega}) = \mathrm{CPM}(-\boldsymbol{\omega})\mathrm{CPM}(-\boldsymbol{\omega}).
$$

As it uses only the diagonal entries of $\mathbf{W}_S$, this algorithm will be referred to as the CAD method.

An alternative method was reported by Peng and Golnaraghi [6]. The authors nonetheless point out that this algorithm is prone to singularity problems. This approach uses the off-diagonal entries of $\mathbf{W}_S$ to estimate the square of the components of the angular velocity. This approach, labelled the CAOD method, is summarized below:

$$
\begin{aligned}
&\texttt{for } i,j,k = 1,2,3, \quad i \neq j \neq k \neq i, \\
&\quad \hat{\xi}_i = \hat{w}_{i,j}\hat{w}_{k,i}/\hat{w}_{j,k}, \\
&\quad \hat{\omega}_{CAOD,i} = \operatorname{sgn}(\hat{\omega}_{TA,i})\mathrm{u}(\hat{\xi}_i)\sqrt{\hat{\xi}_i}. \\
&\texttt{end}
\end{aligned} \tag{5}
$$

3 First Method: Computing the Adjoint Matrix of $\mathbf{W}_S$

From Eq. (2), one notices that $\mathbf{W}_S$ has a rank of two, just as $\boldsymbol{\Omega}$. Moreover, $\boldsymbol{\omega}$ is bound to lie in the one-dimensional nullspace $\mathcal{N}\{\mathbf{W}_S\}$ of $\mathbf{W}_S$, since $\mathbf{W}_S\boldsymbol{\omega} = \boldsymbol{\Omega}^2\boldsymbol{\omega} = \mathbf{0}_3$. Hence, the nullspace of $\mathbf{W}_S$ provides a suitable estimate of the direction of $\boldsymbol{\omega}$. As per the relation

$$
\operatorname{adj}(\mathbf{W}_S)\mathbf{W}_S = \mathbf{W}_S\operatorname{adj}(\mathbf{W}_S) = \det(\mathbf{W}_S)\mathbf{1}_{n\times n} = \mathbf{O}_{3\times 3}, \tag{6}
$$

where $\mathbf{O}_{3\times3}$ is the 3×3 zero-matrix, the range of the adjoint matrix[2] $\mathrm{adj}(\mathbf{W}_S)$ of $\mathbf{W}_S$ is bound to lie in the nullspace of $\mathbf{W}_S$ itself. Hence, any vector $\mathbf{u}$ that does not lie in the nullspace of $\mathrm{adj}(\mathbf{W}_S)$ provides an estimate $\mathrm{adj}(\mathbf{W}_S)\mathbf{u}$ of the direction of $\boldsymbol{\omega}$. In fact, one may verify that $\mathrm{adj}(\mathbf{W}_S) = \mathrm{adj}(\boldsymbol{\Omega}^2) = \|\boldsymbol{\omega}\|_2^2\boldsymbol{\omega}\boldsymbol{\omega}^T$, which shows that *any* vector that is not orthogonal to $\boldsymbol{\omega}$ is suitable. For robustness, $\mathbf{u}$ has to be as parallel to $\boldsymbol{\omega}$ as possible. An estimate of this vector is given by the TA method, thus leading to the algorithm

```
if tr(Ŵ_S) < 0 and ω̂_TA ≠ 0_3
   ‖ω̂_CAAD‖_2 = √(−(1/2)tr(Ŵ_S)),
   v = adj(Ŵ_S)ω̂_TA,
   ω̂_CAAD = (‖ω̂_CAAD‖_2/‖v‖_2)v,
else
   ω̂_CAAD = 0_3.
end
```

which is labelled the CAAD method, AD standing for "adjoint".

4 Second Method: Augmenting Matrix $\mathbf{W}_S$ with the Range of Its Adjoint Matrix

The idea is to compose a full-rank matrix $\mathbf{X}$ for which the product $\mathbf{X}\boldsymbol{\omega}$ is known *a priori*. Such a matrix can be obtained by stacking the additional row $\mathbf{x}^T\mathrm{adj}(\mathbf{W}_S)$ at the bottom of $\mathbf{W}_S$, where $\mathbf{x} \in \mathbb{R}^3$ is not orthogonal to $\boldsymbol{\omega}$. Let $\mathcal{R}\{\cdot\}$ and $\mathcal{S}\{\cdot\}$ represent the range and the support space (or row space) of $\{\cdot\}$, respectively. Since $\mathrm{adj}(\mathbf{W}_S)\mathbf{x} \in \mathcal{R}\{\mathrm{adj}(\mathbf{W}_S)\} = \mathcal{N}\{\mathbf{W}_S\}$, we have $\mathcal{S}\{[\mathbf{W}_S \quad \mathrm{adj}(\mathbf{W}_S)\mathbf{x}]^T\} = \mathcal{S}\{\mathbf{W}_S\} \oplus \mathcal{N}\{\mathbf{W}_S\} = \mathbb{R}^3$, with $\oplus$ indicating, as usual, the *direct sum* of the two vector-space operands. As in the adjoint method, let us choose $\mathbf{x} = \hat{\boldsymbol{\omega}}_{\mathrm{TA}}$ to be our best estimate of a vector that is not orthogonal to $\boldsymbol{\omega}$, thus leading to the matrix $[\mathbf{W}_S \quad \mathrm{adj}(\mathbf{W}_S)\hat{\boldsymbol{\omega}}_{\mathrm{TA}}]^T$. Hence, we will call this approach the augmented matrix method, or, for short, CAAM. However, the current form of the augmented matrix is not yet satisfactory because it is not dimensionally homogeneous. Indeed, in such a case, choosing inappropriate units renders the matrix ill-conditioned and prone to round-off errors due to the finiteness of the machine precision. To avoid such problems, we normalize all quantities using the estimate of the norm $\|\hat{\boldsymbol{\omega}}_{\mathrm{CA}}\|_2^2 = -(1/2)\mathrm{tr}(\mathbf{W}_S)$, which yields

$$\mathbf{X} \equiv \begin{bmatrix} -2\mathbf{W}_S/\mathrm{tr}(\mathbf{W}_S) \\ 4\hat{\boldsymbol{\omega}}_{\mathrm{TA}}^T\mathrm{adj}(\mathbf{W}_S)/(\mathrm{tr}(\mathbf{W}_S)^2\|\hat{\boldsymbol{\omega}}_{\mathrm{TA}}\|_2) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Omega}^2/\|\boldsymbol{\omega}\|_2^2 \\ \hat{\boldsymbol{\omega}}_{\mathrm{TA}}^T\boldsymbol{\omega}\boldsymbol{\omega}^T/(\|\boldsymbol{\omega}\|_2^2\|\hat{\boldsymbol{\omega}}_{\mathrm{TA}}\|_2) \end{bmatrix}.$$

[2] Not to be confused with the adjoint or Hermitian transpose of a linear operator.

Notice that $\mathbf{X}$ is isotropic, since we have, for a perfect estimate $\hat{\boldsymbol{\omega}}_{\mathrm{TA}} = \boldsymbol{\omega}$, $\mathbf{X}^T\mathbf{X} = \mathbf{1}_{3\times 3}$. For ideal measurements, the mapping of $\mathbf{u}_{\boldsymbol{\omega}} \equiv \boldsymbol{\omega}/\|\boldsymbol{\omega}\|_2$ onto the range of $\mathbf{X}$ is $\mathbf{X}\mathbf{u}_{\boldsymbol{\omega}} = [0 \quad 0 \quad 0 \quad 1]^T \equiv \boldsymbol{\xi}$. The least-square approximation of this overdetermined system is found using Householder reflections, which yields

$$
\begin{aligned}
&\texttt{if } \operatorname{tr}(\hat{\mathbf{W}}_S) < 0 \texttt{ and } \hat{\boldsymbol{\omega}}_{TA} \neq \mathbf{0}_3 \\
&\quad \|\hat{\boldsymbol{\omega}}_{\mathrm{CAAM}}\|_2 = \sqrt{-(1/2)\operatorname{tr}(\hat{\mathbf{W}}_S)}, \\
&\quad \hat{\mathbf{X}} = \begin{bmatrix} -2\hat{\mathbf{W}}_S/\operatorname{tr}(\hat{\mathbf{W}}_S), \\ 4\hat{\boldsymbol{\omega}}_{\mathrm{TA}}^T \operatorname{adj}(\hat{\mathbf{W}}_S)/(\operatorname{tr}(\hat{\mathbf{W}}_S)^2 \|\hat{\boldsymbol{\omega}}_{\mathrm{TA}}\|_2) \end{bmatrix}, \quad \boldsymbol{\xi} = \begin{bmatrix} \mathbf{0}_3 \\ 1 \end{bmatrix}, \\
&\quad \{\mathbf{Q}, \mathbf{R}\} \leftarrow \texttt{Householder}(\hat{\mathbf{X}}), \\
&\quad \mathbf{u}_{\boldsymbol{\omega}} \leftarrow \texttt{Backsubstitution}(\mathbf{R}, \mathbf{Q}^T\boldsymbol{\xi}), \\
&\quad \hat{\boldsymbol{\omega}}_{\mathrm{CAAM}} = (\|\hat{\boldsymbol{\omega}}_{\mathrm{CAAM}}\|_2/\|\mathbf{u}_{\boldsymbol{\omega}}\|_2)\mathbf{u}_{\boldsymbol{\omega}}, \\
&\texttt{else} \\
&\quad \hat{\boldsymbol{\omega}}_{\mathrm{CAAM}} = \mathbf{0}_3. \\
&\texttt{end}
\end{aligned}
$$

5 Simulation

In order to compare the robustness of the foregoing methods with that of the previously proposed ones, the motion of a brick freely rotating in space was simulated.

5.1 Accelerometer Array

The inertial measurement unit (IMU) used to estimate the angular velocity of the brick is assumed to be an array of four triplets of mutually orthogonal accelerometers, attached to four of the vertices of the parallelepiped. The layout is depicted in Figure 1, where unit vector $\mathbf{e}_i$ gives the sensitive direction of the i^{th} accelerometer, whereas point P_i gives its location on the brick. Apparently, all direction vectors are parallel to four of the parallelepiped edges; moreover, $P_1 = P_2 = P_3$, $P_4 = P_5 = P_6$, $P_7 = P_8 = P_9$ and $P_{10} = P_{11} = P_{12}$. The brick dimensions are chosen to be $a = 0.07$ m, $b = 0.14$ m, and $c = 0.21$ m, in the X-, Y- and Z-axis direction, respectively. Let us also define frame $\mathcal{B}$, which is attached to the brick and has its origin B located at the brick centroid. All vector quantities in this simulation are to be expressed in frame $\mathcal{B}$.

The ith accelerometer output is decomposed into one deterministic part $a_{i,r}$, which is the actual point-acceleration, and two stochastic parts $\delta a_{i,b}$ and $\delta a_{i,n}$, which are the bias and noise errors, respectively. This gives, at an instant t, $a_i(t) = a_{i,r}(t) + \delta a_{i,b} + \delta a_{i,n}(t)$. In this equation, the bias error is assumed to be time-independent and normally distributed, while the noise is assumed to be white and normally distributed as well. The mean and variance of $\delta a_{i,b}$ are taken to be 0 and

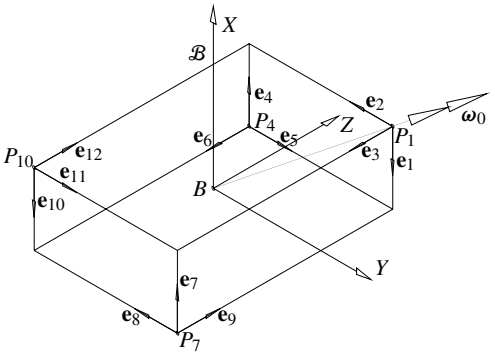


Fig. 1 A brick rotating freely in space.

300 mg, respectively, while the mean and variance of $\delta a_{i,n}$ are taken to be 0 and 75 mg, respectively. A sampling rate of 100 Hz is assumed.

5.2 Trajectory

We assume that neither force nor moment is applied to the brick throughout the simulation. Hence, the angular momentum is preserved, which gives

$$\mathbf{I}_B\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}_B\boldsymbol{\omega} = \mathbf{0}_3, \tag{7}$$

where $\mathbf{I}_B$ is the brick inertia matrix calculated with respect to point B and expressed in frame $\mathcal{B}$. Upon assuming a constant density ρ of the brick material, we obtain $\mathbf{I}_B = \rho\,\mathrm{diag}(10.9,\ 8.4,\ 4.2)\cdot 10^{-6}\ \mathrm{m}^5$ whence the brick motion is independent of ρ. We choose the initial condition to be $\boldsymbol{\omega}_0 = [13.33 \quad 17.77 \quad 22.21]^T$ rad/s, as represented in Figure 1, where one can see that this vector is parallel to one of the great diagonals of the parallelepiped, its magnitude being 10π rad/s.

5.3 Results

The estimates obtained from the existing and proposed CA methods appear in Figure 2. The CAD method exhibits discontinuities whenever one of the components approaches zero, and this, to the point that sign errors are introduced in the estimates. The same problem occurs with a higher amplitude in the CAOD method. Apparently, in this situation, the proposed CA methods are more robust than the existing ones, as can be appreciated from Figure 2b.

The overall performance of a method may be assessed by taking the rms-value of the magnitude of its associated error, i.e., $\delta\omega_{\mathrm{rms}} = \sqrt{(1/n)\sum_{i=1}^{n} \|\hat{\boldsymbol{\omega}}_i - \boldsymbol{\omega}_i\|_2^2}$.The

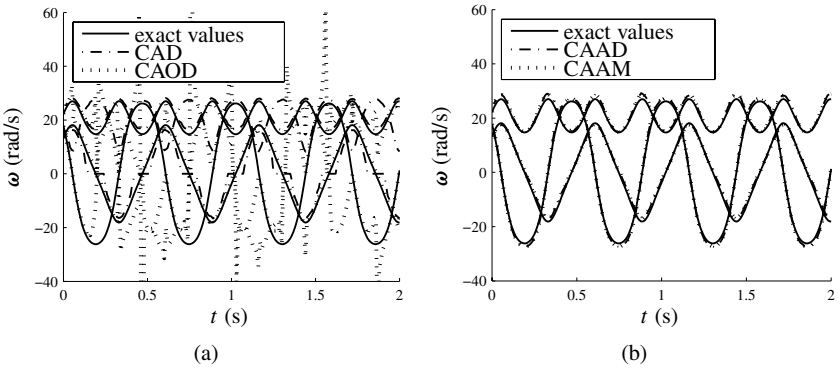


Fig. 2 Angular velocity estimates from (a) existing and (b) proposed CA methods.

Table 1 Error and cost comparisons.

Method	$\delta\omega_{\text{rms}}$	Flops	Square-roots
CAD	27.2 rad/s	18	3
CAOD	55 rad/s	21	3
CAAD	1.3 rad/s	60	2
CAAM	2.05 rad/s	202	6

rms-values resulting from the foregoing simulation are gathered in Table 1, along with the costs associated with their computation. The proposed methods come with heavier computational burdens that correspond roughly to three and ten times those of the already-existing CA methods. In the opinion of the authors, this drawback is less important than the additional robustness provided by the CAAD and CAAM methods, since most applications involve other computations that are far more intensive.

6 Conclusions

Two methods were proposed for the estimation of the angular velocity from centripetal acceleration measurements taken at a constellation of points of a rigid body. In the simulated situation, these methods appear to be more robust than the two existing CAD and CAOD methods. In fact, sign errors render the CAD and CAOD methods unreliable whenever one of the angular velocity components reaches zero; for that reason, their rms-values are significantly higher than those of the other methods. The proposed CA methods circumvent that problem, by relying instead on the square-root of the magnitude of the angular velocity – except, of course, when all the angular velocity components reach zero. We must state, however, that the CAD and CAOD methods are computationally less intensive than the proposed methods.

Acknowledgements

This research was made possible by the Natural Sciences and Engineering Reseach Council of Canada, through a Canada Graduate Scholarship awarded to the first author.

References

1. Angeles, J. (1999), The angular-acceleration tensor of rigid-body kinematics and its properties, *Archive of Applied Mechanics* **69**(3), 204–214.
2. Feddema, J.T., Eisler, G.R., and Segalman, D.J. (1990), Integration of a model-based and sensor-based control for a two-link flexible robot arm, in *Proceedings IEEE International Conference on Systems Engineering*, Pittsburgh, PA, USA, pp. 435–439.
3. Kawamura, S., Choe, W., Tanaka, S., and Pandian S.R. (1995), Development of an ultrahigh speed robot FALCON using wire driven system, in *Proceedings IEEE International Conference on Robotics and Automation*, Vol. 1, pp. 215–220.
4. Lin, P.-C., Komsuoğlu, H., and Koditschek, D.E., (2006), Sensor Data Fusion for body state estimation in a hexapod robot with dynamical gaits, *IEEE Transactions on Robotics* **22**(5), 932–943.
5. Parsa, K. (2003), Dynamics, state estimation, and control of manipulators with rigid and flexible subsystems, Ph.D. thesis, McGill University, Montreal, Canada.
6. Peng, Y.K. and Golnaraghi, M.F. (2004), A vector-based gyro-free inertial navigation system by integrating existing accelerometer network in a passenger vehicle, in *Proceedings IEEE Position Location and Navigation Symposium*, Monterey, CA, USA, pp. 234–242.
7. Singh, S.P.N. (2006), Self-contained measurement of dynamic legged locomotion: Design for robot and field environments, Ph.D. thesis, Stanford University, Stanford, CA, USA.
8. Viano, D.C., Pellman, E.J., Withnall, C.W., and Shewchenko, N. (2006), Concussion in professional football: Performance of newer helmet in reconstructed game impacts – Part 13, *Neurosurgery* **59**(3), 591–606.
9. Wang, X. and Mills, J.K. (2005), Active control of configuration-dependent linkage vibration with application to a planar parallel platform, in *Proceedings IEEE International Conference on Robotics and Automation*, Barcelona, Spain, pp. 4327–4332.
10. Yoganandan, N., Zhang, J., Pintar, F.A., and Liu, Y.K. (2006), Lightweight low-profile nine-accelerometer package to obtain head angular accelerations in short-duration impacts, *Journal of Biomechanics* **39**(7), 1347–1354.

A Novel Point of View to Define the Distance between Two Rigid-Body Poses

Raffaele Di Gregorio

Department of Engineering, University of Ferrara, Via Saragat, 1, 44100 Ferrara, Italy, e-mail: raffaele.digregorio@unife.it

Abstract. The concept of distance between two rigid-body poses is central for lots of applications: path planning, positioning precision, mechanism synthesis, etc. When real motion tasks are considered, the definition of a distance must take into account the size of the rigid body that performs the task for providing useful pieces of information about the task. In this context, a geometric approach to identify distance metrics, meaningful for kinematics tasks, will be presented together with criteria to evaluate the parameters that appear in the formulas.

Key words: SE(3), metrics, positioning precision, mechanism synthesis.

1 Introduction

The need of comparing a reached rigid-body pose (position and orientation) with a desired rigid-body pose arises in many robotics applications. For instance, the evaluation of positioning precision, or of path-planning results, or of mechanism-synthesis solutions are only some tasks that require such a comparison. In these applications, the type of comparison can be implemented by means of a real-valued mapping, $\rho(\mathbf{x}_1, \mathbf{x}_2)$, of two six-tuples (i.e., $\rho : R^6 \times R^6 \rightarrow R$), $\mathbf{x}_1$ and $\mathbf{x}_2$, ($\mathbf{x}_1$ and $\mathbf{x}_2$ identify the two rigid-body poses) which satisfies the three properties (positive definiteness, symmetry, and triangle inequality) of distance metrics.

Defining a distance metric in the configuration space (c-space) of a rigid-body is a difficult problem many researchers have dealt with. It has been mainly addressed according to the two following approaches: (i) the attempt to introduce a distance metric directly in the c-space (Ravani and Roth, 1983; Kazerounian and Rastegar, 1992; Rico-Martinez and Duffy, 1995; Park, 1995; Zefran et al., 1996; Lin and Burdick, 2000; Eberharter and Ravani, 2004; Zhang and Ting, 2005); (ii) the attempt to approximate the displacement in the c-space with a spherical or hyper-spherical displacement, and, then, to use a distance metric of the spherical, SO(3), or hyper-spherical, SO(N), space (McCarthy, 1983; Larochelle and McCarthy, 1995; Etzel and McCarthy, 1996; Larochelle, 1999; Tse and Larochelle, 2000; Belta and Kumar, 2002; Angeles, 2005; Larochelle et al., 2007).

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 361–369.

In this literature, a metric is said to be bi-invariant if it depends neither on the choice of the reference system fixed to the rigid body (body frame), nor on the choice of the reference system fixed to the observer (inertial frame) (Figure 1). Moreover, a metric is said left-invariant (right-invariant), if it does not depend on the choice of the body frame (the inertial frame).

With reference to the above-reported definitions, the following results have been demonstrated: (1) no bi-invariant Riemannian metric can be defined in the special Euclidean group, $SE(3)$ (Park, 1995); (2) the size of the rigid-body must be considered for defining meaningful distance metrics (Kazerounian and Rastegar, 1992; Rico-Martinez and Duffy, 1995); (3) bi-invariance is not necessary to define meaningful distance metrics (Lin and Burdick, 2000).

In this paper, some methods to generate families of distance metrics are presented. Then, among these families, a specific one is selected by using the criterion that a limitation on the distance metric must have a clear geometric meaning. Eventually, it is presented how to determine the distance-metric parameters so that a limitation imposed on the actual pose of a rigid body through that distance metric constrains the maximum displacement of the rigid-body points.

The advantage of the proposed approach is that it immediately visualizes which limitation on the rigid-body actual pose is imposed when a condition on the distance metric is given.

2 Generation of Distance Metrics

A distance metric in an n-dimensional c-space ($n \leq 6$) is a real-valued mapping $d : R^n \times R^n \rightarrow R$ of type $d(\mathbf{y}_1, \mathbf{y}_2)$, where $\mathbf{y}_1, \mathbf{y}_2 \in R^n$ identify two poses in the c-space, that satisfies the following properties:

$d(\mathbf{y}_1, \mathbf{y}_2) = d(\mathbf{y}_2, \mathbf{y}_1)$ (symmetry);

$d(\mathbf{y}_1, \mathbf{y}_2) > 0$ if $\mathbf{y}_1 \neq \mathbf{y}_2$ and $d(\mathbf{y}_1, \mathbf{y}_2) = 0$ if $\mathbf{y}_1 = \mathbf{y}_2$ (positive definiteness);

$d(\mathbf{y}_1, \mathbf{y}_2) \leq d(\mathbf{y}_1, \mathbf{y}_3) + d(\mathbf{y}_3, \mathbf{y}_2)$ for any $\mathbf{y}_3 \in R^n$ (triangle inequality).

It is easy to demonstrate the following proposition:

Proposition 1. If $d_i(\mathbf{y}_1, \mathbf{y}_2)$ for $i = 1, \ldots, m$ are m different distance metrics defined on the same n-dimensional c-space, then any linear combination, $\delta(\mathbf{y}_1, \mathbf{y}_2)$, with non-negative coefficients of the $d_i(\mathbf{y}_1, \mathbf{y}_2)$ is a distance metric of the same c-space, provided that at least one coefficient be different from zero. In formulae:

$$\delta(\mathbf{y}_1, \mathbf{y}_2) \in \left\{ \sum_{i=1,m} a_i d_i(\mathbf{y}_1, \mathbf{y}_2) \mid (a_i \geq 0) \;\&\; (\exists k : a_k \neq 0) \right\}.$$

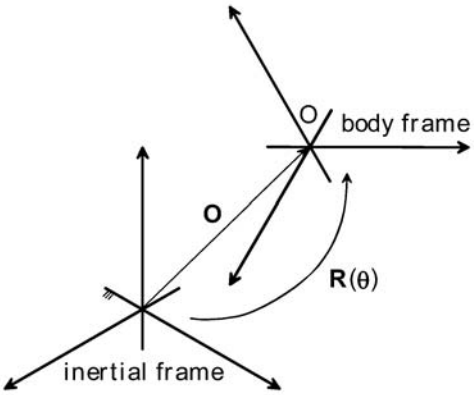


Fig. 1 Notation.

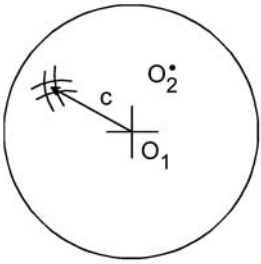


Fig. 2 Geometric meaning of the condition $\delta_T(\mathbf{O}_1, \mathbf{O}_2) < c$.

Considering the displacement of a body frame from the coincidence with the inertial frame to an assigned rigid-body pose, and, then, giving the geometric parameters of that displacement is a common way to identify the assigned pose in the c-space.

The special Euclidean group, $SE(3)$, collects all the possible displacements of a rigid body. $SE(3)$ admits 10 subgroups (Hervé, 1978) with dimension greater than zero and lower than six (lower-mobility subgroups). The elements of $SE(3)$ are 4×4 transformation matrices that depend on six geometric parameters. These parameters can be collected in the six-tuple $\mathbf{x}$. Hereafter, the six-tuple $\mathbf{x}$ will be confused with the element of $SE(3)$ which is computed through the entries of $\mathbf{x}$, and the same simplification of notation will be used for the subgroups of $SE(3)$.

A generic element of $SE(3)$ can be always obtained through the composition of elements that belong to particular lower-mobility subgroups (e.g. one translation composed with one spherical motion). This property allows a rigid-body pose to be parameterized through a six-tuple, $\mathbf{x}$, whose entries can be collected into p n_k-tuples, ${}^k\mathbf{y}$ with $k = 1, \ldots, p$ and $\sum_{k=1,p} n_k = 6$, where each ${}^k\mathbf{y}$ identifies an element of a given lower-mobility subgroup, S_k with dimension n_k. In formulae:

$$\mathbf{x} \in \{({}^1\mathbf{y}^T, \ldots, {}^p\mathbf{y}^T)^T \mid {}^k\mathbf{y} \in S_k \subset SE(3) \text{ for } k = 1, \ldots, p\}.$$

In this context, it is easy to demonstrate the following proposition:

Proposition 2. If $\delta_k({}^k\mathbf{y}_1, {}^k\mathbf{y}_2)$ is a distance metric of S_k, then any linear combination, $\rho(\mathbf{x}_1, \mathbf{x}_2)$, where $\mathbf{x}_1 = ({}^1\mathbf{y}_1^T, \ldots, {}^p\mathbf{y}_1^T)^T$ and $\mathbf{x}_2 = ({}^1\mathbf{y}_2^T, \ldots, {}^p\mathbf{y}_2^T)^T$, of the $\delta_k({}^k\mathbf{y}_1, {}^k\mathbf{y}_2)$ with positive coefficients is a distance metric of $SE(3)$. In formulae:

$$\rho(\mathbf{x}_1, \mathbf{x}_2) \in \left\{ \sum_{k=1,p} b_k \delta_k({}^k\mathbf{y}_1, {}^k\mathbf{y}_2) \mid (\mathbf{x}_1 = ({}^1\mathbf{y}_1^T, \ldots, {}^p\mathbf{y}_1^T)^T) \right.$$
$$\left. \& \ (\mathbf{x}_2 = ({}^1\mathbf{y}_2^T, \ldots, {}^p\mathbf{y}_2^T)^T) \ \& \ (\forall b_k > 0) \right\}.$$

Proposition 2 makes it possible to generate a large family of distance metrics of $SE(3)$ by decomposing a generic displacement into displacements of lower-mobility subgroups, and by defining one distance metric in each subgroup. Once a family of distance metrics is determined, how to select meaningful distance metrics among the members of this family still is an open problem. In the next section, this problem will be addressed by looking for distance metrics that have an immediate geometric meaning.

3 Choice of Meaningful Distance Metrics

The decomposition of a generic rigid-body displacement into displacements that belong to lower-mobility subgroups can be implemented in many ways. Nevertheless, only some subgroups have distance metrics that are easy to use and with a straightforward geometric interpretation. The subgroup of the spatial translations, $T(3)$, and the subgroup of the spherical motions, $S(3)$, are among these subgroups. Since any displacement can be obtained by composing one spatial translation with one spherical motion, $T(3)$ and $S(3)$ will be used to decompose spatial displacements.

When a rigid body is constrained to translate, its pose ($\equiv$ position) is uniquely identified by the coordinates of the origin, O (Figure 1), of the body frame measured in the inertial frame.

The following distance metric is commonly adopted in $T(3)$

$$\delta_T(\mathbf{O}_1, \mathbf{O}_2) = |\mathbf{O}_2 - \mathbf{O}_1|, \tag{1}$$

where $\mathbf{O}_1$ and $\mathbf{O}_2$ are two position vectors, measured in the inertial frame, that locate the position of the origin of the body frame in the two poses, and $|(\cdot)|$ denotes the magnitude of the vector $(\cdot)$.

The distance metric $\delta_T(\mathbf{O}_1, \mathbf{O}_2)$ is bi-invariant in $T(3)$. Moreover, a limitation on $\delta_T(\mathbf{O}_1, \mathbf{O}_2)$ (e.g. $\delta_T(\mathbf{O}_1, \mathbf{O}_2) < c$) has a clear geometric meaning. In fact, it means that O_2 must be located inside a sphere with center O_1 and radius given by the imposed condition (Figure 2).

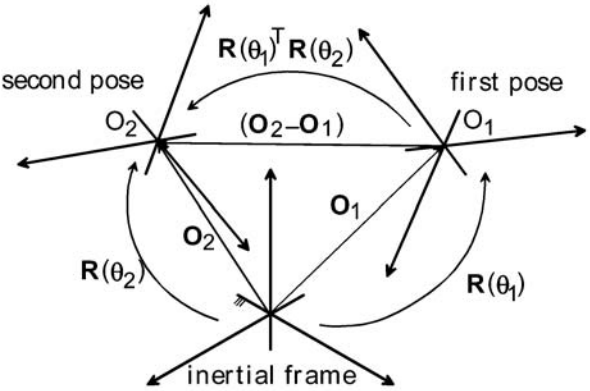


Fig. 3 Relationships between two poses of the same body frame.

When a rigid body is constrained to perform spherical motions with the same center, hereafter assumed coincident with the origin of the body frame, its pose ($\equiv$ orientation) is uniquely identified by the rotation matrix, $\mathbf{R}(\boldsymbol{\theta})$, whose column are the three unit vectors of the body-frame axes projected onto the inertial frame (Figure 1). Such a matrix can be written as an explicit function of three independent parameters, which, in the notation adopted here, are collected in the 3-tuple $vec\theta$. The set that collects all the rotation matrices is named $SO(3)$, and the above considerations state an isomorphism between $S(3)$ and $SO(3)$.

In the literature, a number of distance metrics have been proposed for $SO(3)$ (e.g., Ravani and Roth, 1983; Park, 1995; Larochelle et al., 2007). The following distance metric is among the proposed ones:

$$\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \cos^{-1}\left(\frac{\mathrm{tr}(\mathbf{R}(\boldsymbol{\theta}_1))^T\mathbf{R}(\boldsymbol{\theta}_2)) - 1}{2}\right), \tag{2}$$

where $\mathbf{R}(\boldsymbol{\theta}_1)$ and $\mathbf{R}(\boldsymbol{\theta}_2)$ are the two rotation matrices that locate the two rigid-body orientations with respect to the inertial frame (Figure 3), the image of $\cos^{-1}(\cdot)$ is restricted to the range $[0, \pi]$, and $\mathrm{tr}(\cdot)$ denotes the trace of the matrix $(\cdot)$.

The distance metric $\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ is bi-invariant. It measures the convex rotation angle, in radians, of the finite rotation that brings the first rigid-body orientation to coincide with the second one. On a unit sphere, centered at the center of the spherical motion, such an angle measures the length of the great-circle arc between two points that lie on the equatorial circle perpendicular to the rotation axis, and coincide with each other after the above-mentioned finite rotation.

A limitation on $\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$ (e.g. $\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) < \phi$) has a clear geometric meaning. In fact, it means that each body-frame axis at the second orientation is confined to lie inside a circular cone with vertex at the center of the spherical motion, cone axis coincident with the homologous body-frame axis at the first orientation, and vertex angle given by the imposed condition (Figure 4).

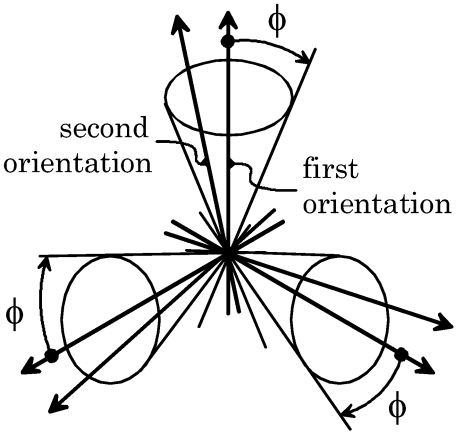


Fig. 4 Geometric meaning of the condition $\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) < \phi$.

The distance metrics (1) and (2), and Proposition 2 can be used to generate the following family of distance metrics of $SE(3)$:

$$\rho_u(\mathbf{x}_1, \mathbf{x}_2) = \delta_T(\mathbf{O}_1, \mathbf{O}_2) + u\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2), \tag{3}$$

where $\mathbf{x}_1 = (\mathbf{O}_1^T, \boldsymbol{\theta}_1^T)^T$, $\mathbf{x}_2 = (\mathbf{O}_2^T, \boldsymbol{\theta}_2^T)^T$, and u is an arbitrary positive constant that is evaluated in the same unit as δ_T.

The analysis of definition (3) reveals that a limitation on $\rho_u(\mathbf{x}_1, \mathbf{x}_2)$ expressed in the following form:

$$\rho_u(\mathbf{x}_1, \mathbf{x}_2) < h, \tag{4}$$

implies the following limitations on δ_T and δ_s, and the associated geometric meanings:

$$\delta_T(\mathbf{O}_1, \mathbf{O}_2) < h, \tag{5a}$$

$$\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) < \frac{h}{u}. \tag{5b}$$

Since the value of $\rho_u(\mathbf{x}_1, \mathbf{x}_2)$ depends on the choice of the origin, O, of the body frame, the distance metrics defined by (3) are, in general, left-invariant.

3.1 Determination of the Parameter u

The meaning of $\rho_u(\mathbf{x}_1, \mathbf{x}_2)$ depends on the value of the parameter u and on the choice of the origin, O, of the body frame. In this section, these parameters are determined so that a limitation on $\rho_u(\mathbf{x}_1, \mathbf{x}_2)$ corresponds to a constraint on the maximum displacement the rigid-body points can perform.

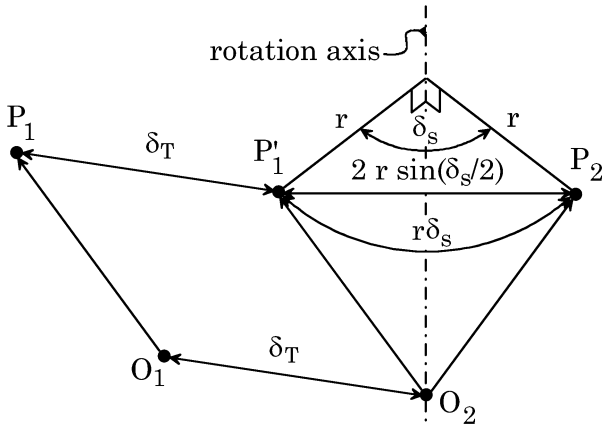


Fig. 5 Displacement of a rigid-body point, P, from the position P_1 to the position P_2 (O_1 and O_2 have the same meaning as in Figure 3).

The displacement, $(\mathbf{P}_2 - \mathbf{P}_1)$, of a rigid-body point, P, at distance r from a rotation axis passing through the body-frame origin, O, is the vector sum of the displacement of O, which has magnitude $\delta_T(\mathbf{O}_1, \mathbf{O}_2)$, and the displacement due to the rotation, which has magnitude $2r \sin(\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)/2)$ and is located in a plane perpendicular to the rotation axis (Figure 5).

The maximum magnitude of $(\mathbf{P}_2 - \mathbf{P}_1)$ occurs when the two vector components are parallel, and r has its maximum value, $r_{\max}$, which coincides with the maximum distance of P from O. Thus, the following relationship holds:

$$|\mathbf{P}_2 - \mathbf{P}_1| \leq \delta_T(\mathbf{O}_1, \mathbf{O}_2) + 2r_{\max} \sin\left(\frac{\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)}{2}\right). \tag{6}$$

Moreover, the following mathematical relationship holds, too:

$$2r_{\max} \sin\left(\frac{\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)}{2}\right) < r_{\max}\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2). \tag{7}$$

With reference to Figure 5, relationship (7) expresses the fact that the chord is shorter than the arc.

Relationships (6) and (7) demonstrate the following inequality:

$$|\mathbf{P}_2 - \mathbf{P}_1| < \rho_{r_{\max}}(\mathbf{x}_1, \mathbf{x}_2), \tag{8}$$

where

$$\rho_{r_{\max}}(\mathbf{x}_1, \mathbf{x}_2) = \delta_T(\mathbf{O}_1, \mathbf{O}_2) + r_{\max}\delta_s(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2). \tag{9}$$

Therefore, if the parameter u is chosen equal to $r_{\max}$, a condition on the maximum point displacement will be imposed through a limitation on $\rho_{r_{\max}}(\mathbf{x}_1, \mathbf{x}_2)$.

Moreover, since choosing the geometric centroid of the rigid body as origin O of the body frame reduces the value of $r_{\max}$, such a choice is advisable when a specific limitation on the orientation error (i.e., a specific condition (5b)) is not necessary. In doing so, condition (5b) becomes looser and easier to satisfy.

4 Conclusions

A technique for generating families of distance metrics of $SE(3)$ has been presented. Among these families, a specific one has been selected which has the property that a limitation on its distance metrics has a clear geometric meaning.

Eventually, the distance-metric parameters have been determined so that a limitation imposed on that distance metric constrains the maximum displacement of the rigid-body points.

The proposed distance metric is simple to be used, and has the advantage that it allows the immediate visualization of which limitation is imposed on the rigid-body actual pose when a condition on the distance metric is given.

Future investigations will try to determine the distance-metric parameters according to other design criteria and to study other families of metrics generated with the proposed technique.

Acknowledgment

This work has been supported by Italian MIUR funds.

References

Angeles, J. (2005), Is there a characteristic length of a rigid-body displacement?, in *Proceedings of the 2005 International Workshop on Computational Kinematics*, Cassino, Italy.

Belta, C. and Kumar, V. (2002), An SVD-based projection method for interpolation on SE(3), *IEEE Transactions on Robotics and Automation* **18**(3), 334–345.

Eberharter, J. and Ravani, B. (2004), Local metrics for rigid body displacements, *ASME Journal of Mechanical Design* **126**(5), 805–812.

Etzel, K. and McCarthy, J.M. (1996), A metric for spatial displacements using biquaternions on SO(4), in *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, Minneapolis, USA.

Hervé, J.M. (1978), Analyse structurale des mécanismes par groupe des déplacements, *Mechanism and Machine Theory* **13**(4), 437–450.

Kazerounian, K. and Rastegar, J. (1992), Object norms: A class of coordinate and metric independent norms for displacements, in *Proceedings of the ASME 1992 Design Engineering Technical Conferences*, Scotsdale, USA.

Larochelle, P. (1999), On the geometry of approximate bi-invariant projective displacement metrics, in *Proceedings of the 10th World Congress on the Theory of Machines and Mechanisms*, Oulu, Finland, Vol. 2, pp. 548–553.

Larochelle, P. and McCarthy, J.M. (1995), Planar motion synthesis using an approximate bi-invariant metric, *ASME Journal of Mechanical Design* **117**(4), 646–651.

Larochelle, P., Murray, A.P. and Angeles, J. (2007), A distance metric for finite sets of rigid-body displacements via the polar decomposition, *ASME Journal of Mechanical Design* **129**(8), 883–886.

Lin, Q. and Burdick, J. (2000), Objective and frame-invariant kinematic metric functions for rigid bodies, *The International Journal of Robotics Research* **19**(6), 612–625.

McCarthy, J.M. (1983), Planar and spatial rigid body motion as special cases of spherical and 3-spherical motion, *ASME Journal of Mechanisms, Transmissions, and Automation in Design* **105**, 569–575.

Park, F.C. (1995), Distance metrics on the rigid-body motion with application to mechanism design, *ASME Journal of Mechanical Design* **117**(1), 48–54.

Ravani, B. and Roth, B. (1983), Motion synthesis using kinematic mappings, *ASME Journal of Mechanisms, Transmissions, and Automation in Design* **105**, 460–467.

Rico Martinez, J.M. and Duffy, J. (1995), On the metrics of rigid body displacements for infinite and finite bodies, *ASME Journal of Mechanical Design* **117**(1), 41–47.

Tse, D.M. and Larochelle, P.M., (2000), Approximating spatial locations with spherical orientations for spherical mechanism design, *ASME Journal of Mechanical Design* **122**(4), 457–463.

Zefran, M., Kumar, V. and Croke, C., (1996), Choice of Riemannian metrics for rigid body kinematics, in *Proceedings of the ASME 1996 Design Engineering Technical Conferences*, Irvine, USA.

Zhang, Y. and Ting, K.-L. (2005), Point-line distance under Riemannian metrics, in *Proceedings of the ASME 2005 Iternational Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Long Beach, USA.

Parallel Robot Calibration by Working Mode Change

P. Last, C. Budde, D. Schütz, J. Hesselbach and A. Raatz

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

Abstract. The positioning accuracy of robotic manipulators can be enhanced by identification and correction of the geometry parameters of the controller model in a way that it best matches the real physical robot. This procedure, denoted as kinematic calibration, is performed by analyzing the difference between conflicting information gained by the kinematic model and corresponding redundant measurement information. Most traditional robot calibration approaches require extra sensors or special constraint fixtures in order to obtain redundancy. This paper proposes a new calibration method that does not require any special calibration equipment, thus being very economical. The presented technique which is designed to be applied to parallel robots is based on a working mode change and incorporates special knowledge about serial singularities. Exemplarily the approach is verified by means of simulation studies on a 3-RRR-structure.

Key words: parallel kinematics, calibration, singularity, working mode.

1 Introduction

The absolute positioning accuracy of industrial robots highly depends upon the kinematic controller model, defining the manipulator geometry. Manufacturing errors and assembly tolerances cause a deviation between the model parameters and the real robot's geometry. In consequence the controller internal model equations do not represent the reality what leads to limited absolute accuracy. A kinematic calibration helps to overcome this drawback. Its purpose is to adjust the parameters in a way to more closely replicate the physical manipulator [11].

Parametric calibration in general requires redundant measurement information that is usually obtained by additional internal or external measurement systems such as lasertracker-devices, theodolites, camera systems or passive joint sensors. Alternatively the robot's degree of freedom (dof) may be restrained by passive mechanical devices. In that case the actuator encoders of the system deliver enough information allowing for parameter identification. Various calibration techniques are compared in [6]. Recently singularity based calibration was introduced [8, 9] constituting a new class of robot calibration methods that is completely free of any additional cal-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 371–380.

ibration equipment. Redundancy is in that case achieved by special knowledge about singularities of type II, also known as direct kinematic singularities.

Although very promising, singularity based calibration as initially introduced has several drawbacks which limit its application, see [8]. In order to overcome these drawbacks while still profiting from the advantages, a new calibration scheme is proposed here which can also be categorized into the class of singularity based techniques. In comparison to the method presented in [8] this contribution provides an idea on how to calibrate a robot by utilizing special knowledge about type I – not type II – singularities. Due to the abandonment of external measurement systems or constraint devices the proposed calibration approach is very inexpensive compared to other techniques. Moreover, it can be automated and repeated whenever necessary.

2 Idea of the Calibration Scheme

As the title of this paper implies a basic step of the calibration technique proposed here is to change the working mode of a parallel robot. According to [1] the working mode is defined as one of the several solutions of the inverse kinematic problem (IKP). Consequently, whenever a working mode is changed a special configuration has to be passed where several solutions of the IKP, at least those between which is switched, coincide. Such configuration is called, inverse kinematic singularity, type I singularity or serial singularity [3, 10]. For parallel robots the IKP is decoupled for each of the limbs connecting base and end-effector platform, meaning that it can be independently solved for each limb. This implies, that each limb may cause an inverse kinematic singularity.

Geometrically type I singularities occur, if the links of a limb are completely stretched or folded upon each other [3]. From a physical point of view the singularity constitutes a situation in which the actuator of the limb which causes the singularity is not participated in compensating for forces acting on the end-effector [5]. This well known fact is the starting point for the new calibration approach presented here. As will be shown it allows for an experimental identification of the singular configuration during a working mode change. In comparison with corresponding information obtained from the kinematic model such measurements enable a robot calibration.

As a prerequisite for the calibration scheme it is assumed that the actuator forces/torques of the robot's motors can be monitored during singularity passing. This is valid assumption because the vast majority of industrial robots is driven by electrical motors for which the actuator current, which can be read from the drive amplifier is directly proportional to the motor torque. Furthermore we assume that each motor is equipped with a position measurement system, allowing to monitor the actuator coordinates.

For the new calibration scheme it is required that the working mode change is conducted by moving just one of the actuators of the system while holding the oth-

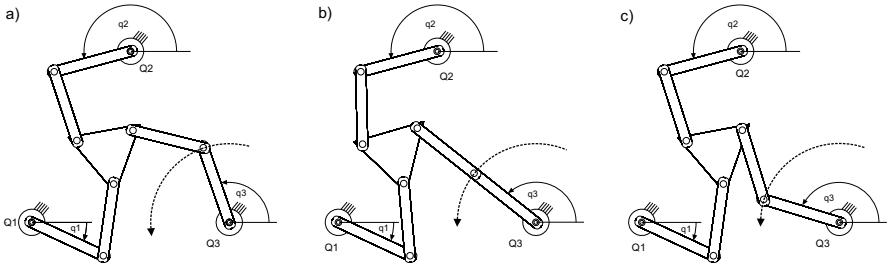


Fig. 1 3-RRR mechanism changing its working mode: (a) before, (b) in, (c) after inverse singular configuration.

ers at a constant position as exemplarily shown in Figure 1 for the 3-RRR structure. The motors which remain at a constant position during singularity passing are called fixed actuators in what follows. Their coordinates are described by $\mathbf{q}_{\text{fixed}} = \{q_{f1}, q_{f2}\}$, while the moving motor's coordinate is q_m. For simplicity only cases are considered, for which a working mode change occurs in the limb, whose actuator is moved.

The motion which is performed in order to make the structure pass the singularity induces a dynamic wrench (forces and torques) to each of the links of the manipulator. With respect to the moving actuator these wrenches may be split into two groups. Part of the wrenches act on the links belonging to the limb which contains the moving motor. These cause an actuator torque/force[1] τ_{F_m} at the moving motor. The remaining forces are discharged into the limb, which contains the moving actuator via the end-effector and thus may be considered as end-effector forces. They cause a torque $\tau_{F_{EE}}$ at the moving motor. Thus in total the torque of the moving actuator is

$$\tau_{m_{\text{total}}} = \tau_{F_m} + \tau_{F_{EE}} \tag{1}$$

In type I singularities, because $\tau_{F_{EE}}$ vanishes as stated above, Equation (1) simplifies to

$$\tau^{\text{sing}}_{m_{\text{total}}} = \tau_{F_m} \tag{2}$$

The amount of τ_{F_m} is unknown, however, we know that the term is independent of the payload $\mathbf{m}$ (mass and inertia parameters of the end-effector), because a dynamic wrench caused by the payload is contained in term $\tau_{F_{EE}}$. This means that if an inverse kinematic singularity is repeatedly passed on the same trajectory (the same path, with the same acceleration- and velocity-parameters) but with differing payloads $\mathbf{m}_1$ and $\mathbf{m}_2$, then $\tau^{\text{sing}}_{F_m}$ remains the same. As a singularity condition it holds

$$\tau^{\text{sing}}_{m_{\text{total}}}(\mathbf{m_1}) = \tau^{\text{sing}}_{m_{\text{total}}}(\mathbf{m_2}) = \tau_{F_m} \tag{3}$$

[1] In accordance to the 3-RRR robot used as an example within this paper we only talk about actuator torques, because this special manipulator only contains rotary motors.

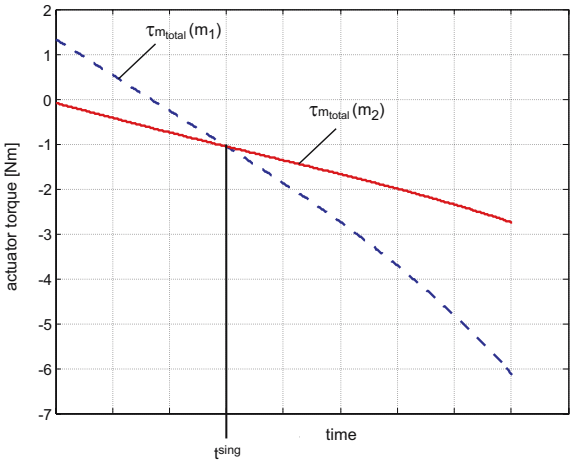


Fig. 2 Torque plots for the moving actuator during transiting a singularity of type I with different payload masses.

This behavior is exemplarily depicted in Figure 2 which shows two actuator-torque-plots $\tau_{1,m_{\text{total}}}(t)$ and $\tau_{2,m_{\text{total}}}(t)$ as a function of time t, obtained for a same trajectory which guides the structure through a singularity of first type. The only difference in both plots is that different payload parameters $\mathbf{m_1}$ and $\mathbf{m_2}$ have been used.

At time t^{sing} both curves intersect, so that singularity condition (3) is fulfilled. If the motor position q_m^{sing} has been monitored at the same time as the motor torque, then q_m^{sing}, which corresponds to the singularity, can be obtained as $q_m^{\text{sing}} = q_m(t^{\text{sing}})$.

The plots in Figure 2 have been generated by means of a multi-body-simulation model of the 3-RRR structure, implemented in SimMechanics©. By means of a co-simulation it could be verified that the interference of both curves indeed occurs at a serial singular configuration.

The described approach allows to detect q_m^{sing} experimentally. Furthermore, since particular geometric conditions need to be fulfilled at a singular configuration of type I, it is possible to compute the actuator coordinate $q_m^{\text{sing}}(\mathbf{k})$ which corresponds to a singularity from the kinematic model including the kinematic parameters $\mathbf{k}$. Comparing both information yields a residual

$$r(\mathbf{k}) = \hat{q}_m^{\text{sing}} - q_m^{\text{sing}}(\mathbf{k}) \tag{4}$$

Passing the inverse kinematic singularity at different locations allows for a formulation of different residual functions. These may be assembled in a vector $\mathbf{r}(\mathbf{k})$. Ideally, if the kinematic model exactly matches the real robot-structure, then $\mathbf{r}(\mathbf{k}) = \mathbf{0}$. Since we assume parameter errors, mathematical optimization methods may be applied to find $\mathbf{k}$ such that the deviations in $\mathbf{r}(\mathbf{k})$ are minimized.

It is convenient to perform the described calibration procedure in a way, that the end-effector payload parameters have to be changed only once during the whole process. Besides this no manual action is required. By means of an automatic tool changer the calibration process can be completely automated. It should be noted that although the proposed technique relies on force measurements no dynamic model of the robot is required. Even the payload parameters do not need to be known. The only thing to guarantee, is that the payload parameters are really different.

Finally a limitation should be mentioned. For parallel kinematic machines with extendable legs, such as the 3-RPR structure [12] or the Gough-platform [4] the presented calibration approach is not applicable. This is because for such manipulators a type I singularity in the sense as described above only occurs if the length of the extendable leg equals zero. This situation is only of theoretical interest but physically not possible.

3 Validation

In order to validate the presented robot calibration approach, the well-known 3-RRR-structure [7] already shown in Figure 1 will be calibrated in a simulation-experiment. The 3-RRR-mechanism is a fully parallel planar manipulator. Its end-effector platform, guided by three independent kinematic chains, can be controlled in three dof, the two translations along the x- and y-axes and a rotation around the z-axis of a Cartesian base coordinate frame. Each of the three identical limbs of the 3-RRR-structure is composed of two links – a crank and a bar element which are connected to each other by a revolute joint. Two additional revolute joints connect the limb to the base-platform and the end-effector-platform, respectively.

3.1 3-RRR-Robot Kinematics

For a kinematic description of the 3-RRR manipulator two coordinate systems – a base frame $\{O\}$ and a platform frame $\{P\}$ – are introduced. In addition several geometric parameters are necessary in order to characterize the kinematic behavior. For each of the three limbs $i = \{1, 2, 3\}$ these are (Figure 3):

- two parameters combined in a vector $\mathbf{r}_{A_i} = [r_{A_i,x}, r_{A_i,y}]^T$ pointing from the base frame's origin O to base joint A_i. $\mathbf{r}_{A_i}$ is expressed in the base frame $\{O\}$;
- two parameters building a vector $\mathbf{r}_{PC_i} = [r_{PC_i,x}, r_{PC_i,y}]^T$ pointing from the platform frame's origin P to platform joint C_i. $\mathbf{r}_{PC_i}$ is expressed in the platform frame $\{P\}$;
- one parameter s_i defining the length $\overline{A_i B_i}$ of the crank element;
- one parameter S_i defining the length $\overline{B_i C_i}$ of the rod connecting crank and end-effector platform.

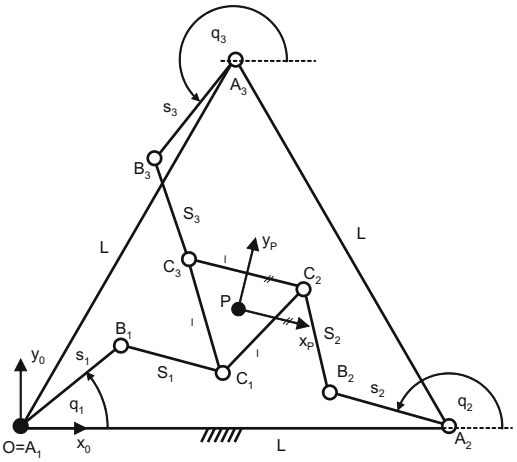


Fig. 3 Kinematic scheme of the 3-RRR mechanism.

Table 1 Nominal kinematic parameters for kinematic limb i of the 3-RRR-structure.

Parameter	Value [mm]
$r_{A_i,x}$	$\frac{L}{2\cos(30°)}\cos(i \cdot 120^\circ + 90°) + \frac{L}{2}$
$r_{A_i,y}$	$\frac{L}{2\cos(30°)}\sin(i \cdot 120^\circ + 90°) + \frac{\sin(60°)L}{3}$
$r_{PC_i,x}$	$\frac{l}{2\cos(30°)}\cos(i \cdot 120^\circ + 150°)$
$r_{PC_i,y}$	$\frac{l}{2\cos(30°)}\sin(i \cdot 120^\circ + 150°)$
s_i	30
S_i	60

Helping-variable	Value [mm]
L	120
l	20

These parameters can be computed according to the definitions given in Table 1.

While typical kinematic problems, such as the inverse or the direct kinematic problem are concerned with relating end-effector and actuator coordinates, the calibration approach presented here requires to determine the actuator coordinate q_m^{sing} from arbitrary given fixed actuator coordinates $\mathbf{q}_{\text{fixed}}$ and a vector of kinematic parameters $\mathbf{k}$ (see Equation (4))

$$q_m^{\text{sing}} = f_{SKP_I}(\mathbf{k}, \mathbf{q}_{\text{fixed}}) \tag{5}$$

We refer to this problem as the *Singular Kinematic Problem (SKP) of type I*.

Unfortunately the 3-RRR-structure's SKP cannot be solved analytically. Hence, an iterative numeric search is conducted in order to evaluate its solutions. The general idea of this approach is adopted from [9]: In a first step the 3-RRR-structure is decomposed by disconnecting the limb which contains the moving actuator at platform-joint C_m (Figure 4). Considering that the two motors of the remaining system are held at constant known values q_{f1} and q_{f2}, we are left with a four bar

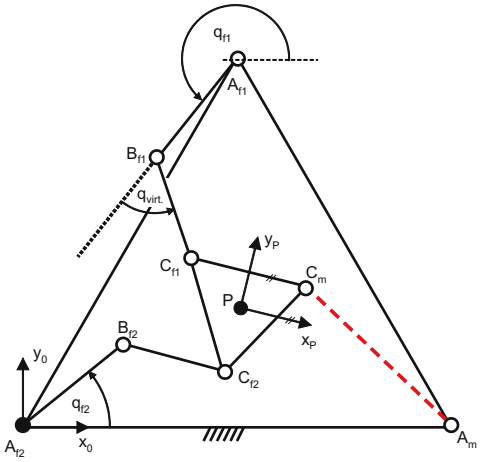


Fig. 4 Disassembled structure for SKP-solution.

linkage defined by the four corner points B_{f1}, B_{f2}, C_{f1} and C_{f2}. Introducing of a virtual motor in one of the passive revolute joints, say B_{f1}, allows for a defined actuation of the four-bar mechanism. That is, given a motor coordinate q_{virt} of the virtual actuator the coordinates of the variable points C_{f1} and C_{f2} can be computed analytically. Additionally it is possible to evaluate the coordinates of C_m which in turn allows to solve the inverse kinematics of the disassembled link, leading to $q_m(q_{\text{virt}})$. The procedure to vary q_{virt} and to compute a corresponding C_m is repeatedly conducted until one of the two geometric conditions $\overline{A_m C_m} = |S_m + s_m|$ or $\overline{A_m C_m} = |S_m - s_m|$ is satisfied. They indicate that the links of the limb containing the moving actuator are completely stretched or folded upon each other respectively and hence, the mechanism is in an inverse kinematic singularity. Once the iteration is terminated with $q_{\text{virt}}^{\text{sing}}$, the IKP is solved for the disconnected limb, leading to $q_m^{\text{sing}}(q_{\text{virt}}^{\text{sing}})$. In simulation studies this algorithm proved to work successful and reliable.

3.2 Simulation Studies

For a proof of principle but without a loss of generality only the three parameters S_1, S_2 and S_3 of the 3-RRR mechanism have been considered to be disturbed by tolerances within the simulation studies. A vector $\mathbf{k}^{\text{real}}$ containing the actual robot geometry-parameters is generated which adds random values in the range [±1 mm] to the nominal values of the kinematic parameters that are supposed to be identified by the calibration process. Nominal parameters $\mathbf{k}^{\text{nom}}$ were already given in Table 1.

Gathering of redundant measurement information is simulated by application of $\hat{q}_{m,j}^{\text{sing}} = f_{SKP_I}(\mathbf{k}^{\text{real}}, \mathbf{q}_{\text{fixed},j})$, where j indicates a specific configuration, defined

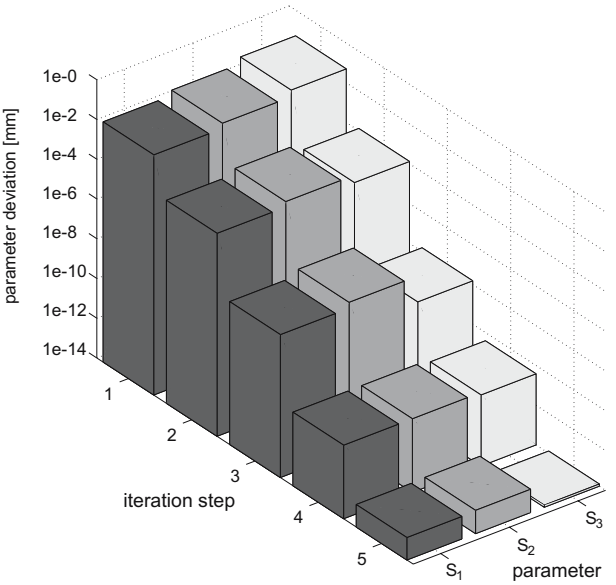


Fig. 5 Typical parameter convergence behavior as found in the simulation studies of the new calibration approach.

by a randomly generated vector $\mathbf{q}_{\text{fixed},j}$. This result is compared to $q_{m,j}^{\text{sing}} = f_{SKP_l}(\mathbf{k}, \mathbf{q}_{\text{fixed},j})$ in a residual $r_j(\mathbf{k})$ according to Equation (4), with $\mathbf{k}$ the current parameter set. Generally $m \geq n$ independent residuals are required for the identification of n kinematic parameters. Hence $\mathbf{q}_{\text{fixed},j}$ is set to $j = 1, \ldots, m$ different values. All $r_j(\mathbf{k})$ are then assembled in $\mathbf{r}(\mathbf{k}) = [r_1, \ldots, r_j]^T$. Starting with $\mathbf{k} = \mathbf{k}^{\text{nom}}$, the goal is to find an optimal parameter set $\mathbf{k}^{\text{opt}}$ that best fits the real robot-structure by minimization of $\mathbf{r}(\mathbf{k})$. For mathematical convenience a function F is defined as $F = \mathbf{r}^T\mathbf{r}$. Minimization of F is then performed by the Levenberg–Marquardt algorithm [13] that has been proven to be successful in various calibration approaches, e.g. [8, 9, 14].

Our first experience with the new calibration approach shows that the identification problem is often ill-conditioned. It is thus recommended to carefully choose the m different measurement configurations. For this reason we applied an algorithm, similar to that proposed in [2], in order to ensure a numerically stable parameter estimation process.

Figure 5 illustrates a typical result of our calibration studies. It shows the deviation between the real and the current parameters in each iteration step of the identification procedure. As can be seen the parameters converge to the real values within a small number of iteration steps indicating that the algorithm works successful.

Clearly, more realistic simulations could have been conducted by considering measurement noise. However, estimating noise amplitudes is not a simple task in our case. This is because in addition to the actuator-position-signals the motor-

torque-signals will be noisy as well. Both signals effect the computation of q_m^{sing} in a different way and the combination of both effects is hard to predict without detailed investigations. Thus, the consideration of noise influence will be subject to further research work.

4 Conclusions

A robot calibration approach, intended to be applied to parallel robots has been presented. It almost exclusively relies on the information delivered by the robot-system itself. Hence, because no special calibration equipment is required within the parameter identification method, the proposed technique is very economical. The basic idea of the new calibration scheme has been derived and validated by means of simulation studies on a simple planar parallel structure. The results emphasize the promising potential of the approach. Future work will focus on investigation of the effect of sensor noise in the actuator position, as well as in the actuator-torque signal.

Acknowledgement

The research work reported here was supported by the German Reserach Foundation (DFG) within the scope of the Collaborative Research Center SFB 562.

References

1. Bonev, I. (2002), http://www.parallemic.org/Terminology/Kinematics.html.
2. Driels, M.R., Pathre, U.S. (1990), Significance of observation strategy on the design of robotic calibration experiments, *Journal of Robotic Systems* **7**(2), 197–223.
3. Gosselin, C., Angeles, J. (1990), Singularity analysis of closed-loop kinematic chains, *IEEE Trans. on Robotics and Automation* **6**(3), 281–290.
4. Gough, V.E., Whitehall, S.G. (1962), Universal tire test machine. In *Proceedings of the 9th International Technical Congress F.i.S.I.T.A*, Vol. 117, pp. 117–135.
5. Helm, M. (2003), *Durchschlagende Mechanismen für Parallelroboter*, Dissertation, TU Braunschweig.
6. Hollerbach, J.M., Wampler, C.W. (1996), The calibration index and taxonomy for robot kinematic calibration methods, *International Journal of Robotics Research* **15**.
7. Hunt, K.H. (1983), Structural kinematics of in-parallel actuated robot arms, *Journal of Mechanisms, Transmission and Automation in Design* **105**(4), 705–712.
8. Last, P., Hesselbach, J. (2006), A new calibration strategy for a class of parallel mechanisms. In *Advances in Robot Kinematics: Mechanisms and Motion*, J. Lenarčič, B. Roth (Eds.), Springer, Dordrecht, pp. 331–338.
9. Last, P., Schütz, D., Raatz, A., Hesselbach, J. (2007), Singularity based calibration of 3-DOF fully parallel planar manipulators. In *Proceedings of the 12th IFToMM World Congress*, Besancon, France.

10. Merlet, J. (2006), *Parallel Robots*, 2nd edition, Springer, Dordrecht.
11. Mooring, B.W., Roth, Z.S., Driels, M.R. (1991), *Fundamentals of Manipulator Calibration*, John Wiley and Sons.
12. Murray, A.P., Pierrot, F. (1998), N-position synthesis of parallel planar RPR platforms. In *Advances in Robot Kinematics*, J. Lenarčič, M.L. Husty (Eds), Kluwer, Dordrecht, pp. 69–78.
13. Scales, L.E. (1985), *Introduction to Non-Linear Optimization*, Macmillan Publishers.
14. Zhuang, H. (1997), Self-calibration of parallel mechanisms with a case study on Stewart platforms, *IEEE Transaction on Robotics and Automation* **13**(3), 387–397.

Augmented Model of the 3-PRS Manipulator for Kinematic Calibration

Steven M. O'Brien and Juan A. Carretero

Department of Mechanical Engineering, University of New Brunswick, Fredericton, Canada, e-mail: {steven.obrien, juan.carretero}@unb.ca

Abstract. Kinematic calibration helps improve the accuracy on any manipulator whereas an extended kinematic model will help even further by providing a closer kinematic representation of the physical manipulator. In this paper an augmented kinematic model of the 3-PRS manipulator is developed. The original architecture and kinematics of the manipulator are presented. The revised kinematic model is then presented, followed by a derivation of the new inverse and forward displacement solutions. Simulation results show that the augmented model improves the accuracy of the manipulator by several orders of magnitude assuming even moderate deviations from the nominal kinematic model.

Key words: parallel manipulator, calibration, kinematics.

1 Introduction

One of the great challenges when working with manipulators is pose seeking. While it is relatively simple to reproduce a recorded position with high accuracy, it is far more difficult for a manipulator to attain an arbitrary position within its task space. This is the distinction between repeatability and accuracy and it is generally accepted that robotic manipulators are much better at the former than the latter [1]. The main cause for this discrepancy is differences between the real manipulator and the theoretical kinematic model (e.g., links are not exactly as long as specified or angles between joints do not match the design). To correct for these defects, the parameters defining the true model need to be measured. This kinematic calibration process has seen a variety of solutions in recent years (e.g., [2, 3]).

Unfortunately, this type of calibration will only improve performance to an extent. In many cases, it is not just the parameters within the model that may be erroneous, but instead the model itself. That is, when kinematic models of manipulators are developed, certain assumptions are made about the geometry of the mechanism (e.g., certain angles are assumed equal, some links are assumed parallel or perpendicular, etc.). While convenient, these assumptions may not always be reasonable, especially as manipulators are required to be increasingly accurate. Therefore, when

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 381–390.

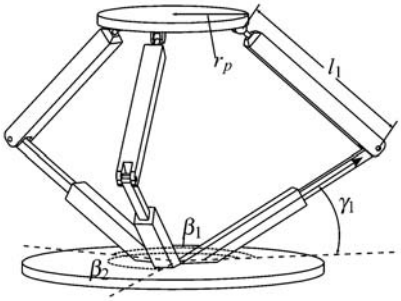


Fig. 1 The 3-PRS manipulator.

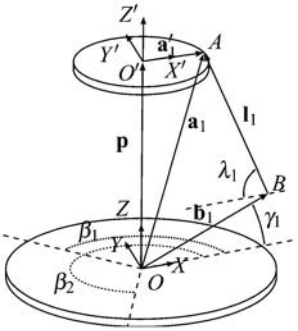


Fig. 2 Vector model of the 3-PRS.

calibrating a particular manipulator, the kinematic model should be reviewed, and augmented where necessary.

The focus of this paper is the review and augmentation of the kinematic models of the 3-PRS parallel manipulator originally developed in [4–6]. In this paper, the assumptions made in the original models are studied. The model is then corrected by the addition of a number of architectural parameters to improve the kinematic model.

2 Ideal Kinematics of the 3-PRS

2.1 Kinematic Parameters

The 3-PRS manipulator has three identical branches, each with five degrees of freedom. Starting at the base, each branch consists of an actuated prismatic joint (P) fixed to the base, followed by a passive revolute joint (R) about which rotates a fixed-length leg link. Finally, connecting the limb to the end effector is a passive spherical joint (S). The manipulator is shown schematically in Figure 1 where the architectural parameters of the ideal kinematic model are shown. The accuracy of these quantities will reflect the accuracy of the kinematic models and are the main parameters being adjusted during calibration.

2.2 Inverse Displacement Solution

The inverse displacement solution for the 3-PRS was developed in [4] and later generalised in [5]. Both these models use vectors defined in Figure 1 and make the following assumptions: (a) all branches are identical, (b) the directions of the prismatic and revolute joints are perpendicular, (c) axis of the revolute joints are

parallel to the base frame, and (d) the line of action of all prismatic joints (i.e., $\mathcal{L}_i$ for $i = 1, 2, 3$) intersect at a common point.

The 3-PRS manipulator has three degrees of freedom (DOF), however, it is a spatial manipulator, which means the end effector can move in all six directions (e.g., x, y, z, ψ, θ, ϕ). Motion in all directions can not be controlled independently, that is, only three of the task-space variables can be made independent. Therefore, the remaining variables are dependent, meaning their values can always be calculated from the independent variables based on the motion constraints imposed by the kinematic structure. Choice of which variables to select as independent is arbitrary and is usually determined by the task at hand. In [4] and most others have used: the elevation of the end effector (displacement along the inertial Z axis), and the tip and tilt angles of the end effector platform (rotations around the inertial X and Y axes defined respectively by ψ and θ). This leaves displacements along X and Y, and rotations around Z (ϕ) as the dependent variables. In fact, calculating the dependent variables from the task variables is the focal point of the IDS. Once the full pose of the end effector is known in space, the position of each spherical joint is easily calculated.

To obtain the dependent variables (x, y, and ϕ), the method shown in [4] exploits the fact that the spherical joints are limited in the positions they can occupy. Based on the assumptions regarding the direction of the revolute joint on branch i, the spherical joint i traces a circle about the revolute joint, and must always lie on a plane that is perpendicular to the XY plane and contains the prismatic joint. Using these assumptions, a simple set of constraint equations can be written that defines the possible positions of each spherical joint.

Substituting the expressions for the position of each spherical joint (calculated from end effector pose) into these constraint equations will yield a system of three equations where only x, y, and ϕ are unknown.

2.3 Forward Displacement Solution

The goal of the FDS is to obtain the desired end effector pose given the position of the actuators. Tsai et al. [6] developed a solution for the forward displacement problem for the 3-PRS. They showed that an analytical solution is possible. However, there are 64 different solutions, and trying to identify the desired solution can be quite challenging. Instead, Tsai et al. suggested a method where an optimisation strategy is used to obtain a solution.

Because the position of the prismatic joints is known, so too is the position of the revolute joint axes. Since the length of the leg links is fixed, each spherical joint can only be located on a circle centred on the revolute joints. The position of each spherical joint can be written in terms of λ_i which is defined as the angle of the leg link to the inertial XY plane. Further, the spherical joints are all fixed to the end effector platform, so the distance between any two joints is both constant and known.

This allows a vector loop equation to be written for each branch combination, which results in three equations where λ_1, λ_2 and λ_3 are the only unknowns (see Figure 1).

Unfortunately, this set of equations is non-linear, which makes a closed form solution difficult, and as previously mentioned, yields many solutions. Instead, these solutions are solved numerically. Tsai *et al.* found that by using this method, they could compel the desired solution by adding additional behaviour constraints to the optimisation algorithm. For this work, a simpler method was used. By estimating the inclination angles λ_i based on the actuator positions, a simple unconstrained local search algorithm (in this case the Levenberg–Marquardt algorithm in [7]) could be used.

After the optimisation, the values of λ_i are used to calculate the position of each spherical joint. From here, it is a simple matter to obtain the position and orientation of the end effector.

3 Additional Kinematic Parameters

As previously mentioned, earlier models make several assumptions about the geometry of the manipulator. To begin, consider the angles between branches: β_1 and β_2. These measurements assume that the angles between the prismatic joints at the base of the manipulator are the same as the angles between the attachment points of the spherical joints on the end effector platform. Instead, α_i is used here to represent the angle between the end effector frame's X axis and the point where the ith spherical joint connects to the platform.

To the same end, it is unlikely that the line of action of all three prismatic joints ($\mathcal{L}_i$ for $i = 1, 2, 3$) will intersect exactly at the origin. To account for this, vector $\mathbf{b}_0^i$ pointing from the origin of the inertial frame to the zero position of each prismatic joint is added. This new point is labelled as B_0^i (see Figure 3). Combined with γ_i and the new angles β_i, each prismatic joint can now be positioned and oriented anywhere in space completely independent of one another.

Next, the axis of the revolute joints are considered. In the previous model it was assumed that they are both perpendicular to the direction of the prismatic joint, and parallel to the inertial XY plane. To eliminate such assumption, two angles are added to each branch: κ_i and ζ_i. Angle κ_i describes the rotation of the revolute joint about an axis parallel to the Z axis, but located at B_i. Next, angle ζ_i describes the rotation of the revolute joint about an axis that is the projection of $\mathcal{L}_i$ onto the XY plane and intersects the revolute joint (see Figure 3). Normally four parameters are required to completely define a revolute joint [8], because this joint is passive, however, there is no need to measure the angulation, and therefore no need to incorporate the joint offset. Furthermore, because the first joint of each branch is prismatic, we can position any reference frame directly on the centre of rotation of the revolute joint, and thus also avoid the typical translation along the joint's axis.

Finally, the spherical joints were also assumed to be ideal in previous models. It was shown in [9] that the error caused by imperfect spherical joints is in fact

Table 1 Kinematic parameters in the complete augmented model.

For each branch i	α_i	β_i	l_i	γ_i	κ_i	ζ_i	b^i_{x0}	b^i_{y0}	b^i_{z0}
Other	r_p								

quite small relative to other types of joints, and is in fact negligible. Based on this information, here it is considered reasonable to model the spherical joints as ideal.

After adding the parameters discussed in this section, the kinematic model will now have 28 independent kinematic parameters, which are summarised in Table 1. It should be noted that while these are all independent parameters, parameters α_1, β_1, ζ_1, b^1_{x0}, b^1_{y0} and b^1_{z0} are likely to be set equal to zero. By doing so, the origin of the inertial frame is located at the zero position of the prismatic joint of branch 1, which is aligned with the X axis. Further, the revolute joint of branch 1 is also set parallel to the XY plane. This fixes the manipulator in space and gives a point of reference for all the other parameters. Depending on the calibration strategy being used fixing the manipulator's location in space may be necessary.

4 Augmented Inverse Displacement Solution

4.1 Augmented Constraint Equations

With a comprehensive kinematic model defined, the next step is to solve the inverse displacement problem. As before with the ideal model, there are only three independent task variables (z, θ, ψ) leaving the other variables as dependent (x, y, ϕ). Similarly, once these six variables are known, so are the locations of the spherical joints, and so it is again simple to obtain the actuator positions in the same manner.

Unfortunately, obtaining the dependent variables from the task variables is no longer straightforward. Previously, the spherical joints were constrained to lie on vertical planes that contained all the elements of the corresponding branch. With the addition of variables ζ_i and κ_i, the constraint is no longer planar. Angle κ_i represents the twist between the leg link l_i and the corresponding prismatic joint. When this angle is non-zero, the circle traced by the spherical joint is skewed with the prismatic joint. As the prismatic joint translates, this circle is extruded to form a cylinder. The resulting cylinder is elliptical because the circle is skewed with the extrusion axis. In most cases the aspect ratio of the ellipse will be so extreme, that the cylinder appears as two parallel planes in the region of interest.

These elliptical cylinders represent the constraint equations for the spherical joints when using the augmented model. In order to define each cylinder, a few frames of reference need to be defined first. All of these frames are shown in Figure 3 for branch 2 (i.e., $i = 2$), and are defined such that {0} indicates frame 0. The frames are:

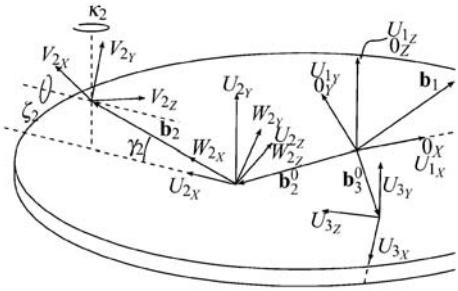


Fig. 3 Various frames used to derive the constraint equations.

- $\{U_i\}$: origin located at B_0^i, U_{iX} axis points along the projection of the prismatic joint onto the inertial XY plane, U_{iZ} lies on the same plane and U_{iY} is perpendicular to the other two.
- $\{V_i\}$: origin located at B_i, V_{iZ} axis aligned with the axis of the revolute joint. Frame is obtained by taking the ideal revolute joint and rotating by ${}^{U_i}R_{V_i} = R_Y(\kappa_i)R_X(\zeta_i)$.
- $\{W_i\}$: origin coincident with $\{U_i\}$ but the W_{iX} axis is aligned with the prismatic line of action, $\mathcal{L}_i$.

From this point forward, the branch indices i will be omitted for the sake of brevity. Of course, all that follows is repeated once for each branch of the manipulator.

To formulate the equation of the constraint cylinder, one starts by defining the circle traced by the centre of the spherical joint when allowed to freely rotate about the axis of the revolute joint. This circle can be defined simply with respect to $\{V_i\}$ by three vectors each representing a point on the circle. The points are chosen as:

$$ {}^{V}\mathbf{G} = l \begin{bmatrix} 1 & 0 & \frac{-1}{\sqrt{2}} \\ 0 & 1 & \frac{-1}{\sqrt{2}} \\ 0 & 0 & 0 \end{bmatrix} \tag{1} $$

These vectors expressed in $\{W\}$. By ignoring the X components of each of these rotated vectors, a projection of the circle onto frame $\{W\}$ is obtained. This projection of the circle forms a cross section of the constraint cylinder, because the X axis of $\{W\}$ is aligned with the direction of the prismatic joint. Since this cross section will be elliptical, it can be defined by the general equation for an ellipse as:

$$ E_1 y^2 + E_2 z^2 + E_3 yz = 1 \tag{2} $$

This equation can be written once for each point on the circle (e.g., the columns of ${}^{V}\mathbf{G}$) projected into $\{W\}$ (i.e., the columns of matrix ${}^{W}\mathbf{G}$). Solving the system of equations it is possible to obtain the values of the ellipse's coefficients: E_1, E_2 and E_3.

When the ellipse defined by this equation is extruded along the W_X axis, it generates an elliptical cylinder whose representative equation represents the possible

locations of the spherical joint expressed in terms of $\{W\}$. To express this equation in terms of the inertial frame ($\{0\}$), first obtain expressions for W_X, W_Y and W_Z in terms of $\{0\}$. Substitution of these expressions into equation (2) produces the equation for an elliptical cylinder in $\{0\}$. That is:

$$\begin{aligned} &D_1x^2 + D_2y^2 + D_3z^2 + D_4xy + D_5xz \\ &\quad + D_6yz + D_7x + D_8y + D_9z + D_{10} = 1 \end{aligned} \tag{3}$$

This equation completely defines the surface of possible locations for the spherical joints. Relative to the end effector platform, the position of the spherical joint can also be written as:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = \begin{bmatrix} x + (c_\theta c_\phi + s_\psi s_\theta s_\phi)a'_x + (s_\psi s_\theta c_\phi - c_\theta s_\phi)a'_y \\ y + c_\psi s_\phi a'_x + c_\psi c_\phi a'_y \\ z + (s_\psi c_\theta s_\phi - s_\theta c_\phi)a'_x + (s_\theta s_\phi + s_\psi c_\theta c_\phi)a'_y \end{bmatrix} \tag{4}$$

where a'_* are the elements of the spherical joint vector $\mathbf{a}'$ relative to $\{0'\}$.

Substituting Equation (4) into Equation (3), yields:

$$\begin{aligned} &C_1x^2 + C_2y^2 + C_3c_\phi^2 + C_4xy + C_5xs_\phi + C_6xc_\phi + C_7ys_\phi + C_8yc_\phi \\ &\quad + C_9s_\phi c_\phi + C_{10}x + C_{11}y + C_{12}s_\phi + C_{13}c_\phi + C_{14} = 1 \end{aligned} \tag{5}$$

where the definition of all the coefficients is omitted here for brevity.

Rewriting this same equation once for each branch produces a system of three non-linear equations.

4.2 Constraint Equations

Because the system of equations obtained from Equation (5) is highly non-linear, a closed-form solution is unlikely, if possible at all. Further, as was the case with the FDS developed in [6], the number of solutions would make identifying the correct one just as onerous. Thus, a numerical technique is applied to solve the equations.

Similar to the FDS, an unconstrained local search algorithm is implemented. Initially, the ideal constraint solution is solved to approximate x, y and ϕ. To improve this first estimate, the position of the spherical joints are evaluated, and then adjusted based on the values of ζ_i and κ_i. For example, a positive value of κ_1 indicates a clockwise twist of the leg link with respect to the prismatic joint (as seen looking in the direction of the negative Z axis). This will cause the spherical joint's position to be adjusted by a distance approximately equal to $(\mathbf{l}_1 \cdot XY)\sin(\kappa_1)$, where $(\mathbf{l}_1 \cdot XY)$ is the projection of $\mathbf{l}_i$ onto the XY plane. From these adjusted positions of the spherical joints, the values of x, y and ϕ are recalculated.

Using these estimated values, a first round optimisation is performed. The goal of this initial optimisation is to obtain values of x, y and ϕ that result in spherical joint positions on the correct side of the constraint surfaces in Equation (5). Because

the 3-PRS manipulator must always have the legs inclined towards the centre, only one side of each constraint cylinder is feasible, and this will depend on the value of κ_i. In the same manner that the spherical joint positions were estimated a priori to this optimisation, it is possible to estimate the correct distance each spherical joint should be from the plane bisecting the constraint cylinder. Using the x, y and ϕ at each iteration to calculate the current position of each spherical joint, it is possible to evaluate the actual distance to this bisection plane. By minimising the discrepancy between this distance and the estimate, the optimisation ensures that the calculated location of the centres of the spherical joints are on the correct side of the constraint cylinder, and also serves as an improved starting point.

With these improved values of x, y and ϕ, a final optimisation is performed. This time, however, the constraint equations derived in Section 4.1 are minimised. A very low termination threshold is used at this stage, as any error in the constraint equations will translate directly to error in the spherical joint positions, which in turn impairs the accuracy of the actuator position calculations. The Levenberg–Marquardt algorithm is the algorithm used in both optimisation procedures. The implementation in [7] is used in this work.

5 Augmented Forward Displacement Solution

Unlike the inverse displacement solution, few changes were needed when solving the augmented forward displacement problem. Because all the kinematic parameters are known, it is still possible to calculate the position of the revolute joint with the prismatic joint position. The leg link inclination angle, λ_i, is no longer a measure of the angle between the leg link and the XY plane, but it can still be used in the same manner as described in Section 2.3. A slight adjustment is made when estimating λ_i to account for the changes caused by κ_i.

The vector loop equations can be derived in exactly the same manner as in Section 2.3, and the optimisation yields the angles, λ_i, necessary to satisfy these equations. From here, while the position of the spherical joints is adjusted by the values of κ_i and ζ_i, the procedure is the same. The end effector pose is calculated in the same fashion.

6 Results

Results for the augmented kinematic model of the 3-PRS manipulator have been very encouraging thus far. In order to simulate the error that arises as a result of the numerical solution of the constraint equation, the IDS is executed first. Using the resulting actuator positions, the FDS is subsequently executed. The position of the spherical joints is recorded for each of the two solutions, and compared. The reported overall error is measured as the 2-norm of the distances between corresponding spherical joints in each solution.

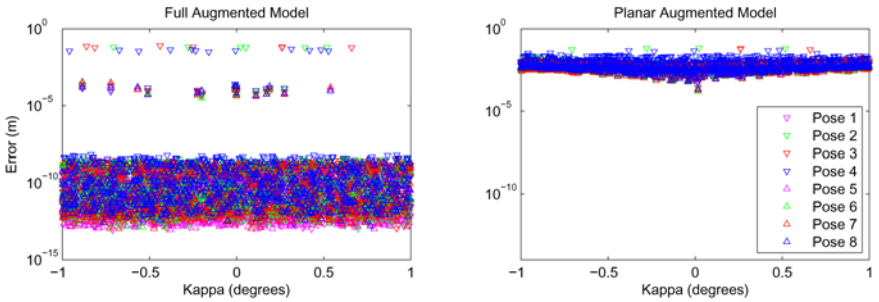


Fig. 4 Error between the IDS and FDS versus parameter κ_1 for both the full model (left) and the planar model (right) that ignores κ.

In general, the resulting error appears as in Figure 4 (left). The vast majority of the time (approx. 98.5% of tests) the error is less than 10^{-8} m, meaning errors of less than 10 nm. The formations of horizontal lines around 10^{-4} to 10^{-6} m are caused when κ_i of any one or more of the branches approaches zero. When this occurs, the distance between the two sides of the elliptical constraint cylinder is so small, that the optimisation algorithm can jump from one side to the other in a single iteration.

Because of these errors, a model that did not include κ_i was also considered. Such a model has the significant advantage that a closed-form solution is possible, which would greatly decrease the time required to solve the IDS. Without κ_i, the constraints on the spherical joint positions is no longer a cylinder, but a plane (although inclined due to ζ_i). This resulted in a linear system which allowed for an analytical solution which, in turn, performed much faster.

Unfortunately, when compared against the full augmented model, the results were somewhat disappointing. To test the planar version of the augmented model (all parameters except κ_i), the IDS was executed, and then the FDS for the full model was used as a baseline for comparison. The results are shown in Figure 4 (right). In general there was at least about 10^{-3} m of error when using the planar model. Also, as κ_i approached zero, the error decreased significantly as would be expected (indeed, if all $\kappa_i = 0$, then the model is exactly the planar model).

It should be noted however, that the increase in accuracy is not without a price. As mentioned previously, the augmented kinematic model precludes a closed-form solution for the inverse displacement problem. The numerical solution is approximately equivalent to solving the ideal IDS n times, where n is the number of iterations required for the algorithm to converge. Results showed that between 10 and 30 iterations were generally required to attain convergence. Generally speaking however this is still quite good, especially when compared to the FDS, which has numerical solutions four layers deep.

The advantage of testing the planar model is that even when the full model encounters difficulty with small values of κ, the error is still better than that demonstrated by the planar model for similar values. This indicates that although in a very small number of cases the optimisation algorithm is terminating on the wrong side

of the constraint cylinder, the two sides are so close together that either one is still an improvement over a reduced kinematic model. Overall the model allows for a much more accurate representation of the 3-PRS manipulator that should prove useful for calibration.

7 Conclusions

This work presented an extended kinematic model for the 3-PRS manipulator that, compared to previous models, can better match a physical manipulator. With proper calibration, this will allow for greater absolute accuracy when controlling a 3-PRS manipulator. The augmented model also allows for improved performance during calibration. Each parameter that is added reduces the number of deviations being estimated by all other parameters. That is, in a reduced model, the remaining parameters will be distorted to compensate for the deviation of the absent parameters. Future work includes experiments on an existing 3-PRS model to verify that the work presented here translates to real world applications, as well as to quantify the actual improvement in accuracy.

Acknowledgement

The authors acknowledge the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. J. Ziegert, P. Datseris, *Basic Considerations for Robot Calibration*, pp. 932–938 (1988).
2. P. Renaud, A. Vivas, N. Andreff, P. Poignet, P. Martinet, F. Pierrot, O. Company, Kinematic and dynamic identification of parallel mechanisms, *Control Engineering Practice* **14**(9), 1099–1109 (2006).
3. S.M. O'Brien, J.A. Carretero, P. Last, Self calibration of 3-PRS manipulator without redundant sensors, *Transactions of the Canadian Society for Mechanical Engineering* **31**(4), 483–494 (2007).
4. J.A. Carretero, R.P. Podhorodeski, M.A. Nahon, C.M. Gosselin, Kinematic analysis and optimization of a new three degree-of-freedom spatial parallel manipulator, *Journal of Mechanical Design* **122**, 17–24 (2000).
5. G. Pond, J.A. Carretero, Kinematic analysis and workspace determination of the inclined PRS parallel manipulator, in: *Proceedings of RoManSy 2004: 15th CISM-IFToMM Symposium on Robot Design*, Montreal, Canada (2004).
6. M.-S. Tsai, T.-N. Shiau, Y.-J. Tsai, T.-H. Chang, Direct kinematic analysis of a 3-prs parallel mechanism, *Mechanism and Machine Theory* **38**(1), 71–83 (2003).
7. M.I.A. Lourakis, A brief description of the Levenberg–Marquardt algorithm implemened by `levmar`, http://www.ics.forth.gr/∼lourakis/levmar/ (February 11, 2005).
8. H. Zhuang, Z.S. Roth, *Camera-Aided Robot Calibration*, CRC Press (1996).
9. O. Masory, J. Wang, H. Zhuang, Kinematic modeling and calibration of a Stewart platform, *Journal of Advanced Robotics* **11**(5), 519–539 (1997).

The Calibration of a Parallel Manipulator with Binary Actuation

Jean-Sébastien Plante[1] and Steven Dubowsky[2]

[1]*Département de Génie Mécanique, Université de Sherbrooke, Québec, Canada J1K 2R1, e-mail: jean-sebastien.plante@usherbrooke.ca*
[2]*Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A., e-mail: dubowsky@mit.edu*

Abstract. This paper investigates how parameter identification can improve the calibration of elastically averaged parallel manipulators. The method developed is applied to a needle manipulator for intra-Magnetic Resonance Imaging (MRI) prostate cancer treatment. The device uses MRI-compatible Dielectric Elastomer Actuators to precisely guide a needle during its insertion for taking biopsy or implanting radioactive seeds in the prostate. Here, an analytical model of the device is presented and a sensitivity analysis identifies the key model parameters. Parameter identification is conducted with an optimization algorithm. The method is tested on both simulated parameter values and experimental data. The results show that the method can improve the device accuracy by at least one order of magnitude (on simulated parameters).

Key words: binary actuation, parallel, dielectric elastomer, calibration.

1 Introduction

Binary actuation such as proposed by Anderson and Horn (1967) and Roth et al. (1973) is a design paradigm for robotic and mechatronic systems that uses a large number of binary (bistable) actuators to achieve continuous-like motion. System design and control is greatly simplified compared to continuous systems since high precision sensors and low level control is not needed. A major challenge of designing practical binary systems is the development of binary actuators and systems architectures that are appropriate for systems containing 10 to 100 actuators. Conventional actuators are too heavy and expensive.

DeVita et al. (2007) proposed a new design paradigm for binary systems called elastically averaged parallel actuation using Dielectric Elastomer Actuators (DEAs). DEAs are a promising technology for binary systems such as demonstrated by Wingert et al. (2002) and Plante et al. (2007). These actuators have good performance and are lightweight and low cost. Therefore, they are practical for systems with large numbers of actuators. They have also been shown to be MRI-compatible.

A functional prototype developed by Tadakuma et al. (2008) experimentally demonstrated the feasibility of elastically averaged parallel systems using binary

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 391–398.

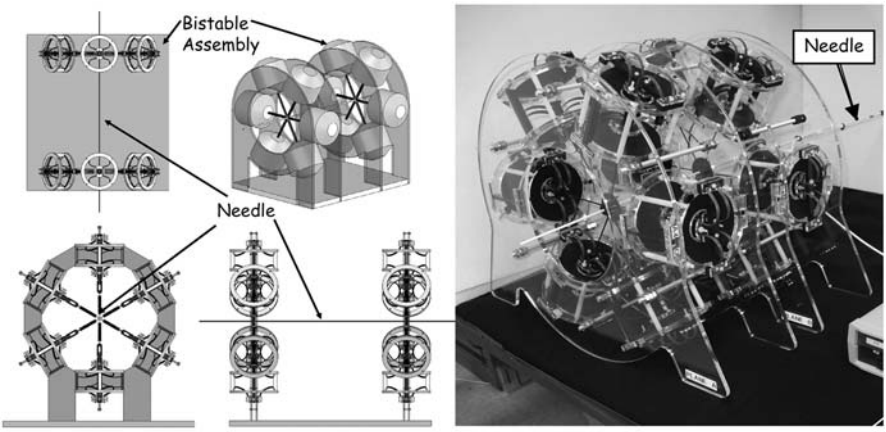


Fig. 1 Elastically averaged parallel manipulator for MRI-guided prostate cancer therapy.

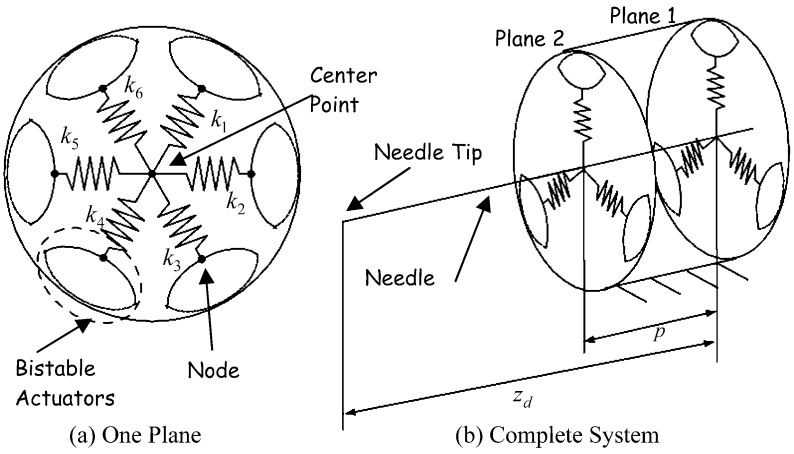


Fig. 2 Schematic of the proposed manipulator.

DEAs, see Figures 1 and 2. In this case, the device is used as a needle manipulator for intra-MRI prostate cancer therapies. It would be guided by surgeons using real-time MR images to drive a needle inside the prostate and reach millimeter sized tumors. The device would allow accurate biopsies and the injection of therapeutic agents like radioactive pellets or liquid nitrogen. Treatment costs and duration would be reduced while improving success rates.

The elastically averaged manipulator of Figures 1 and 2 consists of two planes each with 6 bistable DEAs and 6 springs. Needle motion is achieved by changing the position of the center points of each plane when the static equilibrium of the spring system is altered by the position (extended or retracted) of the bistable actuators.

To date, the kinematics of elastically averaged parallel manipulator remains largely unaddressed as most work on binary system addressed the kinematics of serial chain (snake-like) mechanisms (see, for example, Chirikjian and Burdick, 1995). Such configurations offer large workspaces but have low stiffness and compactness compared to the elastically averaged parallel approach.

This paper shows that the kinematic model of an elastically averaged parallel manipulator can be improved through a relatively simple parameter identification method. A model of the system is presented and its key parameters are identified through sensitivity analysis. Parameter identification is then conducted on simulated parameter values as well as on experimental data. Results show that parameter identification can improve the system accuracy by at least one order of magnitude (on simulated parameters).

2 Analytical Development

2.1 Manipulator Model

Binary manipulator inputs can be represented in a binary sequence of 0 and 1:

$$Q = [a_1 \;\; a_2 \;\; \ldots \;\; a_2], \tag{1}$$

where $a_i = 1$ or 0 is the state (extended or retracted) of the ith bistable assembly and n is the total number of bistable modules. These inputs define a finite set of discrete points within the system's workspace, here $2^{12} = 4096$ points. Systems workspaces are, in general, not evenly distributed. Analytical models that map the manipulator output points for any given inputs are essential to their design and control.

The spatial configuration of an elastically averaged parallel manipulator can be found from the static equilibrium of the springs system such as proposed by Devita et al. (2007). Here, this is done independently on each plane of the device. Spring deformations and applied forces in one plane of the device are represented schematically in Figure 3 (showing only 3 springs). Referring to the figure, the model uses the following inputs:

k_i	stiffness of ith spring
$\mathbf{l}_{0_i} = l_{0_i}\mathbf{w}_i$	position vector of the ith undeformed spring
$\boldsymbol{\beta}_i = \beta_i \mathbf{w}_i$	pre-stretch vector of the ith spring
$\boldsymbol{\delta}_i = \delta_i \mathbf{w}_i$	stroke vector of ith bistable module; the magnitude of the stroke vector is given by $\delta_i = \zeta_i a_i$ with a_i (0 or 1) the state of the bistable module and ζ_i the stroke
$\mathbf{f}_{\text{ext}}$	external force vector at center point
$\mathbf{W}$	weight vector at center point

in order to compute the following output parameters:

$\mathbf{u} = u\mathbf{e}$	displacement vector of the center point

$\mathbf{l}_i = l_i \mathbf{v}_i$ position vector of the ith deformed spring
$\mathbf{F}_i = F_i \mathbf{v}_i$ internal force vector of the ith spring

In the above lists, $\mathbf{w}_i$, $\mathbf{v}_i$ and $\mathbf{e}$ are the respective unit vectors of the undeformed direction of the ith spring, the deformed direction of the ith spring, and the displacement of the center point.

Static equilibrium requires that the sum of the forces at the center point is zero:

$$\sum \text{forces} = \sum_i \mathbf{F}_i + \mathbf{f}_{\text{ext}} + \mathbf{W} = \mathbf{0}. \tag{2}$$

The internal force of the ith spring is found from the spring elongation and is given by:

$$\mathbf{F}_i = (k_i(l_i - l_{0_i})\mathbf{v}_i = k_i[(l_{0_i} + \beta_i + \delta_i)\mathbf{w}_i - u\mathbf{e} - l_{0_i}\mathbf{v}_i]. \tag{3}$$

Eqs. (2) and (3) are solved iteratively for the displacement vector of the center point: $\mathbf{u}$. The method is used to find the center point locations of each plane, $\mathbf{u}_1 = (x_1, y_1, z_1)$ and $\mathbf{u}_2 = (x_2, y_2, z_2)$, from which the needle tip (end effector) location can be found by:

$$\mathbf{u}_d = (x_d, y_d, z_d) = \mathbf{u}_1 + \frac{z_d}{p}(\mathbf{u}_2 - \mathbf{u}_1), \tag{4}$$

where p and z_d are geometric parameters defined in Figure 2.

This paper uses the nominal parameter values of the prototype shown in Figure 1. These are:

- $k_{\text{Plane 1}} = \left[\frac{3k}{2}\ \frac{k}{2}\ k\ \frac{k}{2}\ k\ \frac{k}{2}\right]$ and $k_{\text{Plane 2}} = \left[\frac{k}{2}\ k\ \frac{k}{2}\ \frac{k}{2}\ \frac{k}{2}\right]$ and $k = 0.044$ N/mm;
- $l_{0_i} = 86$ mm, $\beta_i = 0$ mm, $\zeta_i = 13$ mm for all i;
- $z_d = 610$ mm , $p = 180$ mm;
- $\mathbf{f}_{\text{ext}} = \mathbf{W} = \mathbf{0}$.

Note that the prototype of Figure 1 was built at twice its clinical size to accommodate current handmade actuators.

2.2 Sensitivity Analysis

A sensitivity study is conducted to identify the parameters of the manipulator that have the most influence on its accuracy. The parameters considered are spring length, spring stiffness, and actuator stroke (l_{0_i}, k_i, ζ_i). Dimensional and geometric errors of the various parts composing the manipulator structure are not considered here. These errors would be small with proper manufacturing techniques.

Errors are introduced using randomly generated numbers following normal distributions. These distributions are centered on the nominal parameter values:

$$\tilde{l}_{0_i} = l_{0_i} + \sigma_{l_i} z, \quad \tilde{k}_i = k_i + \sigma_{k_i} z, \quad \tilde{\zeta}_i = \zeta_i + \sigma_{\zeta_i} z, \tag{5}$$

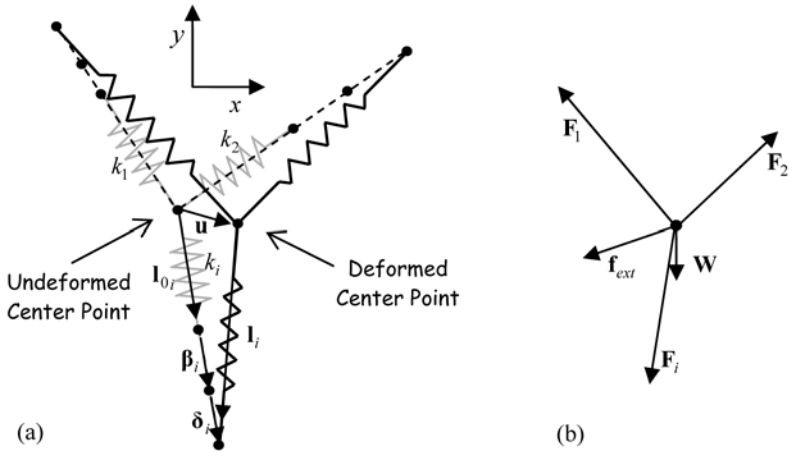


Fig. 3 Analytical model variable definition: (a) node and center point displacements, (b) free body diagram.

where z is a random number centered around 0 and the standard deviations are adjusted by the fraction ρ of the nominal value covering 6 standard deviations giving: $6\sigma_{l_i} = \rho l_{0_i}$, $6\sigma_{k_i} = \rho k_i$, and $6\sigma_{\zeta_i} = \rho \zeta_i$. Unless explicitly noted, values of $\rho = 30\%$ (99.7% of all values are within $\pm 15\%$ of the nominal values) are used everywhere in this paper

Two workspaces are computed: one with errors and one without. The metric used to quantify accuracy is the mean output error between each corresponding end-effector locations (needle tip):

$$\varepsilon = \frac{1}{n} \sum_{i=1}^{n} \sqrt{(\tilde{x}_{d_i} - x_{d_i})^2 + (\tilde{y}_{d_i} - y_{d_i})^2}, \tag{6}$$

where $\tilde{x}_{d_i}$, $\tilde{y}_{d_i}$ are the coordinates with errors, x_{d_i}, y_{d_i} are the coordinates without errors, and n is the number of points.

Sensitivity analysis results are shown in Table 1 for 5 combinations of parameter errors. Each combination is repeated 50 times on 54 randomly selected points for a total of 2700 points. Table 1 shows that spring stiffness and actuator stroke are the dominant parameters and that spring length can be neglected. This is explained by the fact that, in Eq. (3), spring length is multiplied by the difference in unit vectors $\mathbf{w}_i$ and $\mathbf{v}_i$ which appears to be very small, at least for this manipulator configuration.

Table 1 Sensitivity analysis of the manipulator model.

Parameter Errors			Mean Output Error		
Spring length (ρ)	Spring stiffness (ρ)	Actuator stroke (ρ)	Mean (mm)	Max. (mm)	Std. dev. (mm)
30%	0	0	0.03523	0.07773	0.01130
0	30%	0	0.7506	1.719	0.355
0	0	30%	0.9948	1.654	0.2875
0	30%	30%	1.333	2.613	0.4100
30%	30%	30%	1.353	2.651	0.4298

2.3 Parameter Identification

Parameter identification is done on spring stiffness and actuator strokes (k_i, ζ_i) with a multivariable optimization algorithm (Nelder–Mead method, Matlab's fminsearch function). Starting from the nominal parameter values, the algorithm minimizes the mean output error (Eq. (6)) by calibrating the parameters toward their true experimental values. The method is tested on simulated experimental data obtained using randomly generated spring stiffness and actuator strokes, see Eq. (5).

Parameter identification is performed for 2 to 100 calibration points taken randomly from the simulated data. The effect of unaccounted errors like dimensional, geometric, and measurement errors is studied by adding random noise to the calibration points. The noise follows a normal distribution with a fraction $\rho = 10\%$ (estimate of a high quality system).

Figure 4 illustrates how parameter identification improves model accuracy on 100 random points for 50 calibration points with noise. Figure 5a shows how parameter identification reduces the mean output error on 1000 random points. The figure shows that a minimum number of calibration points is needed for effective performance. Without noise, a minimum of $\sim$40 calibration points or 10% of the total number of points is needed to keep the error reduction around a factor of $\sim$40. With noise, the error reduction keeps increasing and reaches a maximum factor of $\sim$20 at 100 calibration points. Noise obviously reduces calibration performance and must be minimized. From these results, it appears that parameter identification has the potential to improve system accuracy by at least one order of magnitude with enough calibration points.

3 Experimental Results

Parameter identification is further illustrated on experimental points taken with the laboratory prototype of Figure 1. This device was not built with high precision standards and contains too many unaccounted errors (dimensional, geometric, measure-

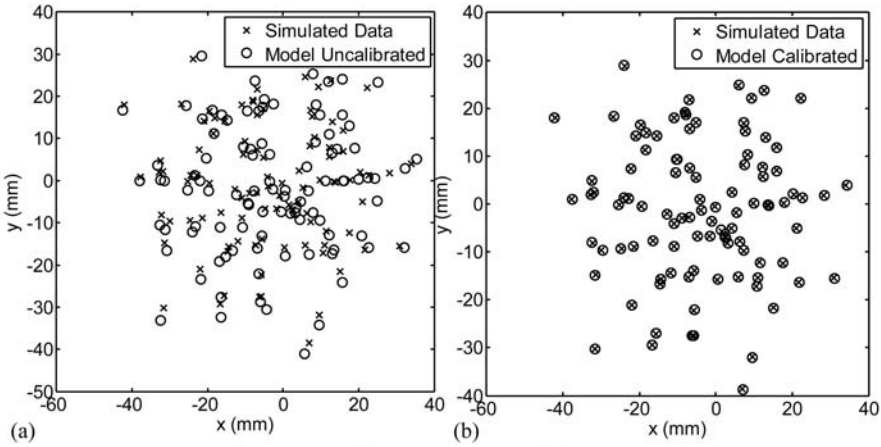


Fig. 4 Uncalibrated (a) and calibrated (b) model performance.

ment) to yield precise experimental data. However, it does provide some interesting results for illustrative purposes.

Here, the prototypes' 12 actuator strokes and 12 spring stiffness are identified using only 13 experimental points. The results are shown in Figure 5b for the same points. Points from the calibrated model are measurably closer to the experimental data than points from the uncalibrated model (see the dashed circles). Mean output error is reduced by a factor of 2.26, from 3.18 mm to 1.4 mm. This suggests that parameter identification can increase system accuracy even with limited experimental data.

4 Conclusion

This paper showed how parameter identification can be used to calibrate an elastically averaged parallel manipulator using binary actuators to improve accuracy. Parameter identification was done with an optimization algorithm using simulated experimental data and limited experimental data. The results show that the method has the potential to significantly improve model accuracy. Such methods can lead to self-calibrating, very high precision binary manipulators. Future works will include the development of a high precision prototype to conduct a thorough experimental validation.

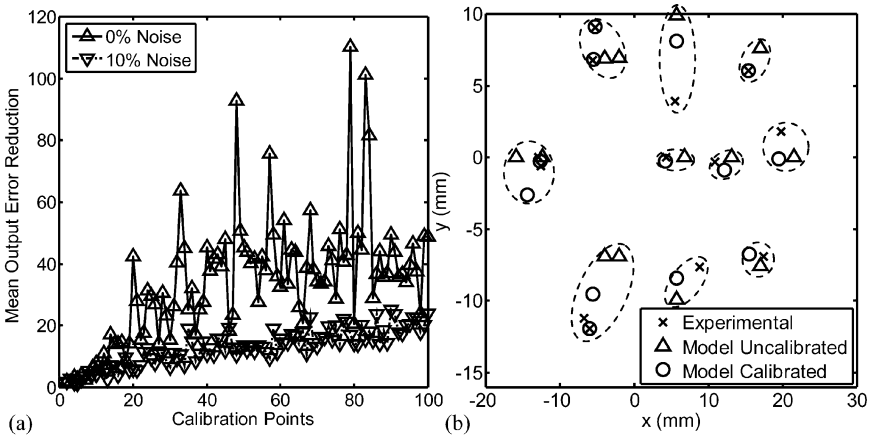


Fig. 5 (a) Mean output error reduction on simulated data. (b) Experimental results.

Acknowledgement

The works of Kenjiro Tadakuma and Lauren DeVita for the development of the experimental prototype is gratefully acknowledged.

References

Anderson, V. and Horn, R. (1967), Tensor arm manipulator design, *ASME, 67-DE-57*.

Chirikjian, G. and Burdick, J. (1995), The kinematics of hyper-redundant robot locomotion, *IEEE Transactions on Robotics and Automation* **11**(6), 781–793.

DeVita, L.M., Plante, J.S. and Dubowsky, S. (2007), The design of high precision parallel mechanisms using binary actuation and elastic averaging: With application to MRI cancer treatment, in *Proceeding of the 2007 IFToMM World Congress on Machines and Mechanisms*, Besançon, France.

Plante, J.S., Devita, L.M. and Dubowsky, S. (2007), A road to practical dielectric elastomer actuators based robotics and mechatronics: Discrete actuation, in *Proceedings SPIE Smart Structures and Materials 2007*, San Diego, USA.

Roth, B., Rastegar, J. and Scheinman, V. (1973), On the design of computer controlled manipulators, in *Proceedings First CISM-IFTMM Symposium on Theory and Practice of Robots and Manipulators*, pp. 93–113.

Tadakuma, K., DeVita, L.M., Plante, J.S., Shaoze, Y. and Dubowsky, S. (2008) The experimental study of a precision parallel manipulator with binary actuation: With application to MRI cancer treatment, in *Proceedings 2008 IEEE International Conference on Robotics and Automation*, Pasadena, USA.

Wingert, A., Lichter, M.D. and Dubowsky, S. (2002), On the kinematics of parallel mechanisms with bi-stable polymer actuators, in *Proceedings of the 8th International Symposium on Advances in Robot Kinematics*, Barcelona, Spain.

Stability Measure of Postural Dynamic Equilibrium Based on Residual Radius

Sébastien Barthélemy and Philippe Bidaud

Pyramide – Tour 55, Boîte courrier 173, 4 place Jussieu, 75252 Paris Cedex 05, France, e-mail: {barthelemy, bidaud}@robot.jussieu.fr

Abstract. This paper formalizes the robustness of a virtual human dynamic equilibrium through the residual radius of its admissible generalized force set. The admissible generalized force set is defined as the image of the contact force constraints (corresponding to the Coulomb model) in the generalized force space. This set is approximated by a polyhedron and its residual radius is computed using a linear program. The measure relevance is analyzed from experimental data of a sit-to-stand motion on which residual radius is evaluated.

Key words: equilibrium robustness, residual radius, human motion, postural stability.

1 Introduction

Generating whole body motion for virtual human that appears natural is a long standing problem in character animation, highligting the lack of a solution to Bernstein's redundancy problem. A lot of studies have identified invariants in various motions such as minimum jerk for reaching movements [1], minimum torque change, minimum muscle tension change, minimum motor command change, minimum of biological noise, etc.). Such models are well suited to generate reference trajectories which can be further modified to compensate expected perturbations in a feedforward way. Additionaly, sensory-driven feedback strategies are needed to cope with unexpected disturbances.

However, some strong perturbations cannot be compensated due to limitations in contact forces and joint torques. Some "distance to constraint violation" may therefore be monitored in order to ensure equilibrium by triggering adjustment motion when necessary.

The "quality" of equilibrium for humanoid robots is generally measured as the distance between some caracteristic point (ZMP, CdM projection, FRI) and the support polygon borders. Popovic [7] offers a comprehensive review of those caracteristic points and study their evolution during human walk. But these measures are only valid if the contacts are coplanar. Harada [3] extended the ZMP measure in or-

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 399–407.

der to handle situations where the upper-limbs are in contact with the environment. However, the limits in frictional contact forces are still not taken into account.

To overcome this restriction the force closure measures which have been developed in the past for grasp and fixture analysis can be +revisited. Wieber [8], then Hirukawa et al. [4] have proposed a formulation of the frictional constraints as a feasable contact wrench domain that has led to a more universal measure, however, to our knowledge, this measure has never been used to evaluate the set of possible human motions.

This paper is concerned with the quantification of a posture quality regarding the dynamical balance. We first consider a single body in contact and show how the set of feasable contact wrenches can be computed (Section 2). We then define its residual radius as a measure of the contact set quality (Section 3). This method is then generalized to the case of a virtual human in contact with its environment in Section 4. Finally, the relevance of the measure is evaluated on a sit-to-stand motion in Section 5.

2 Resistable and Applicable Contact Wrenches

Let us consider a rigid body b in contact at m points with bodies e_i $(i = 1, \ldots, m)$ from its environment.

For each contact point (let us say the i-th), we can define the frame $\{\mathrm{c}_i\}$ and the contact force $\boldsymbol{f}_{\mathrm{c}_i}$. Considering the punctual with friction contact model, when no sliding occurs, the overall friction force lies within a revolution cone:

$$\mathcal{F}_{\mathrm{c}_i} = \left\{ \boldsymbol{f}_{\mathrm{c}_i} : \boldsymbol{f}_{\mathrm{c}_i}^{\mathrm{T}} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \boldsymbol{f}_{\mathrm{c}_i} \leq \mu_i \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \boldsymbol{f}_{\mathrm{c}_i} \right\} \tag{1}$$

where μ_i is the static coefficient of friction. The set of wrenches that these contact forces can produce is called the set of *resistable contact wrench* $\mathcal{W}_{\mathrm{c}}$:

$$\mathcal{W}_{\mathrm{c}} = \left\{ \boldsymbol{w}_{\mathrm{c}} : \boldsymbol{w}_{\mathrm{c}} = G\boldsymbol{f}_{\mathrm{c}}; \forall \boldsymbol{f}_{\mathrm{c}} \in \mathcal{F}_{\mathrm{c}_1} \times \cdots \times \mathcal{F}_{\mathrm{c}_m} \right\} \quad \text{with } \boldsymbol{f}_{\mathrm{c}} = \begin{bmatrix} \boldsymbol{f}_{\mathrm{c}_1} \\ \vdots \\ \boldsymbol{f}_{\mathrm{c}_m} \end{bmatrix} \tag{2}$$

If the body b is a manipulated object, G is called the *grasp matrix*. A planar example is depicted in Figure 1, where the resistable contact wrench set is unbounded but does not span the entire $\mathbb{R}^3$.

The image of a revolution cone through a linear application (i.e. the grap matrix) cannot be computed directly. Therefore, we approximate it by a linear cone. Let us choose p regularly spaced unitary vectors on the Coulomb cone border of the i-th contact and call them $\boldsymbol{f}_{\mathrm{c}_i}^{j}$ (where $j = 1..p$). The set spanned by their positive combinations is a convex linear cone:

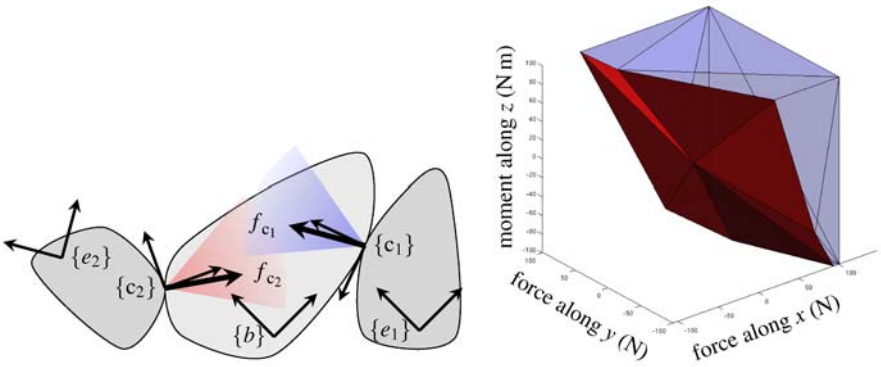


Fig. 1 A rigid body b in contact with two other bodies (left) and the set of resistable contact wrench $\boldsymbol{w}_b \in \mathcal{W}_c$ (right). The wrench is expressed in the frame $\{b\}$.

$$\mathcal{F}^*_{c_i} = \left\{ \boldsymbol{f} : \boldsymbol{f} = \sum_{j=1}^{p} a_j \boldsymbol{f}^j_{c_i}; a_j \geq 0 \right\} = \text{pos}\left(\boldsymbol{f}^1_{c_i}, \ldots, \boldsymbol{f}^p_{c_i}\right) \tag{3}$$

which (as any polyhedron) can also be defined as a generalized inequality:

$$\mathcal{F}^*_{c_i} = \left\{ \boldsymbol{f} : A_{c_i} \boldsymbol{f} \leq \boldsymbol{b}_{c_i} \right\} \tag{4}$$

where the rows of A_{c_i} are vectors chosen normal to the facets and outgoing. The set $\{\boldsymbol{f}^1_{c_i}, \ldots, \boldsymbol{f}^m_{c_i}\}$ is called the *vertice representation* of $\mathcal{F}^*_{c_i}$ and A_{c_i} and $\boldsymbol{b}_{c_i}$ are called its *half-space representation*.

The image $\mathcal{W}^*_c$ of $\mathcal{F}^*_{c_i} \times \cdots \times \mathcal{F}^*_{c_m}$ through the grasp map is also given by a positive combination:

$$\mathcal{W}^*_c = \left\{ \boldsymbol{w}_c : \boldsymbol{w}_c = \sum_{i=1}^{m} \left(S^{c_i} A d_b\right)^T \boldsymbol{f}_{c_i} \text{ where } \boldsymbol{f}_{c_i} \in \mathcal{F}^*_{c_i} \right\} \tag{5}$$

$$= \text{pos}\left(\boldsymbol{w}^1_{c_1}, \ldots, \boldsymbol{w}^j_{c_i}, \ldots, \boldsymbol{w}^p_{c_m}\right) \text{ where } \boldsymbol{w}^j_{c_i} = \left(S^{c_i} A d_b\right)^T \boldsymbol{f}^j_{c_i} \tag{6}$$

When a grasping problem is considered, the contact force is often actively controlled and also limited by additional (actuators-related) constraints. $\mathcal{W}_c$ is then replaced by the set of *applicable contact wrench* $\overline{\mathcal{W}}_c$:

$$\overline{\mathcal{W}}_c = \left\{ \boldsymbol{w}_c : \boldsymbol{w}_c = G \boldsymbol{f}_c; \forall \boldsymbol{f}_c \in \mathcal{F}_{c_1} \times \cdots \times \mathcal{F}_{c_m} : \chi\left(\boldsymbol{f}_c\right) = 1 \right\} \tag{7}$$

$$\text{where } \chi\left(\boldsymbol{f}_c\right) = \begin{cases} 1 \text{ if the additional constraints hold} \\ 0 \text{ otherwise} \end{cases} \tag{8}$$

A common choice for χ is to limit the total normal contact force, because its approximation leads to the following convex hull:

$$\overline{\mathcal{W}}_{\mathrm{c}}^{*} = \alpha \left\{ \boldsymbol{w} : \boldsymbol{w} = \sum_{i=1}^{m} \sum_{j=1}^{p} a_{i,j}\, \boldsymbol{w}_{\mathrm{c}_i}^{j};\, 0 \leq a_{i,j} \leq 1 \right\}$$

$$= \alpha \ \mathrm{conv}\left(\mathbf{0}, \boldsymbol{w}_{\mathrm{c}_1}^{1}, \ldots, \boldsymbol{w}_{\mathrm{c}_i}^{j}, \ldots, \boldsymbol{w}_{\mathrm{c}_m}^{p}\right) \tag{9}$$

where $\alpha \in \mathbb{R}^{+}$ is a scale factor. Mishra [6] proposed several other choices for χ.

3 Residual Ball and Radius

3.1 Definition

For a manipulation task, $\overline{\mathcal{W}}_{\mathrm{c}}$ (or $\mathcal{W}_{\mathrm{c}}$) is strongly related to the grasp quality: it can be used to check whether some expected external wrench $\boldsymbol{w}_e$ (for instance the wrench of gravity and inertial effects) is sustainable or not (simply check if $\boldsymbol{w}_e \in \overline{\mathcal{W}}_{\mathrm{c}}$). Moreover, the "further" $\boldsymbol{w}_e$ is from the boundary of $\overline{\mathcal{W}}_{\mathrm{c}}$, the "more robust" is the grasp. Kirkpatrick et al. [5] used the *residual radius* of $\overline{\mathcal{W}}_{\mathrm{c}}$ (or $\mathcal{W}_{\mathrm{c}}$) as a quantitative measure of this robustness.

Let us consider the largest hypersphere centered at $\boldsymbol{w}_e$ and fully contained inside $\overline{\mathcal{W}}_{\mathrm{c}}$. The hypersphere is called the *residual ball* and its radius the *residual radius*. Physically, the residual radius is the norm of the largest wrench which can be sustained in any direction.

3.2 Implementation

Finding the largest hypersphere included in the polyhedron $\mathcal{W}_{\mathrm{c}}^{*}$ can be written as the following linear program (LP): maximize r subject to $\boldsymbol{d}r \leq \boldsymbol{b} - A\boldsymbol{c}$, where A, $\boldsymbol{b}$ are $\mathcal{W}_{\mathrm{c}}^{*}$ half-space representation, $\boldsymbol{c}$ is the hypersphere center and $\boldsymbol{d}$ is computed from A as follows:

$$A = (a_{ij}), \quad \boldsymbol{d} = (d_i), \quad d_i = \sum_{j} \sqrt{a_{ij}^2}$$

In what follows, we make use of the polyhedral computation library *cddlib* written by Fukuda [2] to compute the half-space representations (A and $\boldsymbol{b}$).

The process of computing the residual radius can be summarized as follows:

1. for each contact, compute the vertices $\boldsymbol{f}_{\mathrm{c}_i}^{j}$,
2. compute each vertice image in the wrench space $\boldsymbol{w}_{\mathrm{c}_i}^{j}$,
3. compute the half-space representation of $\mathcal{W}_{\mathrm{C}}^{*}$,
4. compute the center $\boldsymbol{c}$ of the sphere (which is given by the external wrenches),
5. solve the LP to compute the residual radius.

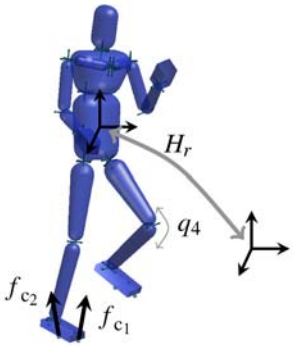


Fig. 2 The virtual human with 36 active dof and the homogeneous transformation matrix accounting for the position of the free floating reference body (the pelvis).

4 Virtual Human Dynamics

Let us consider the problem of controlling a multi-legged system such as a human. These systems are underactuated and thus rely on the contact forces to produce motion. Therefore, the limitations holding on the contact forces reduce the range of possible motions. We will show here how these limitations can be accounted for in the configuration space and then generalize the residual radius measure.

4.1 Equations of Motion

We model the virtual human as set of rigid bodies linked together by a tree-like structure. The mechanism has $n = 36$ degrees of freedom (dof). Let us denote $\boldsymbol{q} \in \mathbb{R}^n$ the generalized coordinates of these joints. The whole system can be viewed as a free flying robot, whose configuration is given by the vector $\boldsymbol{q}$ and the pose of a "reference" body H_r (see Figure 2). The generalized velocities and accelerations are then respectively $\boldsymbol{\nu} = \begin{bmatrix} \boldsymbol{v}_r \\ \dot{\boldsymbol{q}} \end{bmatrix}$ and $\dot{\boldsymbol{\nu}} = \begin{bmatrix} \dot{\boldsymbol{v}}_r \\ \ddot{\boldsymbol{q}} \end{bmatrix}$.

Assuming that all the contacts occur with static bodies, the condition of adhesion and non-lifting are written as $3m$ kinematic constaints:

$$\mathbf{0} = J_{\mathrm{c}}(\boldsymbol{q})\dot{\boldsymbol{\nu}} + \dot{J}_{\mathrm{c}}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{\nu} \tag{10}$$

where $J_{\mathrm{c}} \in \mathbb{R}^{(3m\times(n+6))}$ is the Jacobian of all the contact points.

The Newton–Euler equations of motion are given by

$$M(\boldsymbol{q})\dot{\boldsymbol{\nu}} + N(\boldsymbol{q}, \boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{g}(H_{\mathrm{r}}, \boldsymbol{q}) + S\boldsymbol{\gamma}(t) + J_{\mathrm{c}}(\boldsymbol{q})^{\mathrm{T}} \boldsymbol{f}_{\mathrm{c}} \tag{11}$$

where $M, N \in \mathbb{R}^{(n+6)\times(n+6)}$ are respectively the inertia and non-linear effects matrices, $\boldsymbol{g} \in \mathbb{R}^{n+6}$ is the vector of gravitational generalized forces, and $\boldsymbol{\gamma}(t) \in \mathbb{R}^n$

are the control input functions which are mapped with the constant matrix $S = \begin{bmatrix} 0_{6\times 6} \\ I_{n\times n} \end{bmatrix}$ into the actively controlled joints. Eventually, $\boldsymbol{f}_{\mathrm{c}}$ accounts for the contact forces, which can also be viewed as the Lagrangian multipliers of the contact-related kinematic constraints.

4.2 Resistable Generalized Contact Force

When considering the virtual human equilibrium, we are interested in the impact of the contact force limitations on the set of admissible generalized forces. In Section 2, we mapped the Coulomb cones through the grasp matrix into the resistable contact wrench, we may extend this method by mapping the friction cone through the transposed contact point Jacobian into the generalized force space $\mathcal{T}_{\mathrm{c}}$. Its linear approximation $\mathcal{F}^*_{\mathrm{c}_m}$ is given by:

$$\mathcal{T}^*_{\mathrm{c}} = \left\{\tau : \tau = J_{\mathrm{c}}(\boldsymbol{q})^{\mathrm{T}} \boldsymbol{f}_{\mathrm{c}} : \boldsymbol{f}_{\mathrm{c}} \in \mathcal{F}^*_{\mathrm{c}_i} \times \ldots \times \mathcal{F}^*_{\mathrm{c}_m}\right\} \tag{12}$$

$$= \mathrm{pos}\left(\boldsymbol{\tau}^1_{\mathrm{c}_1}, \ldots, \boldsymbol{\tau}^j_{\mathrm{c}_i}, \ldots, \boldsymbol{\tau}^p_{\mathrm{c}_m}\right) \text{ where } \boldsymbol{\tau}^j_{\mathrm{c}_i} = J_{\mathrm{c}_i}(\boldsymbol{q})^{\mathrm{T}} \boldsymbol{f}^j_{\mathrm{c}_i} \tag{13}$$

$\mathcal{T}^*_{\mathrm{c}}$ is a polyhedron into $\mathbb{R}^{(n+6)}$ which represents the set of generalized contact forces which can be sustained by the contact forces. One can compute its residual radius as in Section 3.2. However, it is harder to give a physical meaning to the radius in this space. Moreover, computing the half-space representation of this polyhedron in this high-dimensional space is time consuming. Therefore, we will only consider a subset of the generalized forces. As the ground force limitations are much more restrictive for the unactuaded DOFs than for the actuated ones (because in the latter case, the actuators can compensate for the restrictions), we choose to consider the generalized forces corresponding to the unactuated DOFs. In the case of a virtual human, the only unactuated DOFs are the ones positioning the root body (the bust), therefore the 6 corresponding generalized forces consist of a wrench and the measure developed for the grasping can be applied directly by replacing G with the first 6 lines of $J_{\mathrm{c}}(\boldsymbol{q})^{\mathrm{T}}$.

Let us multiply the dynamic equilibrium equation with $P^{\perp} = \left[I_{6\times 6}\ 0_{36\times 36}\right]$, it becomes:

$$P^{\perp}\left(M(\boldsymbol{q})\dot{\boldsymbol{v}} + N(\boldsymbol{q}, \boldsymbol{v})\boldsymbol{v} - \boldsymbol{g}(H_{\mathrm{r}}, \boldsymbol{q})\right) = P^{\perp} J_{\mathrm{c}}(\boldsymbol{q})^{\mathrm{T}} \boldsymbol{f}_{\mathrm{c}} \tag{14}$$

One can then compute the residual radius measure on the resulting 6-dimensional space as follows:

1. for each contact, compute the vertices $\boldsymbol{f}^j_{\mathrm{c}_i}$,
2. compute each vertice image through $P^{\perp} J_{\mathrm{c}}^{\mathrm{T}}$,
3. compute the half-space representation of the polyhedron,
4. compute the center $\boldsymbol{c} = P^{\perp}\left(M(\boldsymbol{q})\dot{\boldsymbol{v}} + N(\boldsymbol{q}, \boldsymbol{v})\boldsymbol{v} - \boldsymbol{g}(H_{\mathrm{r}}, \boldsymbol{q})\right)$, of the sphere,
5. solve the LP to compute the residual radius.

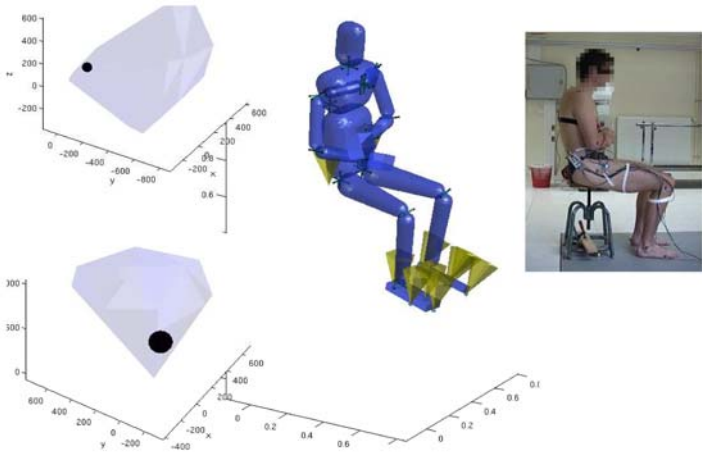


Fig. 3 (Right) The virtual human at the beginning of the sit-to-stand motion, in contact with the ground and the seat. The linearized friction cones appear in yellow. (Top left) Resistable contact moment (numerical values in Nm). (Bottom left) Resistable contact force (numerical values in N).

4.3 Evaluation on Human Motion

In order to evaluate the relevance of this measure, we computed it on a sit-to-stand motion recorded with an optical motion tracking system.

The subjet is initially sitting on a stool, with its bottom in contact with the stool and each foot in contact with the ground. We modeled the contact between each foot and the ground as well as between each thick and the stool with respectively 4 and 1 punctual with friction contacts (see Figure 3).

We computed the measure as detailed in Section 4.2, using the captured motion as input. We choose the frame $\{b\}$ to be parallel to the inertial frame with its origin at the virtual human center of mass. The corresponding set of resistable generalized contact force space at the beginning of the motion is shown in Figure 3 together with the residual ball.

Figure 4 shows the evolution of the residual radii of both torque and force components during the motion. One can notice that:

- there is a discontinuity when the seat contacts lift off,
- the radii present a maximum when the normal force is the highest, that is when the vertical acceleration is maximal;
- the radii reach low level, indicating that the system is really near to tip.

This evolution is very conform to expectations and suggests that the proposed measure can be successfully used to monitor the equilibrium stablility during a motion.

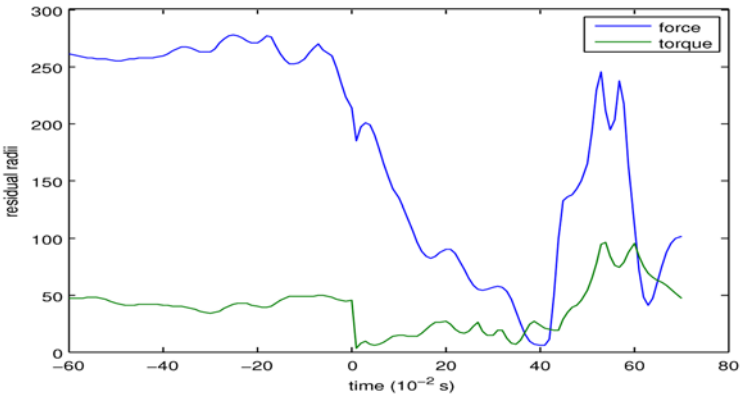


Fig. 4 Residual radii of the contact wrench (force and torque) evaluated over the time during a sit-to-stand motion. The origin of time is chosen at the lift-off.

5 Conclusion and Perspectives

We defined a measure of dynamic equilibrium robustness and showed its relevance on a sit-to-stand motion. Further work is needed to study how sensitive the measure is to errors in contact parameters estimation. Checking the measure behaviour against other human motions, typically involving contacts between the upper limbs and the environment would also be useful. Eventually, we could use the measure to adjust a virtual human posture in order to improve its equilibrium robustness.

References

1. Flash, T. and Hogan, N., The coordination of arm movements: An experimentally confirmed mathematical model. *The Journal of Neuroscience* **5**(7), 1688–1703, 1985.
2. Fukuda, K., cdd Software, 1993. cdd.c: C-implementation of the double description method for computing all vertices and extremal rays of a convex polyhedron given by a system of linear inequalities. Departement of Mathematics, Swiss Federal Institute of Technology, Lausanne, Switzerland, 1993.
3. Harada, K., Hirukawa, H., Kanehiro, F., Fujiwara, K., Kaneko, K., Kajita, S. and Nakamura, M., Dynamical balance of a humanoid robot grasping an environment. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 2, pp. 1167–1173, 2004.
4. Hirukawa, H., Shizuko, H., Harada, K., Kajita, S., Kaneko, K., Kanehiro, F., Fujiwara, K. and Morisawa, M., A universal stability criterion of the foot contact of legged robots – adios zmp. In *IEEE International Conference on Robotics and Automation*, pp. 1976–1983, 2006.
5. Kirkpatrick, D., Mishra, B. and Yap, C.-K., Quantitative Steinitz's theorems with applications to multifingered grasping. *Discrete and Computational Geometry* **7**(1), 295–318, 1992.
6. Mishra, B., Grasp metrics: Optimality and complexity. In K. Goldberg, D. Halperin, J. Latombe, and R. Wilson (Eds.), *Algorithmic Foundations of Robotic*, A.K. Peters, Wellesley, MA, pp. 137–166, 1995.

7. Popovic, M.B., Goswami, A. and Herr, H., Ground reference points in legged locomotion: Definitions, biological trajectories and control implications. *International Journal of Robotics Research* **24**(12), 1013–1032, 2005.
8. Wieber, P.B., Modélisation et commande d'un robot marcheur anthropomorphe. PhD Thesis, Mines de Paris, 2000.

KINEMATIC ANALYSIS AND WORKSPACE

A Geometrical Characterization of Workspace Singularities in 3R Manipulators

M. Husty[1], E. Ottaviano[2] and M. Ceccarelli[2]

[1]*Institute for Basic Sciences in Engineering, University of Innsbruck, Innsbruck, Austria, e-mail: manfred.husty@uibk.ac.at*
[2]*LARM: Laboratory of Robotics and Mechatronics – DiMSAT – University of Cassino, Cassino (Fr), Italy, e-mail: {ottaviano, ceccarelli}@unicas.it*

Abstract. In this paper we present an algorithm, based on a level set representation of a cross-section of the Cartesian workspace of 3R regional manipulators, which is useful to show clearly the nature of the cusps and double points on the boundary. Furthermore it is shown that singularities of the level set surface (graph of the level set) characterize non-generic manipulators and we demonstrate the non-singular posture change ability of cuspidal manipulators with help of the level set surface.

Key words: serial manipulators, workspace, singularities, level set, posture change.

1 Introduction

Workspace analysis of serial manipulators is of great interest since the workspace geometry can be considered a fundamental issue for manipulator design, robot placement and trajectory planning. Great attention has been addressed to manipulators' classification as function of geometric singularities [1, 3, 7, 8, 10, 11].

Cuspidal manipulators are said to be non-singular posture changing because of the presence of cusps on the boundary curve [11]. In this paper we give a new insight into this phenomenon using the level set representation of the workspace cross-section introduced in [4]. It is shown that singularities of the graph of the level set correspond to non-generic manipulators. The set of singularities of the manipulator itself corresponds to contour curve on the level set surface. It is believed that this representation yields a lot of insight into the internal structure of the workspace cross-section and especially shows nicely why cuspidal manipulators have the non-singular posture change ability.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 411–418.

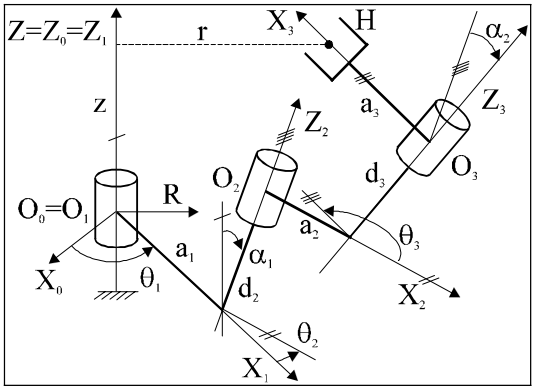


Fig. 1 A kinematic scheme for a general 3R manipulator .

2 Level Set Workspace Analysis for 3R Manipulators

A general 3R manipulator is sketched in Figure 1, in which the kinematic parameters are denoted by the standard Hartenberg and Denavit (H-D) notation. Without loss of generality the base frame is assumed to be coincident with $X_1Y_1Z_1$ frame when $\theta_1 = 0, a_0 = 0, d_1 = 0$. The point H is placed on the X_3 axis at a distance a_3 from O_3, as shown in Figure 1. The general 3R manipulator is described by the H-D parameters $a_1, a_2, d_2, d_3, \alpha_1, \alpha_2, \theta_i$ $(i = 1, \ldots, 3)$, as shown in Figure 1. r is the distance of point H from the Z_1-axis and z is the axial reach, both are expressed in H-D parameters. The position workspace of the 3R manipulator can be obtained by a θ_1 rotation of the generating torus that is traced from H by full revolution of θ_2 and θ_3.

The workspace boundary of a general 3R manipulator can be expressed as function of radial and axial reaches, r and z respectively, with respect to the base frame. The reaches r and z can be evaluated as functions of coordinates of the position vectors in the form

$$r_0 = (H_0^x)^2 + (H_0^y)^2 = (H_1^x \cos\theta_1 - H_1^y \sin\theta_1)^2 + (H_1^x \sin\theta_1 + H_1^y \cos\theta_1)^2,$$
$$z = H_0^z, \tag{1}$$

which can be equivalently expressed in the form

$$r_0 = (H_1^x)^2 + (H_1^y)^2, \quad z = H_1^z, \tag{2}$$

in which H_i is the position vector with respect to reference frame i.

Equation (2) represents a 2-parameter family of curves, whose envelope gives the cross-section workspace contour in a cross-section plane as a function of the H-D parameters that can be used to express the vector components H_1^x, H_1^y and H_1^z in the form of a ring equation [2]. In the following this two-parameter set of functions Eq. (2) is interpreted as a level set. The level set of a differentiable function

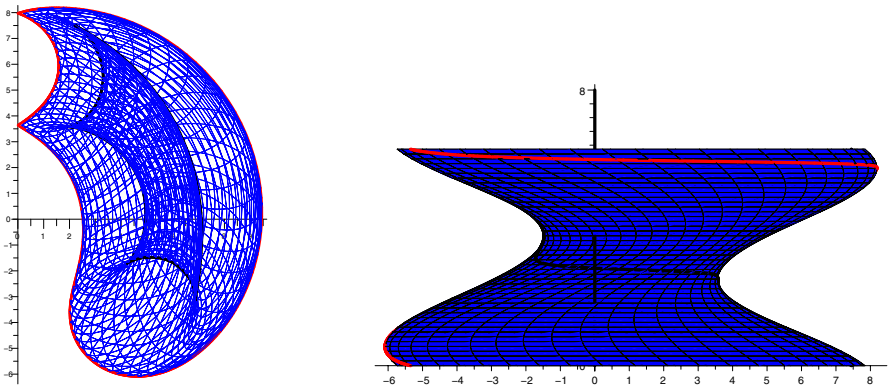


Fig. 2 A kinematic scheme of level set workspace representation for a general 3R manipulator: top and front view.

$f : \mathbb{R}^n \to \mathbb{R}$ corresponding to a real value c is the set of points [9]:

$$\{(x_1, \ldots, x_n) \in \mathbb{R}^n : f(x_1, \ldots, x_n) = c\}. \tag{3}$$

The level set interpretation has been successfully applied to the workspace analysis of 3R manipulators in [4–7], because it provides a lot of geometric insight into the internal structure of the cross-section of the manipulators workspace.

The level sets belonging to constant values of θ_3 are curves in the rz-plane. Therefore, this one parameter set of curves can be viewed as the contour map of a surface S, which conveniently can be used to analyze the workspace of a manipulator. Using Eq. (2) the surface S is defined via the functions

$$X^2 = r_0^2, \quad Y = z, \quad Z = \tan\frac{\theta_3}{2}. \tag{4}$$

By performing the half-tangent substitution $v = \tan\theta_3/2$ in Eq. (4) and eliminating the parameter v one can obtain an implicit equation of the surface S: $F(X, Y, Z) = 0$. The surface S is an algebraic surface of degree 20 as shown in [6]. Geometrically, S is generated by taking a cross-section of the workspace that is parameterized by θ_2 and θ_3 and explode the overlapping level set curves ($\theta_3 = \text{const.}$) in the direction of the Z-axis, as shown in the example in Figure 2. The major advantage of this procedure is that on S one can see clearly the number of solutions of the Inverse Kinematics (IK) belonging to one point of the workspace cross-section. In particular one can identify the regions with one, two, or four solutions as delimited by contours of the envelope boundary. In Figure 2 this is shown for a general illustrative case with H-D parameters $\alpha_1 = \pi/3$, $\alpha_2 = \pi/2$, $a_1 = 1.3$, $a_2 = 5$, $a_3 = 2.5$, $d_1 = 2.1$, $d_2 = 2.3$. In Figure 2 on the left-hand side the level set curves are shown in the cross-section plane and the right-hand side shows the level set surface S in a front view.

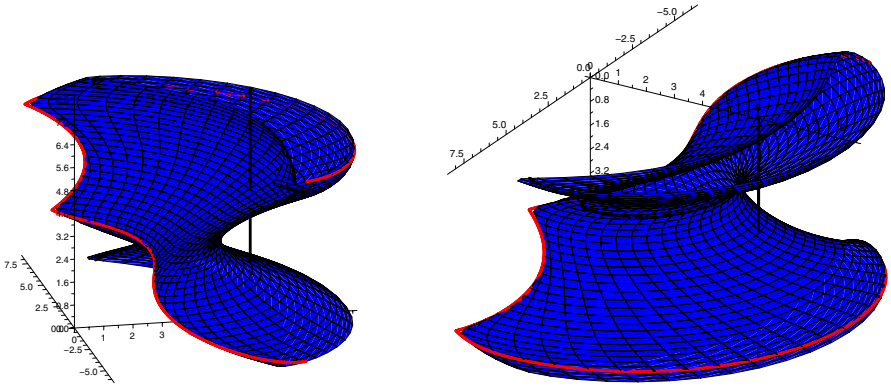


Fig. 3 Two axonometric views of the level set workspace representation for a general 3R manipulator:

In the workspace cross-section two different one-parameter sets of level-curves can be traced as function of $\theta_3 = \text{const.}$ and $\theta_2 = \text{const.}$, respectively. In Figure 3 the corresponding surface S is displayed in a three-dimensional view. Geometrically the level set curves in the cross-section in Figure 2 (left) are the orthogonal projections of the intersection curves with planes $Z = \text{const.}$ and the surface S onto the XY-plane. The level set curves for $\theta_3 = \text{const.}$ in Figure 2 (right) are therefore the horizontal parameter lines. Additionally, in Figure 3 we have displayed a gross line parallel to the Z-axis ($X = 6, Y = 0, Z = Z$). This line shows clearly four intersection points with the surface S (in the right figure the surface is upside down!). Therefore, the corresponding point $X = 6, Y = 0$ in the level set plane in Figure 2 (left) corresponds to a four fold solution of the IK.

3 Singularities, Cusps and Double Points

Singularities of the manipulator can be easily visualized within the setting of the level set surface: They are either singularities of S or they must be on the boundary of the level set itself. Geometrically the boundary of the level set in the XY-plane corresponds to those points on S which have a tangent plane being in edge view with respect to the level set plane XY. Or with other words: whenever a Z-axis parallel line is tangent to S then the tangent point of this line corresponds to a boundary point of the level set and therefore is also a singular point of the manipulator workspace. Points on a surface having tangent planes in edge view with respect to a projection direction are called contour points. The set of all contour points on S is called the contour curve c. The orthogonal projection of the contour curve onto the level set plane is the boundary curve of the level set. As each projection ray is tangent to S the tangent point itself corresponds to a twofold solution of the IK.

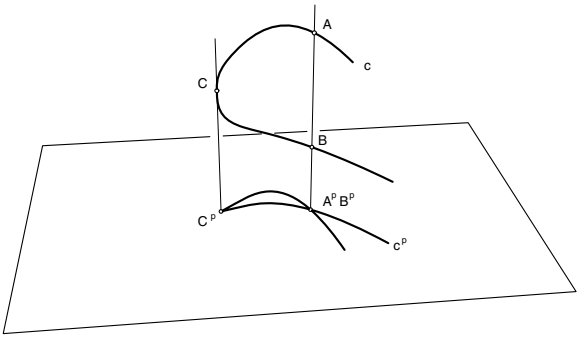


Fig. 4 Singularities of projected curves.

3.1 Singularities of the Boundary Curve

A cusp of the projected contour curve (boundary of the level set) comes from the following geometric feature: whenever a projection ray is tangent in a point of the (space)curve, then the projected curve has a cusp in the projection of this point. In Figure 3 point C is projected to a cusp C^p.

A double point of the boundary comes from the following geometric property: whenever a projection ray is a secant of the contour curve then the (two) intersection points of the ray with c map to a double point of the boundary. In Figure 3 points A, B are projected to a double point $A^p = B^p$.

3.2 Singularities of Surface S

Singularities of the level set surface S can be computed as functions of H-D parameters. The occurrence of singularities on S makes the corresponding manipulator non-generic [1]. The equation of surface S consists of two parts S_1 and S_2, as shown in [4, 6]. Zeros of the set of equations $S_1 = 0$, $S_2 = 0$; $\partial S_2/\partial X = 0$; $\partial S_2/\partial Y = 0$ and $\partial S_2/\partial Z = 0$ identify the singularities of the surface S. This set of equations is equivalent to three polynomials in the form

$$P_1 = d_3^2 \sin^2(\alpha_2) + (a_3 - a_2)^2, \quad P_2 = c_4 \cos^4(\alpha_2) + c_2 \cos^2(\alpha_2) + c_0,$$
$$P_3 = (a_3^2 - a_2^2)\cos^2(\alpha_2) - d_3 \sin^2(\alpha_2), \tag{5}$$

in which the coefficients c_i are given by

$$c_0 = [(a_2^2 + a_3^2) + d_3^2][(a_2^2 - a_3^2) + d_3^2], \quad c_2 = 2[(a_3^2 + d_3^2)^2 + a_2^2(d_3^2 - a_3^2)],$$
$$c_4 = (a_3^2 + d_3^2)^2. \tag{6}$$

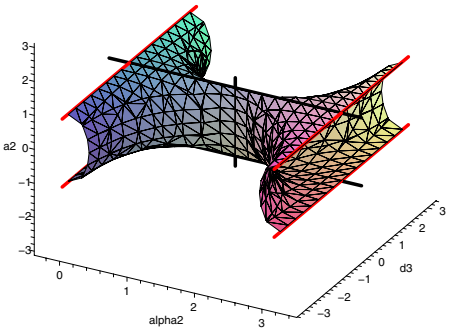


Fig. 5 Singularities of projected curves.

In particular, the polynomials P_i vanish for the following conditions:

- P_1 is equal to zero if $a_2 = a_3$ and either $d_3 = 0$ or $\alpha_2 = 0$.
- P_2 is equal to zero if $\alpha_2 = \pi/2$ (because $c_4 = 0$ has no real solution) and $c_0 = 0$. From this follows $\alpha_2 = \pi/2$, $a_2 = \pm a_3$ and $d_3 = 0$.
- The condition $P_3 = 0$ gives the most general case for the singularities of the level set surface S. In particular, since a_3 cannot be zero, one can set $a_3 = 1$ to obtain a 2-parameter set of possible design conditions. This set is represented by a surface Φ in the three-dimensional affine design (sub)space with coordinates α_2, a_2, d_3. The surface Φ representing the singularity condition $P_3 = 0$ when $a_3 = 1$ is displayed in Figure 5. In addition, it can be noted that the zeros of P_1 and P_2 are contained in P_3. These zero conditions are represented by the gross lines in Figure 5.

All singularities of S have the kinematic meaning that the end-effector point H is placed on the second rotation axis. An arbitrary rotation about this axis does not move the manipulator out of the singularity. Furthermore it should be noted that a singularity of a surface always belongs to the contour curve and therefore every singularity of S belongs in the projection to the boundary of the level set. In some cases this projection of surface singular points may lead to an acnode, which also belongs to the boundary. In Figure 6 we show for example the level set and the corresponding level set surface S of a non-generic manipulator having two singularities on the level set surface.

4 Non-Singular Posture Change and Level Set Surface

It was shown in previous papers (see e.g. [12]) that cuspidal robots have the ability of posture change without crossing a singularity. Using the level set surface and the explanation of cusps and double points on the boundary of the level set itself, it is quite natural to understand how this can be possible. In Figure 7 this is demonstrated with the same H-D parameters as Figure 2. The two points H_1 and H_2 belong to the

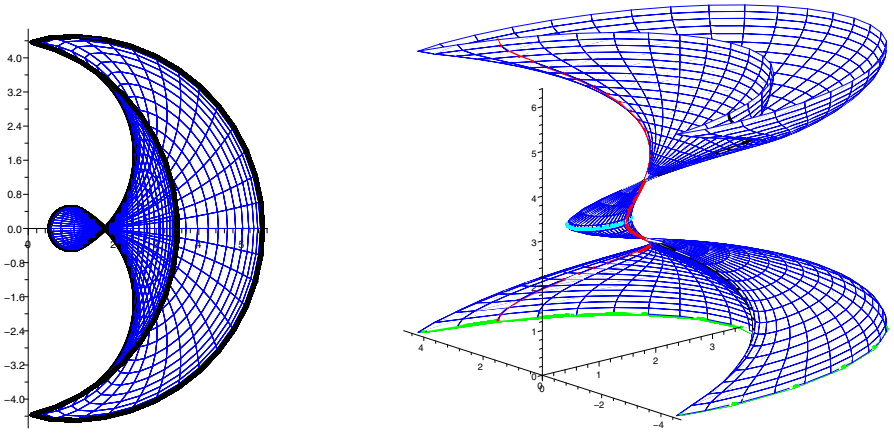


Fig. 6 Level set and level set surface of a manipulator with H-D parameters $\alpha_1 = \alpha_2 = \pi/s$, $a_1 = 1, a - 2 = 2, a_3 = 5/2, d_2 = d_3 = 0$.

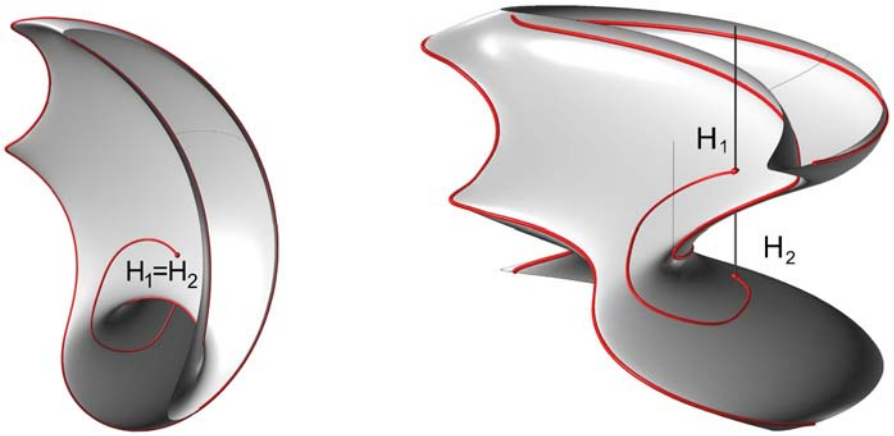


Fig. 7 Non-singular posture change.

same end-effector position. On the level set surface they are apart. They easily can be connected with a curve on S that does not cross the contour curve on the surface. Therefore the path of the end effector corresponding to this curve is singularity free. On the left figure one can see clearly that on the level set plane (projection plane) the path seems to cross the boundary curve. From the three-dimensional figure on the right it is obvious that this intersection is only apparent. To make the path on S such that it will be singularity free on just has to take care of the contour curve and the pullback of the cusp to S. The preimage of the cusp is the point on the contour curve having a tangent parallel to the projection rays. This tangent is the light parallel line to the projection ray connecting H_1 and H_2.

5 Conclusions

In this paper we have shown a level set representation of the workspace cross-section of 3R manipulators as a useful means to identify workspace singularities and to clearly show the nature of cusps and double points on the cross-section boundary of the workspace of three-revolute manipulators. In particular, the proposed formulation has been exploited by the level set analysis to characterize the non-singular posture changing ability of cuspidal manipulators. Furthermore it was shown that level set surfaces having singularities characterize non-generic manipulators.

References

1. Burdick, J.W., A classification of 3R regional manipulator singularities and geometries, *Mechanism and Machine Theory* **30**(1), 71–89, (1995).
2. Ceccarelli, M., On the workspace of 3R robot arms, in *Proc. 5th IFToMM Int. Symp. on Theory and Practice of Mechanism*, Bucharest, Vol. II-1, pp. 37–46 (1989).
3. Ottaviano, E., Ceccarelli, M., Lanni, C., A characterization of ring void in workspace of three-revolute manipulators, in *Proc. 10th World Congress on the Theory of Machines and Mechanisms*, Oulu, Finland, Vol. 3, pp. 1039–1044 (1999).
4. Ottaviano E., Husty M., Ceccarelli M. (2004), A Cartesian representation for the boundary workspace of 3R manipulators, in *On Advances in Robot Kinematics*, J. Lenarčič and C. Galletti (Eds.), Kluwer, Dordrecht, pp. 247–254 (2004).
5. Ottaviano, E., Husty, M., Ceccarelli, M., Identification of the workspace boundary of a general 3-R manipulator, *ASME Journal of Mechanical Design* **128**(1), 236–242 (2006).
6. Ottaviano, E., Husty, M., Ceccarelli, M., Level-set method for workspace analysis of serial manipulator, in *Advances in Robot Kinematics: Mechanisms and Motion*, J. Lenarčič and B. Roth (Eds.), Springer, Dordrecht, pp. 307–314 (2006).
7. Ottaviano, E., Husty, M., Ceccarelli, M., Workspace topologies of industrial 3R manipulators, *International Journal of Advanced Robotic Systems* **4**(3), 355–364 (2007).
8. Parenti-Castelli, V., Innocenti, C., Position analysis of robot manipulators: Regions and subregions, in *Advances in Robot Kinematics*, Ljubljana, pp. 150–158 (1988).
9. Sethian, J.A., *Level-Set Methods and Fast Marching Methods*, Cambridge University Press, Cambridge (1996).
10. Zein, M., Wenger, P., Chablat, D., An exhaustive study of the workspaces topologies of all 3R orthogonal manipulators with geometrical simplifications, *Mechanism and Machine Theory* **41**(8), 971–986 (2006).
11. Wenger, P., Some guidelines for the kinematic design of new manipulators, *Mechanism and Machine Theory* **35**(3), 437–449 (2000).
12. Wenger, P., Uniqueness domains and regions of feasable paths for cuspidal manipulators, *IEEE Transactions on Robotics* **20**(4), 745–750 (2004).

Kinematic Analysis of a Planar Tensegrity Mechanism with Pre-Stressed Springs

Carl D. Crane III[1], Jahan Bayat[1], Vishesh Vikas[1] and Rodney Roberts[2]

[1]*Center for Intelligent Machines and Robotics, University of Florida, Gainesville, FL 32611, USA, e-mail: {ccrane, bayat, vishesh}@ufl.edu*
[2]*Department of Electrical Engineering, Florida State University, Tallahassee, FL 32310, USA, e-mail: rroberts@eng.fsu.edu*

Abstract. This paper presents the equilibrium analysis of a planar tensegrity mechanism. The device consists of a base and top platform that are connected in parallel by one connector leg (whose length can be controlled via a prismatic joint) and two spring elements whose linear spring constants and free lengths are known. The paper presents three cases: (1) the spring free lengths are both zero, (2) one of the spring free lengths is zero and the other is non-zero, and (3) both free lengths are non-zero. The purpose of the paper is to show the increase in complexity that results from non-zero free lengths. It is shown that 6 equilibrium configurations exist for case 1, 20 equilibrium configurations exist for case 2, and 62 configurations exist for case 3.

Key words: planar mechanisms, tensegrity.

1 Introduction

The word tensegrity is a combination of the words tension and integrity (Edmondson, 1987; Fuller, 1975). Tensegrity structures are spatial structures formed by a combination of rigid elements in compression (struts) and connecting elements that are in tension (ties). No pair of struts touch and the end of each strut is connected to three non-coplanar ties (Yin et al., 2002). The entire configuration stands by itself and maintains its form solely because of the internal arrangement of the struts and ties (Tobie, 1976).

The development of tensegrity structures is relatively new and the works related have only existed for approximately 25 years. Kenner (1976) established the relation between the rotation of the top and bottom ties. Tobie (1976) presented procedures for the generation of tensile structures by physical and graphical means. Yin (2002) obtained Kenner's results using energy considerations and found the equilibrium position for unloaded tensegrity prisms. Stern (1999) developed generic design equations to find the lengths of the struts and elastic ties needed to create a desired geometry for a symmetric case. Knight (2000) addressed the problem of stability of tensegrity structures for the design of deployable antennae.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 419–427.

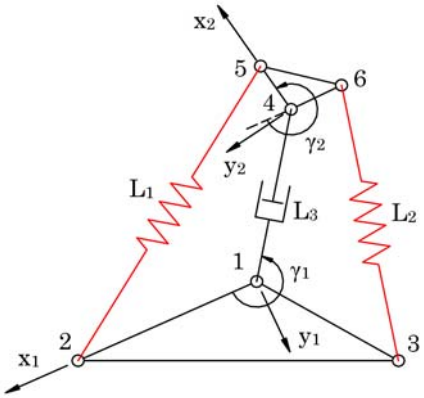


Fig. 1 Compliant mechanism.

2 Problem Statement

The mechanism to be analyzed here is shown in Figure 1. The top platform (indicated by points 4, 5, and 6) is connected to the base platform (indicated by points 1, 2, and 3) by two spring elements whose lengths are L_1 and L_2 and by a variable length connector whose length is referred to as L_3. Although this does not match the exact definition of tensegrity, the device is prestressed in the same manner as a tensegrity mechanism. The exact problem statement is as follows:

Given:

L_{12}	distance between points 1 and 2
p_{3x}, p_{3y}	coordinates of point 3 in coord. system 1
L_{45}	distance between points 4 and 5
p_{6x}, p_{6y}	coordinates of point 6 in coord. system 2
L_3	distance between points 1 and 4
k_1, L_{01}	spring constant and free length of spring 1
k_2, L_{02}	spring constant and free length of spring 2

Find: All static equilibrium configurations.

It is apparent that since the length L_3 is given, the device has two degrees of freedom. Thus there are two descriptive parameters that must be selected in order to define the system. For this analysis, the descriptive parameters are chosen as the angles γ_1, the angle between the x_1 axis and the line defined by points 1 and 4, and γ_2, the angle between the x_1 axis and the line defined by points 4 and 5. No other set of tested parameters yielded a less complicated solution than is presented here.

3 Solution Approach

Two possible solution approaches were considered, i.e. (1) satisfy force and moment conditions for equilibrium and (2) obtain configurations of minimum potential energy. Each approach was found to realize the same set of constraint equations. As such, obtaining the condition for force and moment balance is presented here.

The first step of the analysis is to determine the coordinates of the six points in terms of the base coordinate system as expressed in terms of the descriptive parameters γ_1 and γ_2. The coordinates of the three points in the base may be written with respect to the $x_1 y_1$ coordinate system as

$$\mathbf{P}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \mathbf{P}_2 = \begin{bmatrix} L_{12} \\ 0 \end{bmatrix}, \quad \mathbf{P}_3 = \begin{bmatrix} p_{3x} \\ p_{3y} \end{bmatrix}. \tag{1}$$

The coordinates of the three points in the top platform may be written as

$$\mathbf{P}_4 = \begin{bmatrix} L_3 c_1 \\ L_3 s_1 \end{bmatrix}, \quad \mathbf{P}_5 = \begin{bmatrix} L_3 c_1 + L_{45} c_2 \\ L_3 s_1 + L_{45} s_2 \end{bmatrix}, \quad \mathbf{P}_6 = \begin{bmatrix} L_3 c_1 + p_{6x} c_2 - p_{6y} s_2 \\ L_3 s_1 + p_{6x} s_2 + p_{6y} c_2 \end{bmatrix}, \tag{2}$$

where s_i and c_i, $i = 1, 2$, represent the sine and cosine of the angle γ_i.

A free body diagram of the top platform indicates that the sum of the forces along the three connector lines must equal zero at equilibrium. The unitized Plücker coordinates of a connector line can be obtained as

$$\$ = \frac{1}{d_i} \begin{bmatrix} x_t - x_b \\ y_t - y_b \\ [(x_b \mathbf{i} + y_b \mathbf{j}) \times ((x_t - x_b)\mathbf{i} + (y_t - y_b)\mathbf{j})] \cdot \mathbf{k} \end{bmatrix}, \tag{3}$$

where (x_t, y_t) and (x_b, y_b) are respectively the coordinates of the points on the top and bottom platforms that are on the line and di is the distance between the points that is calculated as

$$d_i^2 = (x_t - x_b)^2 + (y_t - y_b)^2. \tag{4}$$

For the two spring connectors, $i = 1, 2$ and

$$d_1^2 = 2L_3 L_{45}(c_1 c_2 + s_1 s_2) - 2L_{12}(L_3 c_1 + L_{45} c_2) + L_{12}^2 + L_3^2 + L_{45}^2, \tag{5}$$

$$\begin{aligned} d_2^2 = {} & 2L_3(p_{6x} s_2 + p_{6y} c_2 - p_{3y}) s_1 + 2(p_{6y} p_{3x} - p_{6x} p_{3y} - L_3 p_{6y} c_1) s_2 \\ & + 2L_3(p_{6x} c_2 - p_{3x}) c_1 + L_3^2 + p_{3x}^2 + p_{3y}^2 + p_{6x}^2 + p_{6y}^2. \end{aligned} \tag{6}$$

The force in each of the springs can be written as

$$f_1 = k_1(d_1 - L_{01}), \tag{7}$$

$$f_2 = k_2(d_2 - L_{02}). \tag{8}$$

The summation of the three forces that are acting on the top platform may be written as

$$f_1\$_1 + f_2\$_2 + f_3\$_3 = 0. \tag{9}$$

It is interesting to note that this equation implies that a necessary condition for static equilibrium is that the three line coordinates are linearly dependent.

The three line coordinates which were defined by (3) may now be written as

$$\$_i = [l_i, m_i, n_i]^T, \quad i = 1, \ldots, 3. \tag{10}$$

Equation (9) may be rearranged as

$$f_3 \begin{bmatrix} l_3 \\ m_3 \\ n_3 \end{bmatrix} = - \begin{bmatrix} f_1 l_1 + f_2 l_2 \\ f_1 m_1 + f_2 m_2 \\ f_1 n_1 + f_2 n_2 \end{bmatrix}. \tag{11}$$

In order for a solution to exist, it is necessary that the three scalar equations represented by (11) be satisfied. Since $n_3 = 0$, the equation represented by the third row of (11) may be written as

$$f_1 n_1 + f_2 n_2 = 0. \tag{12}$$

Eliminating the unknown f_3, from the two scalar equations obtained from the first two rows of (11) gives

$$l_3(f_1 m_1 + f_2 m_2) - m_3(f_1 l_1 + f_2 l_2) = 0. \tag{13}$$

Equations (12) and (13) represent the conditions that must be satisfied for the mechanism to be in static equilibrium. All the terms in these equations have been defined in terms of the descriptive parameters γ_1 and γ_2.

4 Case 1 – Both Free Lengths Equal Zero

For this simple case it is assumed that the free lengths of the two springs, i.e. L_{01} and L_{02}, are both equal to zero. The forces in the two springs as defined in (7) and (8) now reduce to

$$f_1 = k_1 d_1, \tag{14}$$

$$f_2 = k_2 d_2. \tag{15}$$

Substituting these expressions as well as the line coordinate terms defined by (3) into (12) and (13) give

$$\begin{aligned} &L_3(k_1 L_{12} + k_2 p_{3x}) s_1 + [k_1 L_{12} L_{45} + k_2(p_{3x} p_{6x} + p_{3y} p_{6y})] s_2 \\ &\quad - k_2 L_3 p_{3y} c_1 + k_2(p_{3x} p_{6y} - p_{3y} p_{6x}) c_2 = 0, \end{aligned} \tag{16}$$

$$(k_1 L_{45} + k_2 p_{6x})(c_1 s_2 - s_1 c_2) + k_2 p_{6y}(c_1 c_2 + s_1 s_2) + (k_2 p_{3x} + k_1 L_{12}) s_1 - k_2 p_{3y} c_1 = 0. \quad (17)$$

Note that since the free lengths of the springs are zero that the terms d_1 and d_2 have vanished.

The solution for the values of the angles γ_1 and γ_2 that simultaneously satisfy (16) and (17) proceeds by defining their tan-half angles as

$$x_i = \tan \frac{\gamma_i}{2} \quad (18)$$

and then introducing the trigonometric identities

$$s_i = \frac{2x_i}{1 + x_i^2}, \quad c_i = \frac{1 - x_i^2}{1 + x_i^2}. \quad (19)$$

Substituting (19) into (16) and (17) and rearranging yields

$$(A_1 x_2^2 + A_2 x_2 + A_3) x_1^2 + (A_4 x_2^2 + A_5 x_2 + A_6) x_1 + (A_7 x_2^2 + A_8 x_2 + A_9) = 0, \quad (20)$$

$$(B_1 x_2^2 + B_2 x_2 + B_3) x_1^2 + (B_4 x_2^2 + B_5 x_2 + B_6) x_1 + (B_7 x_2^2 + B_8 x_2 + B_9) = 0, \quad (21)$$

where the coefficients A_1 through B_9 can be evaluated in terms of given values.

Crane and Duffy (1998) show how Bezout's method can be used to yield in general eight solutions for x_1 and x_2 that satisfy the bi-quadratic equations (20) and (21). In this case the solution was found symbolically to reduce to sixth degree.

Several numerical examples were evaluated. Typically four real solutions and two complex solutions were obtained. All six cases satisfied the equilibrium conditions defined by (12) and (13).

5 Case 2 – One Non-Zero Free Length

For this case it is assumed that the free length of spring 1 is non-zero and the free length of spring 2 is zero. The forces in the two springs as defined in (7) and (8) are now written as

$$f_1 = k_1(d_1 - L_{01}), \quad (22)$$

$$f_2 = k_2 d_2. \quad (23)$$

Substituting (22) and (23) as well as the line coordinate terms defined by (3) into (12) and (13) and rearranging now gives

$$A_1 d_1 + A_2 = 0, \quad (24)$$

$$B_1 d_1 + B_2 = 0, \quad (25)$$

where the terms A_1 through B_2 are expressed in terms of the sines and cosines of γ_1 and γ_2. Equations (24) and (25) express the necessary and sufficient condition for an equilibrium configuration. Note that the square of the distance between points 2 and 5, i.e. d_1^2, is expressed in terms of the angles γ_1 and γ_2 in Equation (5).

Equations (24), (25), and (5) are treated simply as a set of three equations in the three unknowns γ_1, γ_2, and d_1, and no attempt is made to manipulate the equations so that one variable is eliminated by direct substitution. Substituting the tan-half-angle identities, (19), into these three equations and rearranging yields

$$(E_1x_2^2 + E_2x_2 + E_3)d_1 + E_4x_2^2 + E_5x_2 + E_6 = 0, \tag{26}$$

$$(F_1x_2^2 + F_2x_2 + F_3)d_1 + F_4x_2^2 + F_5x_2 + F_6 = 0, \tag{27}$$

$$(G_1x_2^2 + G_2x_2 + G_3)d_1^2 + G_4x_2^2 + G_5x_2 + G_6 = 0, \tag{28}$$

where the coefficients E_1 through G_6 are functions of x_1.

Sylvester's elimination procedure is used to solve the set of equations (26) through (28). These three equations are multiplied by x_2, d_1, and d_1x_2. Equations (26) and (27) are multiplied by d_1^2 and $d_1^2x_2$. This results in a total of 16 equations in the set of 'variables' $d_1^3x_2^3$, $d_1^3x_2^2$, $d_1^3x_2$, d_1^3, $d_1^2x_2^3$, $d_1^2x_2^2$, $d_1^2x_2$, d_1^2, $d_1x_2^3$, $d_1x_2^2$, d_1x_2, d_1, x_2^3, x_2^2, x_2, and 1. A necessary condition for a solution to exist for these 16 'linear' 'homogeneous' equations is that they be linear dependent and thus the determinant of the coefficient matrix must equal zero. Since the coefficients E_1 through G_6 are 2nd order in the variable x_1, expansion of the determinant results in a 32nd degree polynomial in the variable x_1. Values for x_2 and d_2 that correspond to each solution of x_1 are then readily obtained.

Several numerical examples for this case were performed. Eight of the 32 solutions for x_1 were equal to $+i$ or $-i$ which means that the 32nd degree polynomial in x_1 may be divided by the factor $(1 + x_1^2)^4$. Four of the solutions correspond to the case where points 2 and 5 are coincident. It can be shown that the value of x_1 when the two points are coincident may be determined from

$$(L_3^2 + L_{12}^2 - L_{45}^2 + 2L_3L_{12})x_1^2 + (L_3^2 + L_{12}^2 - L_{45}^2 - 2L_3L_{12}) = 0. \tag{29}$$

Thus, the remaining 24th degree polynomial may be divided by this factor to result in a 20th degree polynomial in x_1. It is concluded that the correct degree of the solution is 20.

6 Case 3 – Both Free Lengths Are Non-Zero

For this case, the free lengths of both springs are non-zero and the forces in the two springs are defined in (7) and (8). Substituting (7) and (8) as well as the line coordinate terms defined by (3) into (12) and (13), substituting the tan-half angle identities for the sines and cosines of γ_1 and γ_2, and rearranging now gives

$$(C_1x_2^2+C_2x_2+C_3)d_1d_2+(C_4x_2^2+C_5x_2+C_6)d_1+(C_7x_2^2+C_8x_2+C-9)d_2=0, \tag{30}$$

$$(D_1x_2^2+D_2x_2+D_3)d_1d_2+(D_4x_2^2+D_5x_2+D_6)d_1+(D_7x_2^2+D_8x_2+D_9)d_2=0, \tag{31}$$

where the coefficients C_1 through D_9 are functions of x_1. Substituting the tan-half angle identities into (5) and (6) and rearranging gives

$$(M_1x_2^2+M_2x_2+M_3)d_1^2+(M_4x_2^2+M_5x_2+M_6)=0, \tag{32}$$

$$(N_1x_2^2+N_2x_2+N_3)d_2^2+(N_4x_2^2+N_5x_2+N_6)=0, \tag{33}$$

where the coefficients M_1 through N_6 are functions of x_1. The definition of the coefficients C_1 through N_6 are not listed here for brevity.

Equations (30) and (31) are divided by d_1d_2, Equation (32) is divided by d_1^2, and Equation (33) is divided by d_2^2 to yield the four equations

$$(C_1x_2^2+C_2x_2+C_3)+(C_4x_2^2+C_5x_2+C_6)d_{2i}+(C_7x_2^2+C_8x_2+C_9)d_{1i}=0, \tag{34}$$

$$(D_1x_2^2+D_2x_2+D_3)+(D_4x_2^2+D_5x-2+D_6)d_{2i}+(D_7x_2^2+D_8x_2+D_9)d_{1i}=0, \tag{35}$$

$$(M_1x_2^2+M_2x_2+M_3)+(M_4x_2^2+M_5x_2+M_6)d_{1i}^2=0, \tag{36}$$

$$(N_1x_2^2+N_2x_2+N_3)+(N_4x_2^2+N_5x_2+N_6)d_{2i}^2=0, \tag{37}$$

where $d_{1i}=1/d_1$ and $d_{2i}=1/d_2$. Equations (34) through (37) are expressions that must be solved at an equilibrium configuration and are expressed in terms of the four unknowns x_1 (which is embedded in the coefficients), x_2, d_{1i}, and d_{2i}.

Sylvester's elimination procedure is used to obtain a single polynomial in x_1. Equations (34) through (37) can be treated as four 'homogenous' equations in the 15 unknowns $d_{2i}^2x_2^2$, $d_{2i}^2x_2$, d_{2i}^2, $d_{1i}^2x_2^2$, $d_{1i}^2x_2$, d_{1i}^2, $d_{2i}x_2^2$, $d_{2i}x_2$, d_{2i}, $d_{1i}x_2^2$, $d_{1i}x_2$, d_{1i}, x_2^2, x_2, and 1. Equations (34) and (35) are multiplied by d_{1i}, d_{2i}, $d_{1i}d_{2i}$, d_{1i}^2, d_{2i}^2, $d_{1i}^2d_{2i}$, and $d_{1i}d_{2i}^2$, Equation (36) is multiplied by d_{1i}, d_{2i}, $d_{1i}d_{2i}$, and d_{2i}^2, and Equation (37) is multiplied by d_{1i}, d_{2i}, $d_{1i}d_{2i}$, and d_{1i}^2 to yield a set of 26 equations in a total set of 39 unknowns. Multiplying all 26 of these equations by x_2 finally results in a total set of 52 'homogeneous' equations in 52 unknowns.

A non-trivial answer will exist for these equations if the determinant of the 52×52 coefficient matrix equals zero. Since the coefficients are functions of the variable x_1, a polynomial in x_1 will result. An analysis of the degree of the coefficients as functions of x_1 indicated that the resulting polynomial would be of degree 104.

Corresponding values for the parameters x_2, d_{1i}, and d_{2i} can be obtained by evaluating the coefficients C_1 through N_6 for a particular solution of x_1 and then solving any set of 51 of the 52 equations as true non-homogenous equations for the particular unknowns x_2, d_{1i}, and d_{2i}. In this problem this requires that a 51×51 matrix must be inverted for each of the 104 solutions for x_1.

A numerical example was run and 104 solution sets of x_1, x_2, d_1, and d_2 were obtained. Twenty-six of the x_1 solutions were equal to $\pm i$. These solutions are often

referred to as circular points at infinity and it must be the case that the 104th degree polynomial can be divided by $(1 + x_1^2)^{13}$. It was not surprising that these solutions occurred as the terms M_1, M_3, N_1, and N_3 all equaled $(1 + x_1^2)$.

Of the remaining 78 solutions, 40 were real and 38 were complex. Corresponding values for x_2, d_{1i}, and d_{2i} were obtained for the remaining 78 solutions. All 78 of the solutions were then substituted into (34) through (37) to see if they indeed satisfied the constraint equations. Sixteen of the real solutions did not satisfy (34) through (37) which means that extraneous roots were indeed introduced in the elimination process. It is concluded from this example that no more than 62 solutions exist. Further analysis must be conducted to obtain an elimination procedure that does not introduce extraneous roots.

7 Conclusions

The purpose of this paper was to show the significant increase in complexity that results when springs with non-zero free lengths are incorporated in pre-stressed mechanisms. It has been shown that six equilibrium configurations exist for the case of a simple planar mechanism with two springs where both springs have zero free lengths. Twenty equilibrium configurations were found for the case where one of the springs had a non-zero free length. For the case where both springs had non-zero free lengths, 78 solutions sets were obtained once the circular points at infinity were disregarded. Sixteen of these 78 did not satisfy the equation set which means that the presented elimination technique introduced extraneous roots. The remaining 62 solutions satisfied the equations, but two solutions in the numerical example resulted in cases where the lines along the three legs did not intersect which is puzzling.

Additional work needs to be done before this simple case is fully understood. The approach presented here does however bound the dimension of the solution. The goal of the authors is to extend this work to spatial devices in order to develop a thorough understanding of the nature of these pre-stressed mechanisms.

Acknowledgement

The authors would like to gratefully acknowledge the support of the Department of Energy, grant number DE-FG04-86NE37967.

References

Crane, C. and Duffy, J. (1998), *Kinematic Analysis of Robot Manipulators*, Cambridge Press.

Duffy, J., Rooney, J., Knight, B., and Crane, C. (2000), A review of a familly of self-deploying tensegrity structures with elastic ties, *The Shock and Vibration Digest* **32**(2), March, 100–106.

Edmondson, A. (1987), *A Fuller Explanation: The Synergetic Geometry of R. Buckminster Fuller*, Birkhauser, Boston.

Fuller, R. (1975), *Synergetics: The Geometry of Thinking*, MacMillan Publishing Co., New York.

Kenner, H. (1976), *Geodesic Math and How to Use It*, University of California Press, Berkeley and Los Angeles, CA.

Knight, B.F. (2000), *Deployable Antenna Kinematics using Tensegrity Structure Design*, Ph.D. Thesis, University of Florida, Gainesville, FL.

Stern, I.P. (1999), *Development of Design Equations for Self-Deployable N-Strut Tensegrity Systems*, M.S. Thesis, University of Florida, Gainesville, FL.

Tobie, R.S. (1976), *A Report on an Inquiry into the Existence, Formation and Representation of Tensile Structures*, Master of Industrial Design Thesis, Pratt Institute, New York.

Yin, J., Duffy, J., and Crane, C. (2002), An analysis for the design of self-deployable tensegrity and reinforced tensegrity prisms with elastic ties, *International Journal of Robotics and Automation* **17**(1), Special Issue on Compliance and Compliant Mechanisms.

Inverse Kinematics of Robot Manipulators with Multiple Moving Control Points

Agostino De Santis and Bruno Siciliano

PRISMA Lab, Dipartimento di Informatica e Sistemistica, Università degli Studi di Napoli Federico II, Via Claudio 21, 80125 Napoli, Italy, e-mail: {agodesa, siciliano}@unina.it

Abstract. The growing research area of physical Human–Robot Interaction (pHRI) claims for safe robot control algorithms in the presence of humans. Managing kinematic redundancy via fast techniques is also mandatory for interaction tasks with humans. It is worth noticing that control points on a manipulator can change, e.g., depending on possible multiple collisions (intentional or accidental) with the interacting users. An approach is presented for changing the control point in real time with corresponding proper inverse kinematics. Whole-body modelling is adopted for such a task. A simulation case study is proposed.

Key words: multiple-point control, inverse kinematics, skeleton algorithm, whole-body modelling.

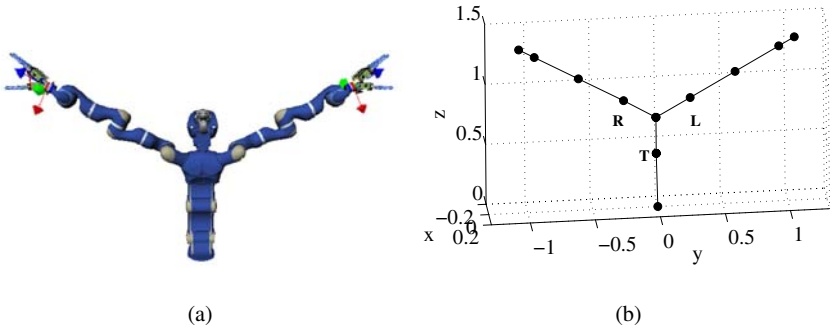
1 Introduction

A central idea in physical Human-Robot Interaction (pHRI) for anthropic domains [1] is the possibility of safely controlling the motion of an arbitrary part of the articulated structure of a robot, via direct touch or remote operation. Eventually, every point on such a robot can collide with a human user, resulting in (even severe) damages. In addition, the different postures that robots assume during their motion can scare the users, leading to sudden movements and forcing the robot control architecture to react properly to such unexpected behaviours.

Multiple control points have then to be considered. The approach presented in [2] takes into account the cited issues via multiple fixed control points. The inherent limitation of this solution is the fact that the control points have to be chosen manually before the experiments. This drawback has been solved in [4], according to the following criteria: geometric environment modelling for analytic computation, multiple-point approach both for multiple inputs and multiple outputs of the robot, arbitrary selection of the control points on the robot, reactive real-time control for safety.

The resulting whole-body modelling is completed in this paper, where the following additional issues are suggested and developed in detail: inverse kinematics,

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 429–438.



(a) (b)

Fig. 1 For the DLR Justin manipulator (a), a skeleton can be found (b) by considering the axes of the arms and the spine of the torso. Segments are drawn between the Cartesian positions of some crucial joints.

management of possible discontinuities, fast Jacobian computation via symbolic description and its consequences.

2 Whole-Body Modelling with the Skeleton Algorithm

As briefly introduced, an approach which automatically selects a control point on a whole kinematic structure, based on sensor information and analytical computation, is useful for pHRI applications. In addition, the control points should be computed fast, based on a model of the environment which leads to simple distance computation and trajectory determination.

These considerations lead to the so-called "skeleton algorithm" developed for collision avoidance applications (see [4] and references therein) whose steps include: building a proper model of the robot, namely the skeleton, useful for analytical computation; finding the control points along the skeleton, via distance computation or explicit user's decision; generating trajectories and corresponding joint commands for the controller.

2.1 Skeleton-Based Modelling

The problem of analyzing the whole volume of the parts of a manipulator is simplified by considering a *skeleton* of the structure, and proper volumes surrounding it. With reference to the DLR Justin manipulator [3], such a skeleton is reported in Figure 1.

If segments are built that "span" the kinematic structure of a manipulator (see Figure 1), its whole shape can be modelled. The underlying idea is that a solid of revolution can express the shape of a link, but this can be partially modified using properly the distance functions from every point of the link, resulting in a bounding volume around the point. These multiple volumes form a virtual region which has to approach the real volume of the considered part of a manipulator. Automatic skeleton building from Denavit–Hartenberg (DH) parameters is possible. In fact, a standard DH table gives the possibility of computing the *Cartesian position of each joint*. These positions can constitute the ends of the segments of the skeleton (*nodes*), and some of these possible nodes will be discarded if they coincide with other nodes already present in the skeleton. It is important to consider segments which cover the *spine* of all the mechanical parts: this has a reflex on DH tables when manipulator links have parts on both sides of a revolute joint as, e.g., for counterbalances or for allocating motors.

Building the skeleton, the focus is then on distance evaluation from the segments on the robot (bounding volumes) to the environment: this is the basis for motion control in an unstructured domain. The complete environment has to be modelled with geometric figures. In the design of the skeleton, some heuristics can help in discarding useless computation; nevertheless, the general case of computing all possible distances between simple objects like segments, regions of a plane (circles, rectangles) or points has the only limit of the time complexity for modelling the whole operation environment. The distances between these simple objects can be obtained via analytical formulas [5].

3 Inverse Kinematics with Multiple Control Points

Every point on the structure can be considered a control point, identified by a set of DH parameters.

The proposed symbolic approach is based on the consideration that, if one considers a manipulator and its direct kinematics equation, changing the value of its DH parameters results in the kinematics equations of another manipulator, whose end-effector is located before the real end-effector: that is equivalent to moving the control point of the structure.

In detail, since control points always lie on the spine of the robot links, the direct kinematics and the Jacobian computation can be carried out in a parametric way for a generic point $\boldsymbol{p}_i$, which is located after the i-th joint. Considering the homogeneous transformation relating the $(i-1)$-th frame (corresponding to the i-th joint) to the next frame, by simply replacing the DH parameter corresponding to the link length with the distance to the considered control point, a new "shorter" manipulator is considered for control. The values of direct kinematics and Jacobian for the specified control point have then to be considered by setting to 0 the DH values corresponding to frames located below the control point in the kinematic chain. If the

displacements on the links vary continuously and sequentially, from the tip of the robot towards the base and vice versa, the whole skeleton can be spanned.

In order to generate proper reference motion for the control points, in general, potential fields or different techniques can be used in order to generate the forces or velocities which will produce the desired motions. In [4] it is discussed how, e.g., repulsion forces can be derived from a potential function. These forces can be naturally used to compute avoidance torques at the manipulator joints via the Jacobian transpose. Nevertheless, it should be pointed out that suitable repulsion velocities could be likewise generated in lieu of forces.

The interesting implementation in velocity control will be presented, since the differential kinematics equation has an important modification, due to the fact that not only the joint angles, but also other kinematic parameters in the DH table may change during the task. The additional suggested tools for a velocity-level implementation are the choice of a modular Jacobian, and the proper management of a moving control point.

3.1 Modular Jacobian

The need for a modular Jacobian comes from the number of matrix multiplications which are necessary for an arbitrary number of degrees of freedom (DOFs). For the purpose of control, the Jacobian matrix is the cornerstone: similarly to the previous discussion, a symbolic Jacobian can be used, where the kinematic parameters in the DH table change as described above, allowing the motion of the control point. The dimensions of such a matrix change, depending on the available DOFs before the control point. The crucial aspect is that derivation of the differential kinematics equation is affected by the motion of the multiple control points. This can be also taken into account in a symbolic expression, as discussed below.

3.2 Differential Kinematics with Moving Points

Considering the direct and differential mappings with the standard DH parameters, the usual differential kinematics equation does not take into account the possibility of varying those kinematic parameters other than joint values. A complete model is as follows. The direct kinematics equation can be written in the form

$$\boldsymbol{p}_i = \boldsymbol{k}(\boldsymbol{q}_i) \tag{1}$$

where the vector $\boldsymbol{q}_i$ contains the vectors of the standard DH variables $\boldsymbol{d}_i$, $\boldsymbol{a}_i$, $\boldsymbol{\theta}_i$, $\boldsymbol{\alpha}_i$. The differential mapping, discarding possible variations of the values in $\boldsymbol{\alpha}_i$, is therefore

$$\dot{\boldsymbol{p}}_i = \frac{\mathrm{d}\boldsymbol{k}}{\mathrm{d}t} = \frac{\partial \boldsymbol{k}}{\partial \boldsymbol{\theta}_i}\frac{\partial \boldsymbol{\theta}_i}{\partial t} + \frac{\partial \boldsymbol{k}}{\partial \boldsymbol{a}_i}\frac{\partial \boldsymbol{a}_i}{\partial t} + \frac{\partial \boldsymbol{k}}{\partial \boldsymbol{d}_i}\frac{\partial \boldsymbol{d}_i}{\partial t}$$
$$= \boldsymbol{J}_{\theta,i}(\boldsymbol{\theta}_i, \boldsymbol{a}_i, \boldsymbol{d}_i)\dot{\boldsymbol{\theta}}_i + \boldsymbol{J}_{a,i}(\boldsymbol{\theta}_i, \boldsymbol{a}_i, \boldsymbol{d}_i)\dot{\boldsymbol{a}}_i + \boldsymbol{J}_{d,i}(\boldsymbol{\theta}_i, \boldsymbol{a}_i, \boldsymbol{d}_i)\dot{\boldsymbol{d}}_i. \quad (2)$$

Given a control point $\boldsymbol{p}_i$, the matrices $\boldsymbol{J}_{a,i}$ and $\boldsymbol{J}_{d,i}$ in (2) are the Jacobians which express the contribution to the motion of the control point of the variations of the DH values which characterise the control point. Moreover, $\boldsymbol{\theta}_i$ expresses the vector of the joint values which contribute to the motion of the control point. Notice that $\boldsymbol{J}_{\theta,i}$ is the ordinary Jacobian up to the control point, for a given set of DH parameters.

There are proper ways [6] for reducing the number of nonnull values in the $\boldsymbol{d}_i$ and $\boldsymbol{a}_i$ vectors of DH parameters. Often, this is intrinsically forced by the manipulator's design. As a simple case, consider a manipulator kinematics where only some values in the vector $\boldsymbol{d}_i$ change. In this situation, the way to compute the joint variables for a moving point on the skeleton of the robot is the following:

$$\dot{\boldsymbol{\theta}}_i = \boldsymbol{J}^{\dagger}_{\theta,W,\mathbf{i}}(\dot{\boldsymbol{p}}_i - \boldsymbol{J}_{d,i}(\boldsymbol{\theta}_i, \boldsymbol{a}_i, \boldsymbol{d}_i)\dot{\boldsymbol{d}}_i) \quad (3)$$

where the subscript W for the pseudoinverse of the Moore–Penrose Jacobian matrix $\boldsymbol{J}^{\dagger}_{\theta,W,\mathbf{i}}$ (corresponding to the control point $\boldsymbol{p}_i$) is referred to possible joint involvement weighing. Based on these simple modifications, multiple-point control, which has shown to be central in interaction with robots, can be accomplished easily both in force and velocity control.

The main issue is that the control points, with the associated Jacobian, move on the robot; therefore, this motion is taken into account in the differential kinematics.

3.3 Continuity of Moving Control Points

When a control point is computed automatically, e.g., via distance evaluation from the closest obstacle (or goal) to the skeleton of the manipulator, there is the possibility that its position changes in a discontinuous fashion. Consider as an example the case of multiple obstacles approaching an articulated robot. Moreover, some heuristics or the need for a reduced number of control points can result in some sudden change of control point, and therefore of the corresponding DH values.

In order to avoid this problem, the DH values for the control point have to be forced to vary with continuity and *in the right sequence*: this corresponds to moving to the next control point *always lying on the skeleton*. This can achieved, e.g., via spline interpolation resulting in moving the current control point towards next node of the skeleton, and then from there to the new control point, via a sequence of Cartesian positions of the joints, i.e., the nodes of the skeleton [5]. The sequence of variation of the DH parameters is important for simulating such a motion on the spine of the links, and the use of spline interpolation is suggested for specifying values of the higher-order derivatives of DH values.

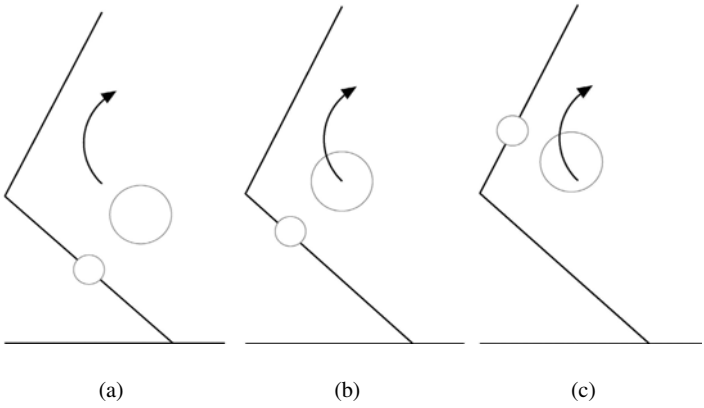


Fig. 2 The control point (small circle) on the skeleton, automatically computed as the closest to a moving object (big circle), can give a discontinuity to reference trajectories, moving abruptly on a new segment.

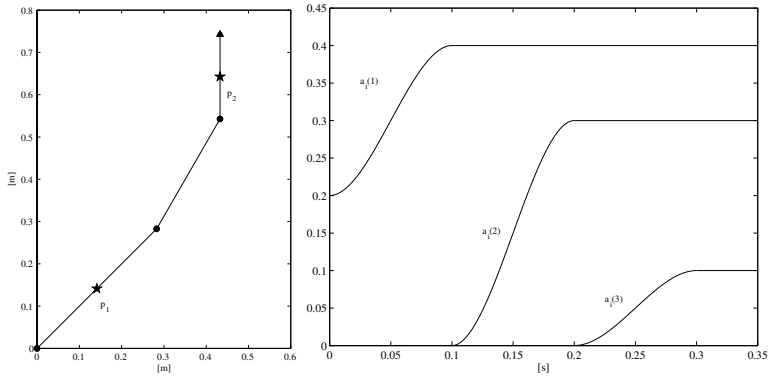
In order to reach the new control point via interpolation, a delay is to be considered before the change of the control point is performed. This delay has to be compatible with parameters related to the current motion of the robot such, e.g., the time-to-collision.

The possibility of smoothly moving the control point is useful also for forcing its motion on the skeleton in case of distance computation between parallel segments, where the computed closest point can move abruptly from an end to the other of the segment of the skeleton, in case of motion of an obstacle segment passing through a configuration which results in a parallelism with respect to the segment on the manipulator's skeleton.

Both first- and second-order inverse kinematics schemes [7] can be easily modified for taking into account the presence of the moving control point with variable kinematic parameters. In the case of second-order algorithms, the derivatives of the additional Jacobian matrices which have been introduced have to be computed also.

With reference to Figure 2, it can be seen that a single control point which is automatically computed as the closest to a collision, based on environment modelling, can move abruptly on the articulated structure. The use of different filters for the motion on the control point depends on the control algorithm too: if the control uses higher order derivatives of the position, motion has to be smooth enough to ensure continuity of these data.

Figure 3 shows an example of variation of the DH parameters which identify the control point for a three-link planar manipulator, whose lengths are 0.4 m, 0.3 m, 0.2 m. In such a manipulator, only the values in the vector $\boldsymbol{a}_i$, i.e., $a_i(1)$, $a_i(2)$, $a_i(3)$, change for a control point $\boldsymbol{p}_i$ on the skeleton. If the control point moves from the middle of the first link ($\boldsymbol{p}_1$) to the middle of the third link ($\boldsymbol{p}_2$), the resulting continuous and sequential change in the DH parameters, obtained via cubic spline



(a) (b)

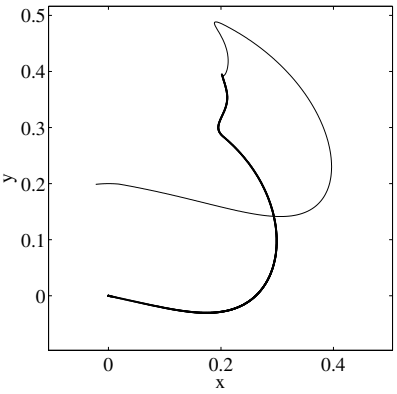
Fig. 3 When a control point's position is expected to change, e.g. from p_1 to p_2 (a), the desired new DH parameters are to be reached sequentially and with continuity (b).

interpolation, is reported. The time for reaching the desired final value for each parameter has been set always equal to 0.1 s. This time interval is a parameter for the spline interpolator. In this case, the change of three parameters results in reaching control point after 0.3 s. Meanwhile, the control point is moving on the skeleton and its motion is taken into account as described above.

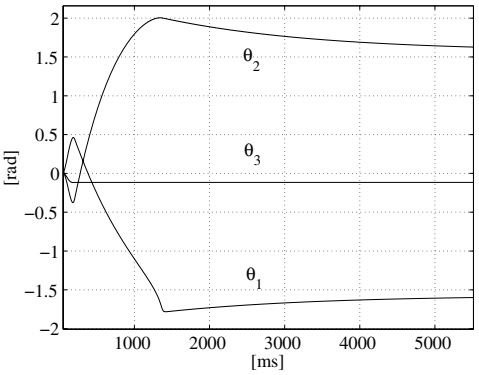
4 A Simulation Case Study

A short example is here reported for simplicity. Consider the case of a 3-link planar manipulator approaching an object located in the position $\boldsymbol{p}_g$. If one just wants that the robot hits the object, the closest point on the object from $\boldsymbol{p}_g$, i.e., the point $\boldsymbol{p}_c$, has to be attracted. Inverse kinematics can be performed as introduced for a single moving control point, which has to be driven towards $\boldsymbol{p}_g$. A velocity is commanded which is proportional to the difference between $\boldsymbol{p}_g$ and $\boldsymbol{p}_c$. When the manipulator moves, if $\boldsymbol{p}_c$ is automatically computed as the closest to the goal point, its desired location may change abruptly from one link to the other (see Figure 3). For the simulations, the length of each link is 0.2 m, the goal position is $\boldsymbol{p}_g = \begin{bmatrix} 0 & -0.2 \end{bmatrix}^T$, and the motion of the control point is limited to the last two links.

In order to smoothen the change of control point's position, a spline interpolation for the DH values of $\boldsymbol{p}_c$ is used. This results in a sliding motion of the control point $\boldsymbol{p}_c$ on the manipulator up to the new desired one. The resulting Cartesian motion and joint motions corresponding to the moving $\boldsymbol{p}_c$ is reported in Figure 4. Notice that a damped-least-squares solution has been adopted in the pseudoinversion of the



(a)



(b)

Fig. 4 (a) Time history of the Cartesian motion of control point (thick line) and end-effector (thin-line): notice that the control point moves soon to the new desired value. (b) Joint motion in the proposed case study.

Jacobian matrix; notice also that, in the final part of the trajectory, the desired direction for the control point is unfeasible (singular direction). These two last remarks are reported for suggesting the need for more global approaches including proper trajectory planning, in order to avoid local minima during the motion. As a simple example, the desired velocity for reaching the goal point $\boldsymbol{p}_g$ can be interpolated for avoiding high speeds in the first part of the motion.

From Figure 5 it is possible to observe the progressive modification of the values in the $\boldsymbol{a}_c$ vector (according to the notation introduced in Section 3.2), corresponding

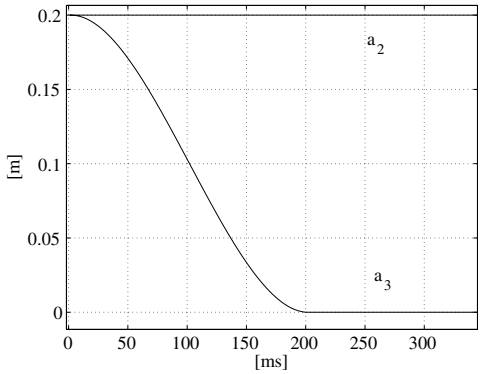


Fig. 5 Time history of the values in the vector $\boldsymbol{a}$.

to the motion of $\boldsymbol{p}_c$, which is initially located at the end-effector (default) and then moves to the second link.

Notice that the desired positions of $\boldsymbol{p}_c$ during the simulations are reached after fixed time slots, specified for the spline interpolators. During this time, no additional new desired points are computed. If the control point does not change any longer, the scheme coincides with the well-known closed-loop inverse kinematics (CLIK) with no feedforward of velocity.

5 Conclusion

An approach to inverse kinematics for possibly moving control points on the kinematic chain of a robot manipulator has been introduced in this paper. It is based on whole-body modelling: analytical computation or explicit choice may be used for setting the control points. Such control points are forced to move smoothly on the spine of the links of the considered robot manipulator. Simulation results have been provided for a simple planar robot manipulator; future work will be aimed at testing the approach on spatial manipulators, such as the DLR Justin manipulator.

Acknowledgements

This work is partly a development of a research started when the first author was a visiting researcher at DLR, supervised by Dr. Alin Albu-Schäffer. This work was supported by the PHRIENDS Specific Targeted Research Project, funded under the 6th Framework Programme of the European Community under Contract IST–045359. 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. De Santis, A., Siciliano, B., De Luca, A. and Bicchi, A., An atlas of physical Human–Robot Interaction, *Mechanism and Machine Theory* **43**, 253–270, 2008.
2. De Santis, A., Pierro, P. and Siciliano, B., The virtual end-effectors approach for human-robot interaction, in *Advances in Robot Kinematics: Mechanisms and Motion*, J. Lenarčič, B. Roth (Eds.), Springer, pp. 133–144, 2006.
3. Ott, C., Eiberger, O., Friedl, W., Bäuml, B., Hillenbrand, U., Borst, C., Albu-Schäffer, A., Brunner, B., Hirschmüller, H., Kiehlöfer, S., Konietschke, R., Suppa, M., Wimböck, T., Zacharias, F. and Hirzinger, G., A humanoid two-arm system for dexterous manipulation, in *Proceedings 2006 IEEE International Conference on Humanoid Robots*, Genova, I, IEEE, pp. 276–283, 2006.
4. De Santis, A., Albu-Schäffer, A., Ott, C., Siciliano, B. and Hirzinger, G., The skeleton algorithm for self-collision avoidance of a humanoid manipulators, in *Proceedings 2007 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Zürich, Switzerland, IEEE, 2007.
5. De Santis, A., *Modelling and Control for Human–Robot Interaction*, Research Doctorate Thesis, Università degli Studi di Napoli Federico II, Italy, 2007.
6. Sciavicco, L. and Siciliano, B., *Modelling and Control of Robot Manipulators*, 2nd edition, Springer-Verlag, London, UK, 2000.
7. Caccavale, F., Chiaverini, S. and Siciliano, B., Second-order kinematic control of robot manipulators with Jacobian damped least-square inverse: Theory and experiments, *IEEE/ASME Transactions on Mechatronics* **2**, 188–194, 1997.

On the Delassus Parallelogram

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

[1]*Department of Tool & Die-Making Engineering, National Kaohsiung University of Applied Sciences, 415 Chien Kung Road, Kaohsiung 80782, Taiwan R.O.C., e-mail: cclee@cc.kuas.edu.tw*
[2]*Ecole Centrale Paris, Grande Voie des Vignes, F-92295 Chatenay-Malabry, France, e-mail: jherve@ecp.fr*

Abstract. A general type of articulated parallelogram including four H joints with parallel axes and related pitches was discovered by Delassus in 1922. Therefore, it is called Delassus parallelogram. The planar or pseudo-planar parallelograms can be considered to be special cases. The possibly singular posture of a sub-chain HHH in a HHHH Delassus parallelogram is also introduced through a special case of degeneracy of Ball's cylindroid. A practical consequence is that the singularity of flattened parallelogram can be avoided under a scalar condition on the screw pitches of a Delassus parallelogram. Thus, the workspace of manipulators implementing parallelograms can be enlarged.

Key words: parallelogram, hinged parallelogram, pseudo-planar motion, Delassus' linkages, singular posture, twist, cylindroids.

1 Introduction

The hinged parallelogram (Hervé and Sparacino, 1991; Wohlhart, 1992) is a simple means to produce translational motion without implementing the prismatic P pair. The simplest parallelogram is a planar linkage with four bars jointed by four revolute R pairs. Opposite bars have equal lengths. This parallelogram was used in several manipulators (Clavel, 1987; Hervé, 1991; Wenger and Chablat, 2000; Gao et al., 2002; Angeles, 2004), especially in the Delta robot. Actually, helical H pairs (or screw joints) can also achieve this function. The relative translational 1-dof motion between two opposite bars will be introduced for the pseudo-planar case. Moreover, the paradoxical chain of Delassus parallelogram is further studied. The possibly singular posture of a sub-chain HHH in the Delassus parallelogram is got rid of through a special case of degeneracy of Ball's cylindroid (Ball, 1900).

After recalling some properties of the pseudo-planar parallelograms, we extend them to one of the Delassus paradoxical linkages, which is an articulated parallelogram with four H pairs having related pitches. It may be worth mentioning that all the Delassus linkages were recalled by Waldron (1968). The article will emphasize the Delassus parallelogram linkage and the avoidance of its singular posture by using Ball's cylindroid method and intrinsic coordinate-free approach.

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 439–449.

2 The Pseudo-Planar Parallelogram

The pseudo-planar four-bar linkages are characterized by the four screw H pairs with parallel axes and equal pitches. When bar lengths are those of a parallelogram, the relative motion between the opposite bars is a one-degree-of-freedom (1-dof) translational motion.

2.1 Chain of Two Hs with Parallel Axes and the Same Pitch

Referring to Figure 1, the three vectors $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ make up an orthonormal vector basis. Adding the datum of any origin O, $(O, \mathbf{i}, \mathbf{j}, \mathbf{k})$ is a Cartesian frame of reference of the Euclidean affine three-dimensional (3D) space of classical geometry. A serial concatenation of two helical (or screw) H pairs with axes parallel to $\mathbf{k}$ is considered. In this HH chain, a first H pair has the fixed axis $(A, \mathbf{k})$ and the pitch p. The real number $k = p/2\pi$ is called reduced pitch. In fact, $(A, \mathbf{k})$ is a frame of reference of the axis. Any point P along this axis is represented by $P = A + z\mathbf{k}$ where the real number z is called the point abscissa in the frame $(A, \mathbf{k})$. The vector $P - A = (\mathbf{AP}) = z\mathbf{k}$ characterizes the position of P in the frame $(A, \mathbf{k})$. In order to obtain simple expressions, A is chosen in the plane $Pl(O, \perp\mathbf{k})$, which passes through O and is perpendicular to $\mathbf{k}$. This screw pair generates the 1D group $\{H(A, \mathbf{k}, p)\}$, which is a Lie subgroup of the 6D Lie group of general displacements. The conventional notation $\{H(A, \mathbf{k}, p)\}$ is a short for $\{H(A, \mathbf{k}, p); \theta \mid \theta \in \mathbb{R}\}$, $\mathbb{R}$ denoting the set of real numbers. By a transformation $H(A, \mathbf{k}, p; \theta)$ of $\{H(A, \mathbf{k}, p)\}$, any point M_1 belonging to the moving body of the pair is transformed into M_1'

$$M_1 \rightarrow M_1' = A + \theta k\mathbf{k} + \exp(\theta\mathbf{k}\times)(\mathbf{AM}_1), \tag{1}$$

the parameter θ is an angle of rotation of the moving body from an arbitrarily chosen home configuration of the H pair. For simplifying calculation, we will choose $\theta = 0$ when the point B of the moving body is located at the fixed point $B = A + r\mathbf{i}$. The linear operator $\exp(\theta\mathbf{k}\times)$ acting on the vector $(\mathbf{AM}_1)$ is the exponential series of the skew-symmetric linear operator $\theta\mathbf{k}\times$ of the vector product by $\theta\mathbf{k}$,

$$\begin{aligned}\exp(\theta\mathbf{k}\times)(\mathbf{AM}_1) \;=\; & (\mathbf{AM}_1) + \theta\mathbf{k} \times (\mathbf{AM}_1) + (\theta^2/2!)(\mathbf{k}\times)^2(\mathbf{AM}_1) + \\ & + \cdots + (\theta^n/n!)(\mathbf{k}\times)^n(\mathbf{AM}_1) + \cdots .\end{aligned} \tag{2}$$

When the angle θ has only the infinitesimal value $\mathrm{d}\theta$, $\exp(\mathrm{d}\theta\mathbf{k}\times)(\mathbf{AM}_1)$ is equal to $(\mathbf{AM}_1) + \mathrm{d}\theta\mathbf{k} \times (\mathbf{AM}_1)$. Then, $(\mathbf{AM}_1') - (\mathbf{AM}_1)$ is an infinitesimal vector $\mathrm{d}\mathbf{M}_1 = \mathrm{d}\theta[k\mathbf{k} + \mathbf{k} \times (\mathbf{AM}_1)]$. The mapping $\mathbf{M}_1 \rightarrow \mathrm{d}\mathbf{M}_1 = \mathrm{d}\theta[k\mathbf{k} + \mathbf{k} \times (\mathbf{AM}_1)]$ is the twist of the foregoing screw motion. $\mathrm{d}\mathbf{M}_1/\mathrm{d}\theta = k\mathbf{k} + \mathbf{k} \times (\mathbf{AM}_1)$ is called rate of twist. The previous expression of the twist depends on a point A, which is located on the screw axis. Another expression of the same twist at the origin O is readily obtained

using $(\mathbf{AM}_1) = (\mathbf{AO}) + (\mathbf{OM}_1)$. Therefore, the general expression of the twist of a screw with an axis parallel to $\mathbf{k}$ has the form $\mathrm{d}\theta[\mathbf{t}_O + \mathbf{k} \times (\mathbf{OM}_1)]$ where $\mathbf{t}_O$ is a vector depending on the choice of O. Classically, the pitch of such a twist is proven to be the scalar product $\mathbf{k} \cdot \mathbf{t}_O$ and the foot N of the perpendicular drawn from O to the screw axis is given by $(\mathbf{ON}) = \mathbf{k} \times \mathbf{t}_O$.

The second H pair from the fixed base generates the group $\{H((B, \mathbf{k}, p)\}$. Due to the relative motion of this H pair, any point M_2 of the moving body is transformed into M_2'

$$M_2 \rightarrow M_2' = B + \varphi k\mathbf{k} + \exp(\varphi\mathbf{k}\times)(\mathbf{BM}_2). \tag{3}$$

The value 0 of the angle φ is chosen for an arbitrarily given home configuration of the second H pair.

In Figure 1, let M be any point belonging to the moving end body of the serial HH chain. Any change of the position of M can be achieved through two steps. In the first step, the second H moves with an angle φ while the first H keeps its home pose ($\theta = 0$); M becomes M_i. In the second step, the second H is locked (angle φ keeps its value) and the first H moves with an angle θ, M_i becomes M'. The first step of the change of point M position is expressed by

$$M \rightarrow M_i = B + \varphi k\mathbf{k} + \exp(\varphi\mathbf{k}\times)(\mathbf{BM}), \tag{4}$$

where B is at its home position $A + r\mathbf{i}$ and the transform $M_i - B = (\mathbf{BM}_i) = \varphi k\mathbf{k} + \exp(\varphi\mathbf{k}\times)(\mathbf{BM})$ is followed by the second transform

$$\begin{aligned} M_i \;\rightarrow\; & M' = A + \theta k\mathbf{k} + \exp(\theta\mathbf{k}\times)(\mathbf{AM}_i) \\ & \text{or } (\mathbf{AM}') = \theta k\mathbf{k} + \exp(\theta\mathbf{k}\times)(\mathbf{AM}_i). \end{aligned} \tag{5}$$

Using $(\mathbf{AM}_i) = (\mathbf{AB}) + (\mathbf{BM}_i)$ yields

$$(\mathbf{AM}') = \theta k\mathbf{k} + \exp(\theta\mathbf{k}\times)(\mathbf{AB}) + \exp(\theta\mathbf{k}\times)(\mathbf{BM}_i). \tag{6}$$

As the chosen home configuration is such as $(\mathbf{AB}) = r\mathbf{i}$,

$$\exp(\theta\mathbf{k}\times)(\mathbf{AB}) = r\exp(\theta\mathbf{k}\times)\mathbf{i} = r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j}). \tag{7}$$

Noticing that $\exp(\theta\mathbf{k}\times)(\varphi k\mathbf{k}) = \varphi k\mathbf{k}$,

$$\begin{aligned} (\mathbf{AM}') &= r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j}) + \theta k\mathbf{k} + \varphi k\mathbf{k} + \exp(\theta\mathbf{k}\times)\exp(\varphi\mathbf{k}\times)(\mathbf{BM}) \\ &= r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j}) + (\theta + \varphi)k\mathbf{k} + \exp[(\theta + \varphi)\mathbf{k}\times](\mathbf{BM}). \end{aligned} \tag{8}$$

This is an explicit expression of a transformation $H(A, \mathbf{k}, p; \theta)H(B, \mathbf{k}, p; \varphi)$ of the 2D product $\{H((A, \mathbf{k}, p)\}\{H(B, \mathbf{k}, p)\}$.

2.2 Parallel Mechanism Generating Circular Translation

Let us assume that M is any point belonging to a moving platform connected to a fixed base by two limbs. One limb is the previous HH chain. The Hs have parallel axes and the same pitch p. $p = 0$ is a possible special case, that is, the planar RR chain. The second limb is a PPP generator of 3-dof spatial translation. The set of feasible displacements generated by the PPP limb is the 3D group $\{T\}$ of spatial translations. The group $\{T\}$ is the set of point transformations

$$M \rightarrow M' = M + a\mathbf{i} + b\mathbf{j} + c\mathbf{k}, \tag{9}$$

which is also expressed by $(\mathbf{MM}') = a\mathbf{i} + b\mathbf{j} + c\mathbf{k}$. The parameters a, b and c are the scalar values of the rectilinear translations produced by the P pairs from a given home configuration of the PPP chain. For any point M, the feasible displacements of the moving platform for the HH-PPP parallel mechanism have to satisfy both

$$M \rightarrow M' = M + a\mathbf{i} + b\mathbf{j} + c\mathbf{k} \tag{10a}$$

and

$$M \rightarrow M' = A + r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j}) + (\theta + \varphi)k\mathbf{k} + \exp[(\theta + \varphi)\mathbf{k}\times)](\mathbf{BM}). \tag{10b}$$

The solution is obviously obtained with $(\theta + \varphi) = 0$, $\exp[(\theta + \varphi)\mathbf{k}\times)](\mathbf{BM}) = (\mathbf{BM}) = (\mathbf{BA}) + (\mathbf{AM}) = -r\mathbf{i} + (\mathbf{AM})$ and $A + (\mathbf{AM}) = M$. Therefore, Eq. (10b) becomes

$$M' = M + r(\cos\theta - 1)\mathbf{i} + r\sin\theta\mathbf{j}. \tag{11}$$

This point transformation is clearly a translation depending on only one parameter, namely θ. M being given, M' moves in a plane parallel to $Pl(O, \perp\mathbf{k})$. Let C_M be a point associated to M by $(\mathbf{C}_M\mathbf{M}) = r\mathbf{i}$.

$$\begin{aligned}(\mathbf{C}_M\mathbf{M}') &= (\mathbf{C}_M\mathbf{M}) + r(\cos\theta - 1)\mathbf{i} + r\sin\theta\mathbf{j} = r\mathbf{i} + r(\cos\theta - 1)\mathbf{i} + r\sin\theta\mathbf{j} \\ &= r\cos\theta\mathbf{i} + r\sin\theta\mathbf{j}. \end{aligned} \tag{12}$$

Clearly, any point M of the end effector moves on a circle of center C_M with the radius r. This circular translation can be produced by an infinity of mechanisms, which are congruent to the mechanism depicted in Figure 2 by any given translation.

In Figure 3, two mechanisms produce the same circular translation. That way, rigidly connecting two end effectors having the same motion leads to a pseudo-planar linkage that generates circular translation, as shown in Figure 4.

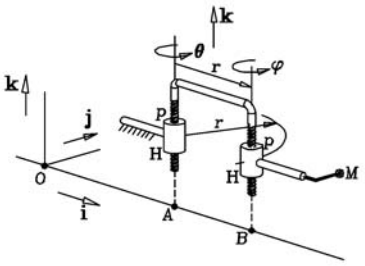


Fig. 1 Two Hs chain.

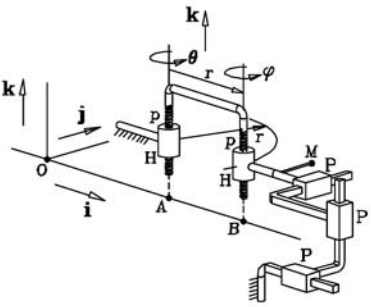


Fig. 2 A HH-PPP parallel chain.

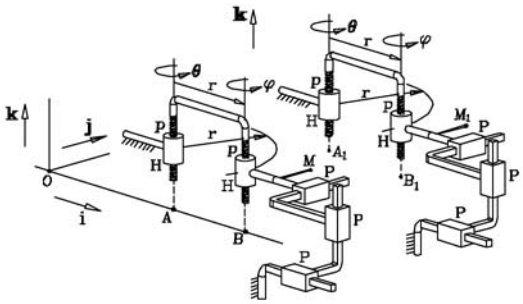


Fig. 3 Two mechanisms generating the same circular translation.

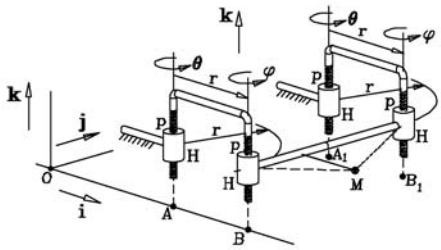


Fig. 4 A pseudo-planar parallelogram.

3 The Delassus Parallelogram

Assume that the two screw pitches shown in Figure 1 are not equal. For instance, the pitch of the first H pair is $p = 2\pi m$ and the second H has the pitch $q = 2\pi n$. Then, $p - q = 2\pi(m - n) \neq 0$. In a similar way, the explicit form of a transformation belonging to the product $\{H(A, \mathbf{k}, p)\}\{H(B, \mathbf{k}, q)\}$ for any point M, is derived as

$$(\mathbf{AM}') = r(\cos\theta\mathbf{i} + \sin\theta\mathbf{j}) + (m\theta + n\varphi)\mathbf{k} + \exp[(\theta + \varphi)\mathbf{k}\times)](\mathbf{BM}). \qquad (13)$$

Translation is achieved with $\theta + \varphi = 0$ or $\varphi = -\theta$

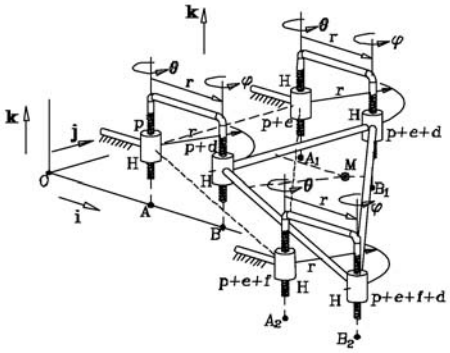


Fig. 5 A new mechanism.

$$\begin{aligned}(\mathbf{AM}') &= r(\cos\theta\mathbf{i}+\sin\theta\mathbf{j})+\theta(m-n)\mathbf{k}+\exp(0\mathbf{k}\times)(\mathbf{BM})\\ &= r(\cos\theta\mathbf{i}+\sin\theta\mathbf{j})+\theta(m-n)\mathbf{k}+(\mathbf{BM})\\ &= r(\cos\theta\mathbf{i}+\sin\theta\mathbf{j})+\theta(m-n)\mathbf{k}+(\mathbf{BA})+(\mathbf{AM})\\ &= r(\cos\theta\mathbf{i}+\sin\theta\mathbf{j})+\theta(m-n)\mathbf{k}-r\mathbf{i}+(\mathbf{AM}).\end{aligned} \tag{14}$$

Hence,

$$\begin{aligned}(\mathbf{AM}')-(\mathbf{AM}) &= (\mathbf{MM}')=r(\cos\theta\mathbf{i}+\sin\theta\mathbf{j})+\theta(m-n)\mathbf{k}-r\mathbf{i}\\ &= r(\cos\theta-1)\mathbf{i}+\sin\theta\mathbf{j}+\theta(m-n)\mathbf{k}.\end{aligned} \tag{15}$$

From any given point M of the end effector, a corresponding point C_M is associated by $C_M = M - r\mathbf{i}$. It is straightforward to establish that

$$(\mathbf{C}_M\mathbf{M}') = r\cos\mathbf{i}+r\sin\theta\mathbf{j}+\theta(m-n)\mathbf{k}. \tag{16}$$

So, M' moves on a helix of pitch $2\pi(m-n) = p-q = d$. Such a helix is drawn on a cylindrical surface having the axis $(C_M, \mathbf{k})$ with the radius r. The end-effector motion is a translation along a helix or a helical translation.

Several limbs producing the same helical translation can be placed in-parallel between a fixed base and a moving end effector. The case of two limbs corresponds to the parallelogram linkage found out by Delassus in 1922. Besides, parallel chains with more than two limbs are new. For instance, the mechanism depicted in Figure 5 is a new mechanism with two independent closed loops. Three HH limbs connect in parallel a fixed frame to a moving platform. The four real numbers p, d, e, and f can be chosen arbitrarily. The platform motion is helical translation with the pitch $p-(p+d) = p+e-(p+e+d) = p+e+f-(p+e+f+d) = -d$ and the radius r. In the noteworthy special case of $d = 0$, the motion of the coupler is a circular translation.

4 Singular Postures

The pseudo-planar parallelogram is singular when the four H axes are in a plane. In this posture, the four twists belong to a 2D vector space. Locally, the dof is 2. As it is well known in the case of a planar parallelogram, the pseudo-planar parallelogram can work with two 1-dof modes. The planar projection of the linkage can be either a parallelogram or a crossed parallelogram. A bifurcation (Lee and Hervé, 2005) happens at the above-mentioned singular posture.

Here, we focus on the case of the Delassus parallelogram, which is more intricate. The detection of the possible singular postures of a HHH sub-chain with three parallel axes can be done through the method used by Ball (1900) when he found out the cylindroid. The linear span of two given twists is characterized by geometric conditions (Lee and and Hervé, 2007). Thus, these conditions allow the identification of the twists, which belong to the 2D vector space of the two given twist. One has to notice that when the two given twists have parallel axes, Ball's cylindroid is degenerated and, therefore, a specific study of this special case is required.

Let us consider two H pairs with pitches $p = 2\pi m$ and $q = 2\pi n$ and parallel axes $(A, \mathbf{k})$ and $(B, \mathbf{k})$. Assume that both A and B belong to the plane $Pl(O, \perp\mathbf{k})$ and do not coincide. The expression of the twist of the first H with respect to O is $\mathbf{d}_1\mathbf{M} = \varphi[\mathbf{k} \times (\mathbf{OM}) + (\mathbf{OA}) \times \mathbf{k} + m\mathbf{k}]$ and the twist of the second one is $\mathbf{d}_2\mathbf{M} = \varphi[\mathbf{k} \times (\mathbf{OM}) + (\mathbf{OB}) \times \mathbf{k} + n\mathbf{k}]$; for brevity, θ and φ are shortened notations standing for $\mathrm{d}\theta$ and $\mathrm{d}\varphi$, which are infinitesimal angles. Then, the twist of a serial concatenation of the two H pairs, which is the resultant twist $\mathbf{d}_R\mathbf{M}$ is

$$\begin{aligned}
\mathbf{d}_R\mathbf{M} &= \mathbf{d}_1\mathbf{M} + \mathbf{d}_2\mathbf{M} \\
&= \theta[\mathbf{k} \times (\mathbf{OM}) + (\mathbf{OA}) \times \mathbf{k} + m\mathbf{k}] + \varphi[\mathbf{k} \times (\mathbf{OM}) + (\mathbf{OB}) \times \mathbf{k} + n\mathbf{k}] \\
&= (\theta + \varphi)\{\mathbf{k} \times (\mathbf{OM}) + \{[\theta(\mathbf{OA}) + \varphi(\mathbf{OB})]/(\theta + \varphi)\} \times \mathbf{k} \\
&\quad + [(\theta m + \varphi n)/(\theta + \varphi)]\mathbf{k}\}. \qquad (17)
\end{aligned}$$

This resultant twist has the general form $\mathbf{d}_R\mathbf{M} = (\theta + \varphi)[\mathbf{k} \times (\mathbf{OM}) + \mathbf{t}_{RO}]$ with $\mathbf{t}_{RO} = [\theta(\mathbf{OA}) + \varphi(\mathbf{OB})]/(\theta + \varphi) \times \mathbf{k} + [(\theta m + \varphi n)/(\theta + \varphi)]\mathbf{k}$. One can readily identify the twist of a screw pair with the angle $\alpha_R = \theta + \varphi$ and an axis parallel to $\mathbf{k}$. The reduced pitch is $k_R = \mathbf{k} \cdot \mathbf{t}_{RO}$ and

$$k_R = (\theta m + \varphi n)/(\theta + \varphi) \quad \text{or} \quad 2\pi k_R = (\theta p + \varphi q)/(\theta + \varphi). \qquad (18)$$

Setting $c_\theta = \theta/(\theta + \varphi)$ and $c_\varphi = \varphi/(\theta + \varphi)$, the pitch $2\pi k_R$ of the resultant twist is expressed as $c_\theta p + c_\varphi q$.

The vector of the perpendicular drawn from O to the axis of the resultant twist is given by $(\mathbf{ON}_R) = \mathbf{k} \times \mathbf{t}_{RO}$. By vector calculation using the formula of the double vector product, it is straightforward to establish

$$(\mathbf{ON}_R) = [\theta(\mathbf{OA}) + \varphi(\mathbf{OB})]/(\theta + \varphi) = c_\theta(\mathbf{OA}) + c_\varphi(\mathbf{OB}). \qquad (19)$$

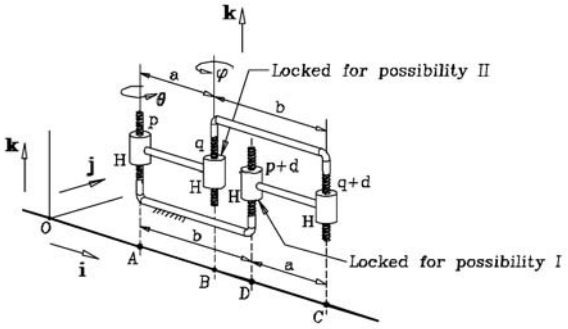


Fig. 6 Delassus parallelogram or sub-chain HHH in a possible singular pose.

From the previous expression, N_R is the barycenter (center of mass) of the point A and the point B with the mass $c_\theta = \theta/(\theta + \varphi)$ and the mass $c_\varphi = \varphi/(\theta + \varphi)$, respectively. In other words, it is a parametric representation of the straight line AB. That proves that necessarily N_R belongs to the line AB. Clearly, the resultant twist axis and the other two axes have to lie in the same plane. One can readily demonstrate

$$(\mathbf{N}_R\mathbf{A}) = (\mathbf{OA}) - (\mathbf{ON}_R) = \varphi(\mathbf{BA})/(\theta + \varphi) = c_\varphi(\mathbf{BA}). \tag{20}$$

By the same token,

$$(\mathbf{N}_R\mathbf{B}) = -\theta(\mathbf{BA})/(\theta + \varphi) = -c_\theta(\mathbf{BA}). \tag{21}$$

The expressions in Eqs. (20) and (21) show that the coefficients $c_\theta = \theta/(\theta + \varphi)$ and $c_\varphi = \varphi/(\theta + \varphi)$ can be expressed with ratios of parallel vectors that are derived of the arrangement on a straight line of the three points A, B and N_R, namely $c_\theta = \theta/(\theta + \varphi) = (\mathbf{BN}_R)/(\mathbf{BA})$ and $c_\varphi = \varphi/(\theta + \varphi) = (\mathbf{AN}_R)/(\mathbf{AB})$.

Now, we consider a Delassus parallelogram whose bar lengths are given and the pitches are p, q, $p + d$ and $q + d$ as shown in Figure 6. We consider a posture including three Hs with parallel axes lying in a plane. Clearly, when three axes are in a plane, the four axes are in the same plane.

Obviously, there are four possibilities (I, II, III and IV) for excerpting HHH sub-chains from a HHHH chain. Each sub-chain is obtained by locking any one of H pairs in the chain with four Hs. First, for possibility I, the screw at D is locked in Figure 6. Let us check if the HHH sub-chain pose of the screws at A, B and C is singular.

We derive the pitch of the singularity from the arrangement of A, B and C on a straight line, in which, $(\mathbf{AB}) = a\mathbf{i}$, $(\mathbf{BC}) = b\mathbf{i}$, and

$$c_\theta = (\mathbf{BC})/(\mathbf{BA}) = -b/a; \quad c_\varphi = (\mathbf{AC})/(\mathbf{AB}) = (a + b)/a. \tag{22}$$

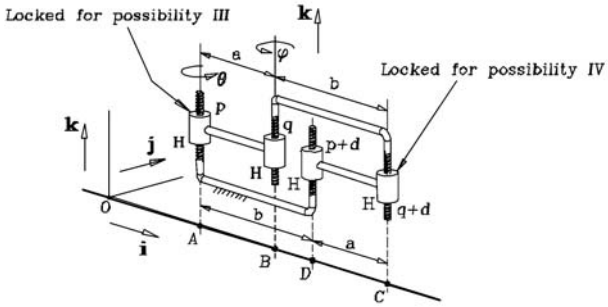


Fig. 7 A sub-chain HHH in a possible singular pose.

From the previous characterization of a singular posture of a HHH chain when two of its H pairs have parallel axes, the pose of the HHH sub-chain of the screws at A, B and C is singular iff the pitch of the screw at C is

$$c_\theta p + c_\varphi q = -bp/a + (a+b)q/a = b(q-p)/a + q, \tag{23}$$

which corresponds to $d = b(q-p)/a$. This singularity can be avoided by an adequate choice of the pitch $q+d$ of the screw at C, that is, $d \neq b(q-p)/a$.

Secondly, the screw at B is assumed to be locked for possibility II, Figure 6. Then we verify if the HHH sub-chain of the screws at A, D and C undergoes a singular pose. In the previous calculation, we just exchange a and b and replace q by $p+d$. When the singularity happens, the pitch of the screw at C is equal to $[ad+b(p+d)]/b$ based on Eq. (18). Then, we obtain the equation for the unknown d as follows:

$$q + d = [ad + b(p+d)]/b \Rightarrow d = b(q-p)/a. \tag{24}$$

Amazingly, we obtain the same condition as cited above, namely $d = b(q-p)/a$.

Next, we further only lock the screw at A for possibility III as shown in Figure 7. Is the HHH sub-chain of the screws at B, C and D in a singular pose? It is seen that

$$c_\theta = (\mathbf{CD})/(\mathbf{CB}) = a/b \quad \text{and} \quad c_\varphi = (\mathbf{BD})/(\mathbf{BC}) = (b-a)/b. \tag{25}$$

The pitch of the singularity is $qa/b + (q+d)(b-a)/b$ through Eq. (18). Hence, we obtain the following equation for the unknown d

$$qa/b + (q+d)(b-a)/b = p + d \Rightarrow d = b(q-p)/a. \tag{26}$$

Once more, we obtain the same magnitude of d.

Finally, for possibility IV, we investigate if the HHH sub-chain of the screws at B, A and D, shown in Figure 7 is at a singular pose. By the same way, we obtain again the same condition of singularity, which is $d = b(q-p)/a$.

An algebraic reasoning confirms and summarizes these findings. Let $\$_A$, $\$_B$, $\$_C$ and $\$_D$ denote the twists in the screws at A, B, C and D, respectively. In a first step, we have proven that the twist $\$_C$ belongs to the linear span of $\$_A$ and $\$_B$ iff $d = b(q - p)/a$. This is expressed by $\$_C \in \text{span}(\$_A, \$_B) \Leftrightarrow d = b(q - p)/a$. $\$_C \in \text{span}(\$_A, \$_B)$ implies that $\$_C$ can be a vector of a vector base of $\text{span}(\$_A, \$_B)$; therefore $\text{span}(\$_A, \$_B) = \text{span}(\$_A, \$_C)$. In a second step, we have proven $\$_C \in \text{span}(\$_A, \$_D) \Leftrightarrow d = b(q - p)/a$; then $\$_C \in \text{span}(\$_A, \$_D) \Rightarrow \text{span}(\$_A, \$_D) = \text{span}(\$_A, \$_C)$. Hence, iff $d = b(q - p)/a$, then the four twists of a Delassus parallelogram linkage in a straight-line posture belong to the same 2D vector space, which is the linear span of any two distinct twists among $\$_A$, $\$_B$, $\$_C$ and $\$_D$.

5 Conclusion

Besides the planar and pseudo-planar parallelograms, which can produce 1-dof translation along a circle, the Delassus parallelogram produces generally 1-dof translation along a helix. By an adequate choice of the screw pitches, circular translation can also be generated by a Delassus parallelogram. There is a special kind of Delassus parallelograms, which undergo a singular pose when the four bars are parallel to a straight line. The condition tying the bar lengths and the screw pitches in this kind of linkage is simple. As a consequence, the Delassus parallelograms can be used for avoiding the singularity of the flattened parallelogram and, thus, the workspace of manipulators implementing parallelograms can be enlarged. It is quite clear that several novel linkages including Delassus' parallelograms can be envisioned from this work.

References

Al-Widyan, K. and Angeles, J. (2004), The robust design of Schönflies-motion generators, in *On Advances in Robot Kinematics*, J. Lenarčič and C. Galletti (Eds.), Kluwer Academic Publishers, Dordrecht, pp. 339–350, 2004.

Angeles, J. (2004), The qualitative synthesis of parallel manipulators, *ASME J. Mech. Design* **126**, 617–674.

Ball, R.S. (1900), *The Theory of Screws*, Cambridge University Press (republished in 1998).

Clavel, R. (1987), Dispositif pour le déplacement et le positionnement d'un éléments dans l'espace, Swiss patent No. 672089A5, 1985, and international patent (PCT) no WO 87/03538.

Delassus, E. (1922), Les chains articulées fermées et déformables à quatre membres, *Bull. Sci. Math.* **46**, 283–304.

Gao, F., Li, W., Zhao, X., Jin, Z. and Zhao, H. (2002), New kinematic structures for 2-, 3-, 4-, and 5-DOF parallel manipulator designs, *Mech. Mach. Theory* **37**, 1395–1411.

Hervé, J.M. (1991), Dispositif pour le déplacement en translation spatiale d'un élément dans l'espace en particuller pour robot mécanique, French Patent No. 9100286.

Hervé, J.M. and Sparacino, F. (1991), Structural synthesis of parallel robots generating spatial translation, in *Proceedings 5th IEEE International Conference on Advanced Robotics*, Pisa, Italy, pp. 808–813.

Lee, C.-C. and Hervé, J.M. (2005), Discontinuously movable seven-link mechanisms via group-algebraic approach, *Proc. IMechE, Part C: J. Mechanical Engineering Science* **219**, 577–587.

Lee, C.-C. and Hervé, J. M. (2007), Cartesian parallel manipulators with pseudoplanar limbs, *ASME J. Mech. Design* **129**(12), 1256–1264.

Wenger, P. and Chablat, D. (2000), Kinematic analysis of a new parallel machine tool: The orthoglide, in *Advances in Robot Kinematics*, J. Lenarcic and M.M. Stanisic (Eds.), Kluwer Academic Publisher, Dordrecht, pp. 305–314.

Waldron, K.J. (1968), Hybrid overconstrained linkages, *J. Mechanisms*, **3**, 73–78.

Wohlhart, K. (1992), Displacement analysis of the general spatial parallelogram manipulator, in *Proceedings of the 3rd International Worksshop on Advances in Robot Kinematics*, September 7–9, Ferrara, Italy, pp. 104–111.

Forward Kinematics and Workspace Determination of a Wire Robot for Industrial Applications

Andreas Pott

Fraunhofer Institute for Manufacturing Engineering and Automation IPA, 70596 Stuttgart, Germany, e-mail: andreas.pott@ipa.fraunhofer.de

Abstract. This paper presents the recent results from a newly designed parallel wire robot which is currently under construction. Firstly, an overview of the system architecture is given and technically relevant requirements for the realization are identified. A technique to compute and transfer an estimation of the workspace to CAD tools is presented. Furthermore, tools to solve the forward kinematics of some special configuration under real-time requirements are explored. Simulation results show the feasibility of the presented algorithms.

Key words: wire robot, workspace, forward kinematics.

1 Introduction

Compared to other manipulators like industrial robots and Stewart–Gough platforms, parallel wire robots are able to achieve very high velocities and accelerations. Furthermore, large workspace and high payloads are possible due to the efficient force transmission through the wires. In the last decade, a lot of research has been carried out to study both, theory (see e.g. [3, 4, 8]) and implementation [1, 5] of these robots.

A new wire robot WiRo (Figure 1a) is currently being setup at the laboratories of Fraunhofer IPA. This new robot provides six degrees-of-freedom with seven wires and focuses on industrial applications in the field of material handling as well as fast pick-and-place applications. The aim of the demonstrator is to implement latest techniques in kinematics and control on industrial hardware capable of working in an automation environment. Among other things, two important issues arise in the construction of a wire robot which will be the topic of this paper. Firstly, one need to quickly calculate and visualize an estimate of the workspace during the construction process. Secondly, one has to provide all kinematic relationships required by an industrial robot controller. The calculation of the inverse kinematic function for set-value generation is trivial for parallel wire robots. On the other hand, forward kinematics is needed for some control algorithms, including workspace supervision and detection of failures. The calculation of the forward kinematics under real-time

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 451–458.

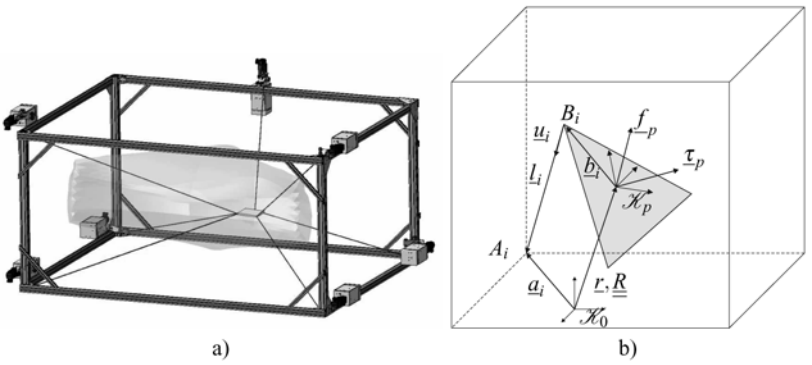


Fig. 1 A spatial wire robot. (a) CAD Draft of WiRo. (b) Geometry and kinematics.

requirements restricts the available computation time, and necessitate a strictly deterministic behavior for the computational time.

2 Kinematic Foundation

For better reference, the kinematic basics of wire robots are briefly reviewed. Figure 1b shows the kinematic structure of a spatial wire robot, where the vectors $\underline{a}_i$ denote the proximal attachment points on the frame, the vectors $\underline{b}_i$ are the relative positions of the distal attachment points on the movable platform, and l_i denote the length of the wires. Applying a vector loop, the closure-constraint reads

$$\underline{l}_i = \underline{a}_i - \underline{r} - \underline{\underline{R}}\underline{b}_i \quad \text{for} \quad i = 1, \ldots, m \tag{1}$$

where the vector $\underline{r}$ is the Cartesian position of the platform and the rotation matrix $\underline{\underline{R}}$ represents the orientation of the platform. The unit vector along the tension becomes $\underline{u}_i = \underline{l}_i |\underline{l}_i|^{-1}$. For force equilibrium it holds that [6, 8]

$$\underbrace{\begin{pmatrix} \underline{u}_1 & \cdots & \underline{u}_m \\ \underline{b}_1 \times \underline{u}_1 & \ldots & \underline{b}_m \times \underline{u}_m \end{pmatrix}}_{\underline{\underline{A}}(\underline{r}, \underline{\underline{R}})} \underbrace{\begin{pmatrix} f_1 \\ \vdots \\ f_m \end{pmatrix}}_{\underline{f}} + \underbrace{\begin{pmatrix} \underline{f}_p \\ \underline{\tau}_p \end{pmatrix}}_{\underline{w}} = \underline{0}, \tag{2}$$

where $\underline{f}_p$, $\underline{\tau}_p$ are the applied forces and torques, respectively, acting on the platform and $\underline{f}$ is the vector of the wire forces. The matrix $\underline{\underline{A}}$ is referred to as *structure matrix* and permits to investigate existence and quality of the workspace. There are a couple of sophisticated criteria to analyze the workspace based on the structure matrix (see e.g. [2]). For the conceptual study of the wire robot WiRo, the notion of *controllable*

workspace is used [9]. Thus, the following method is used to determine an index κ for the workspace. If the matrix $\underline{\underline{A}}(\underline{r}, \underline{\underline{R}})$ has the maximal rank of six for a given pose $\underline{r}$, $\underline{\underline{R}}$, let $\underline{k}$ be an arbitrary element of the one-dimensional kernel (or nullspace) of $\underline{\underline{A}}$. One can determine such an element $\underline{k}$ for example by application of a singular value decomposition of the matrix $\underline{\underline{A}}$. Then the index κ is defined as

$$\kappa = \begin{cases} \frac{\min(\underline{k})}{\max(\underline{k})} & \text{if} \quad \min \underline{k} > 0 \\ \frac{\max(\underline{k})}{\min(\underline{k})} & \text{if} \quad \max \underline{k} < 0 \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

For $\kappa = 0$ the pose $\underline{r}$, $\underline{\underline{R}}$ does not belong to the workspace and for $\kappa = 1$ the forces in all wires are equal providing an optimal transmission. Note, that the index rates the distribution of forces within the wires.

3 Determination of the Hull of the Workspace

In this paper, the translational workspace for a given orientation of the wire robot is represented by a triangulation of its hull. The idea for the determination of the workspace is to start with an identity sphere in the estimated center $\underline{\widehat{m}}$ of the workspace and to successively extent the sphere in radial direction. Clearly, this assumption may lead to an underestimation of the workspace and the estimation depends on chosen value of $\underline{\widehat{m}}$. Contrary, for most technical application, only robots with a compact workspace are interesting and therefore it seems reasonable to restrict the quick design procedure to such a subspace. The surface of the sphere is approximated by triangles which are created from iterative subdivision of the faces of an octahedron.

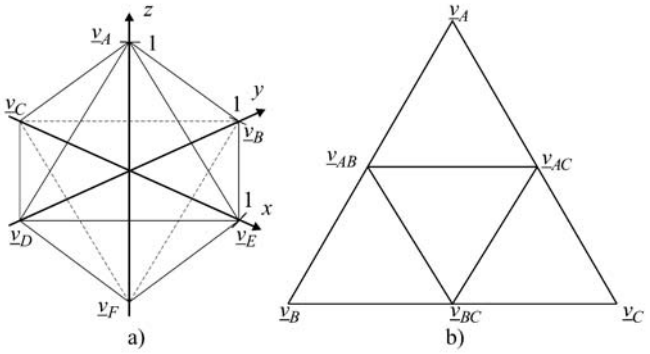


Fig. 2 (a) Unit octahedron. (b) Subdivision step for triangles.

In the first step, the eight faces of an octahedron (Figure 2a) located around the point $\underline{\widehat{m}}$ are described as triplets of vertices, e.g. $F_1 = \{\underline{v}_A, \underline{v}_B, \underline{v}_C\}_i$. Initially, there is a set $\mathcal{L} = \{F_1, \ldots, F_8\}$ containing eight faces. These faces of the octahedron are subdivided into four congruent triangles (Figure 2b). This is done by constructing the three vertices $\underline{v}_{AB}$,$\underline{v}_{AC}$,$\underline{v}_{BC}$ for each triangle F_i in $\mathcal{L}$ and projecting the generated vertices onto a unit sphere

$$\underline{v}_{ij} = \frac{\underline{v}_i + \underline{v}_j}{|\underline{v}_i + \underline{v}_j|}, \quad i, j \in \{A, B, C\}, i \neq j. \tag{4}$$

Then, the original triangle F_i is replaced by the four triangles ($\underline{v}_A$,$\underline{v}_{AB}$, $\underline{v}_{AC}$), ($\underline{v}_B$, $\underline{v}_{AB}$, $\underline{v}_{BC}$), ($\underline{v}_C$, $\underline{v}_{BC}$, $\underline{v}_{AC}$), ($\underline{v}_{AB}$, $\underline{v}_{AC}$, $\underline{v}_{BC}$). This process is repeated n_i times thus generating a set $\mathcal{L}$ containing $n_T = 2^{2n_i+3}$ triangles.

In the second step, the vertices of the triangles are projected onto the hull of the workspace. Starting from the estimated center $\underline{\widehat{m}}$ of the workspace, the line

$$L : \underline{x} = \underline{\widehat{m}} + \lambda \underline{v}_i \quad \lambda \in [0..r] \tag{5}$$

is searched by a regula falsi method for the border of the workspace, which is defined by a given value κ_s. This is done by the algorithm

1. Let $\lambda_{\min} = 0$ be the lower bound and $\lambda_{\max} = r$ be the upper bound.
2. If $\lambda_{\max} - \lambda_{\min} < \varepsilon$, stop the line search.
3. Let $\lambda = \frac{1}{2}(\lambda_{\max} + \lambda_{\min})$ and evaluate $\underline{x} = \underline{\widehat{m}} + \lambda \underline{v}_i$.
4. Calculate the quality index $\kappa(\underline{x})$ of the resulting position $\underline{x}$.
5. If $\kappa > \kappa_s$ then let $\lambda_{\min} = \lambda$ else let $\lambda_{\max} = \lambda$.
6. goto step 2

Finally, one ends up with the vertex $v_i^{(h)} = \underline{\widehat{m}} + \lambda \underline{v}_i$ approximating the hull of the workspace. The correspond triangles are rendered into a new set $\mathcal{L}^{(h)}$. In the presented example these data are directly written in the stereo-lithography data file format (STL) which can be loaded and visualized with most CAD tools (Figure 1a). In order to cope with holes within the workspace, it is trivial to add a second procedure that evaluates the line from the center to boarder with a given step size.

4 Forward Kinematics for Real-time Requirements

For the control of the robot, one has to solve the forward kinematics of the robot, i.e. given the geometry defined by $\underline{a}_i$, $\underline{b}_i$ and the actual length of the wires l_i, one has to determine the pose of the platform. There are two major requirements for that calculation. Firstly, it must be very efficient in order to achieve appropriate cycle times for the controller. The selected controller works with a frequency of 1 kHz due to the high velocities of the platform. For online trajectory planning, a multiple of that value is needed. Secondly, the controller requires strict realtime capabilities of the kinematic function. Therefore, the calculation time must be bounded to a

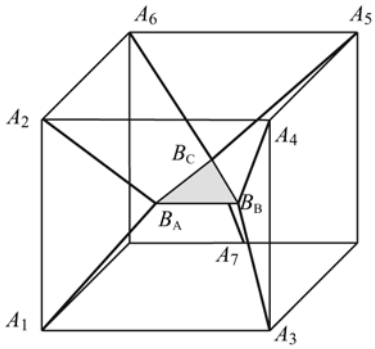


Fig. 3 Six-degree-of-freedom parallel wire robot with three distinct attachment points on the platform.

known value. Thus, explicit formulas are desired and the derived formulas are based on work by Thomas et al. [7] which takes robots with a so-called "3-2-1" configuration into account. For parallel wire robots with seven wires, the distal attachment points $B_1, \ldots, B_7$ can be written as three distinct attachment points B_A, B_B, B_C on the platform. Furthermore, it the configuration it is assumed that either two or three wires share a common attachment point on the movable platform. This configuration is shown in Figure 3 and was used for the proposed robot as well as for a couple of recently built wire robots, since it minimizes the restrictions caused by wire interference elsewhere in workspace [8].

The calculation presented derives the equations in closed-form as they are used for the controller. It is entirely based on a geometric procedure. Firstly, it is recalled that the distal attachment points B_i have to be located on spheres S_i with radii l_i center around the proximal attachment points A_i. If three attachment points on the platform coincide as shown for B_C in Figure 3, their location in space can be calculated from the intersection of three spheres. Once B_C on the platform is known, the distance between B_C and B_A, B_B can be used as distance constraint. Thus, one can apply the same procedure to determine B_A, B_B as well. This provides a closed-form solution for the forward kinematics of the robot. Therefore, the determination of the intersection of three spheres is described in the next section.

An important step in solving this special forward kinematic problem is to compute the intersection of the three given spheres S_1, S_2, S_3 defined by their centers $\underline{m}_1, \underline{m}_2, \underline{m}_3$ and the radii r_1, r_2, r_3, respectively. Generally, there are four possibilities for the number of intersections. There might be no solution, one solution, two symmetric solutions, or an infinite number of solutions. The latter case only occurs if $\underline{m}_1, \underline{m}_2, \underline{m}_3$ define a line. From a kinematic point of view, one of the three constraints degenerates and this happens only in a singular configuration. Firstly, intersection occurs between the spheres only if

$$|r_i - r_j| < |\underline{m}_i - \underline{m}_j| < r_i + r_j \quad \text{for} \quad i, j \in 1, 2, 3, i \neq j \tag{6}$$

holds. Otherwise there is no solution because the distance between the spheres is either too big or too small. Now, the plane E_3 is defined by $\underline{m}_1, \underline{m}_2, \underline{m}_3$. Then its normal vector is given by

$$\underline{n}_3^0 = (\underline{m}_1 - \underline{m}_2) \times (\underline{m}_1 - \underline{m}_3) \tag{7}$$

and the unit normal vector is computed from $\underline{n}_3 = \underline{n}_3^0 |\underline{n}_3^0|^{-1}$. Furthermore, the planes defined by the intersection of S_1, S_2 and S_1, S_3, respectively, are given by

$$E_1 : \underline{x} \cdot \underline{n}_1 = \frac{\underline{m}_2^2 - \underline{m}_1^2 - r_2^2 + r_1^2}{2} \tag{8}$$

$$E_2 : \underline{x} \cdot \underline{n}_2 = \frac{\underline{m}_3^2 - \underline{m}_1^2 - r_3^2 + r_1^2}{2} \tag{9}$$

where the normal vectors $\underline{n}_1, \underline{n}_2$ are constructed from $\underline{n}_1 = \underline{m}_2 - \underline{m}_1, \underline{n}_2 = \underline{m}_3 - \underline{m}_1$. The sought intersection(s) $\underline{x}_i$ of all three spheres is located on the line

$$L : \underline{x} = \underline{x}_0 + \lambda \underline{n}_3^0, \tag{10}$$

where $\underline{x}_0$ is the intersection of the planes E_1, E_2, E_3. Therefore, one can compute $\underline{x}_0$ from the linear system

$$\left[\underline{n}_1 \; \underline{n}_2 \; \underline{n}_3^0\right]^{\mathrm{T}} \underline{x}_0 = \begin{bmatrix} \dfrac{\underline{m}_2^2 - \underline{m}_1^2 - r_2^2 + r_1^2}{2} \\ \dfrac{\underline{m}_3^2 - \underline{m}_1^2 - r_3^2 + r_1^2}{2} \\ m_1 \cdot \underline{n}_3^0 \end{bmatrix}. \tag{11}$$

The parameter λ is then determined from

$$\lambda = \pm \sqrt{r_1^2 - (\underline{x}_0 - \underline{m}_1)^2}. \tag{12}$$

Substituting λ into Eq. (10) yields the two points of intersections.

Since the intersection of two spheres yields two solutions in regular configurations of the robot, up to eight different sets of solutions for B_{A}, B_{B}, B_{C} can be determined. Finally, one has to check which of these solutions is the sought by testing a redundant constraint. This is done by testing if the distance between the points B_{A}, B_{B} matches the known distance between these points.

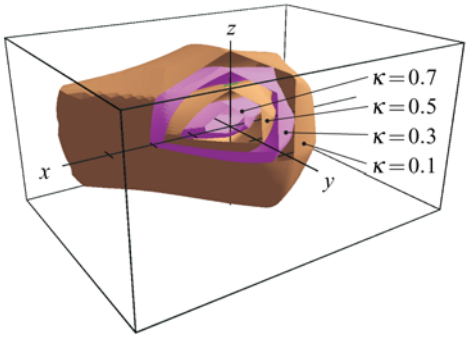


Fig. 4 Workspace of WiRo for different values of $\kappa = \{0.1, 0.3, 0.5, 0.7\}$.

Table 1 Geometrical parameters of the wire robot WiRo.

	Leg i						
Property	1	2	3	4	5	6	7
$\underline{a}_i$	$[0, 0, 0]^{\mathrm{T}}$	$[X, 0, 0]^{\mathrm{T}}$	$[X, Y, 0]^{\mathrm{T}}$	$[0, Y, 0]^{\mathrm{T}}$	$[0, 0, Z]^{\mathrm{T}}$	$[X, 0, Z]^{\mathrm{T}}$	$[\frac{X}{2}, Y, Z]^{\mathrm{T}}$
$\underline{b}_i$	$[-\frac{B}{2}, 0, 0]^{\mathrm{T}}$	$[\frac{B}{2}, 0, 0]^{\mathrm{T}}$	$[0, W, 0]^{\mathrm{T}}$	$[0, W, 0]^{\mathrm{T}}$	$[-\frac{B}{2}, 0, 0]^{\mathrm{T}}$	$[\frac{B}{2}, 0, 0]^{\mathrm{T}}$	$[0, W, 0]^{\mathrm{T}}$

5 Simulation Results

5.1 Workspace Determination

In this section, some simulation results for the determination of the hull of the workspace are presented. The geometrical parameters of the machine frame and the movable platform are given in Table 1, where the following values of the parameter are used $B = 0.25$, $W = 0.25$, $X = 4$, $Y = 3$, $Z = 2$. Figure 4 presents the results of the workspace determination for different values $\kappa = \{0.1, 0.3, 0.5, 0.7\}$. The accuracy for the line search was $\varepsilon = 0.0001$ and the subdivision depth was set to $n_i = 5$ generating $2^{13} = 8192$ triangles. The computational time with a straightforward implementation in Matlab was 12 s for each hull on an Intel Core 2 Duo, 2.4 GHz. Optimized implementations are expected to compute each hull in less then 1 second enabling the user to perform the design of the workspace in an interactive manner.

5.2 Forward Kinematics

The algorithm described in Section 4 was implemented in plane C++. The implementation is free from any loops or iterative procedures. No numerical libraries are

used beside the standard libraries. The algorithm shows a good computational performance with 87673 evaluations per second (measured on an Intel Core 2 Duo, 2.0 GHz). Based on this results, no difficulties are expected for the desired cycle time of 1 ms and the integration into the industrial real-time controller.

6 Conclusions

This paper presents some tools which are useful to develop and control a wire robot in an industrial environment. The algorithm used for workspace determination is simple but effective and allows the user to quickly determine an estimation of the workspace. The results can be fed back into the CAD system for evaluation and layout planning. The computation of the forward kinematics is based on simple geometric considerations and allows for real-time capable computation. Experimental evaluations and measurement is planned for future work. There will be a focus on studying the impact of simplifications in the kinematic functions and shortcomings in the winches as well as in the mechanical realization of the attachment points on the platform.

References

1. Bruckmann, T., Pott, A. and Hiller, M., Calculating force distributions for redundantly actuated tendon-based Stewart platforms. In *Advances in Robot Kinematics*, Ljubljana, Slovenia, pp. 403–412, Springer (2006).
2. Gouttefarde, M., Merlet, J.-P. and Daney, D., Determination of the wrench-closure workspace of 6-dof parallel cable-driven mechanisms. In *Advances in Robot Kinematics*, Ljubljana, Slovenia, pp. 315–322. Springer (2006).
3. Gouttefarde, M., Merlet, J.-P. and Daney, D., Wrench-feasible workspace of parallel cable-driven mechanisms. In *Proceedings IEEE International Conference on Robotics and Automation*, Roma, Italy, pp. 1492–1497 (2007).
4. Hiller, M., Fang, S., Mielczarek, S., Verhoeven, R. and Franitza, D., Design, analysis and realization of tendon-based parallel manipulators. *Mechanism and Machine Theory* **40**(4), 429–445 (2005).
5. Merlet, J.-P. and Daney, D., A new design for wire-driven parallel robot. In *Proceedings 2nd International Congress, Design and Modelling of Mechanical Systems* (2007).
6. Ming, A. and Higuchi, T., Study on multiple degree-of-freedom positioning mechanism using wires (Part 1) – Concept, design and control. *International Journal of the Japanese Society for Precision Engineering* **28**(2), 131–138 (1994).
7. Thomas, F., Ottaviano, E., Ros, L. and Ceccarelli, M., Uncertainty model and singularities of 3-2-1 wire-based tracking systems. In *Advances in Robot Kinematics*, Caldes de Malavalla, pp. 107–116 (2002).
8. Verhoeven, R., Analysis of the Workspace of Tendon-Based Stewart Platforms. PhD Thesis, University of Duisburg-Essen (2004).
9. Verhoeven, R. and Hiller, M., Estimating the controllable workspace of tendon-based Stewart platforms. In Advances in Robot Kinematics, Portorož, Slovenia, pp. 277–284 (2000).

Kinematic Modeling and Workspace Generation for a New Parallel Robot Used in Minimally Invasive Surgery

Doina Pisla, Nicolae Plitea and Calin Vaida

Technical University of Cluj-Napoca, Constantin Daicoviciu 15,
RO-400020 Cluj-Napoca, Romania,
e-mail: {doina.pisla, nicolae.plitea, calin.vaida}@mep.utcluj.ro

Abstract. Parallel robots offer higher stiffness and smaller mobile mass than serial ones, thus allowing faster and more precise manipulations that fit medical applications, especially surgery. This paper presents the kinematic modeling and a method for workspace generation for a new parallel robot used for minimally invasive surgery. Using the developed model of the parallel robot and the kinematic modeling, some numerical and simulation tests are presented.

Key words: parallel robot, minimally invasive surgery, kinematics, workspace.

1 Introduction

Brown (2005) and Plitea et al. (2007) already showed that the progresses from engineering and medicine have opened the way for the use of the robots in the operating rooms. Robots are useful tools in minimally invasive surgery (MIS), providing benefits such as reduction in hand tremor, navigation, and workspace scaling.

Regarding the application of robots for medical applications there are some investigators focused on exploring the capabilities of robots in this field (Ben-Porat, 2000; Glozman, 2001; Grace, 1993). The AESOP robotic arm, used to guide a tiny camera inside the body, was the first robotic system used in surgery dating from 1993 (Brown University, 2005). It was produced by Computer Motion, which developed several such versions of AESOP until they created Zeus^{TM} Robotic Surgical System with three robotic arms attached on the side of the operating table.

A competitor of Computer Motion, Intuitive Surgical, designed another revolutionary equipment, da Vinci^{TM} Surgical System, which became the market competitor of Zeus until 2003 when the two companies merged (Brown University, 2005).

Most of the robots, which assist the surgeons, are serial robots (Brown University, 2005). The serial module generates a large workspace while the parallel module is steadier and offers a high accuracy during the surgical operation.

The actual robotic systems have drawbacks such as: they are large and cumbersome, occupying large volumes around the operating table and above the patient; the surgeon's console ergonomics imposes a very high number of training hours; the

Jadran Lenarčič and Philippe Wenger (eds.), Advances in Robot Kinematics: Analysis and Design, 459–468.

surgeon relies only on visual feedback loosing the tactile facilities; and the current systems are limited to certain types of surgery and the market price is prohibitive.

Parallel robots offer a higher stiffness and smaller mobile mass than serial robots, thus they allow faster and more precise manipulations (Merlet, 2000). In the field of robotics assisted surgery, the drawbacks of serial robots motivate the search of task oriented robot architectures that best fit a specific group of medical applications.

Professor Moshe Shoham performed extensive studies regarding the performances of the surgeon in endoscopy, focused on the fulcrum effect (Ben-Porat et al., 2000) and the development of new miniaturized robots which can be mounted on the bones and a research regarding the different methods for scanning and surface recognition of bones before the surgical intervention (Glozman et al., 2001).

Simaan et al. (2004) developed a new robotic model designed for the MIS of the throat, which could manipulate of two up to three instruments for the suture and handling of soft tissue, using as single access route, the mouth of the patient.

The requirements for minimally invasive robotic assisted surgery are given in Ortmaier et al. (2004).

Lum et al. (2004) presented the kinematics of a serial spherical mechanism used for MIS procedures. Beasly et al. (2004) investigated the kinematic error correction for minimally invasive surgical robots.

In the case of a robotic of a robotic system for MIS, previous studies have shown that the parallel and hybrid structures are more adequate than serial ones in this field. They represent a boost in robotic surgery in terms of increased performances and safety and lower costs.

The paper is organized as follows: Section 2 is dedicated to the presentation of the parallel mechanism for MIS. Section 3 deals with the kinematics of the studied structure. Section 4 presents the workspace generation of the mechanism. Some numerical and simulation results obtained from kinematics are presented in Section 5. The conclusions of this work are presented in Section 6.

2 The New PARAMIS Parallel Robot

We established at first the requirements for a base robotic module for surgical instruments positioning (Plitea, 2006): the robot should have low sizes; the robot assisted procedure must present a minimal damage to the patient; the robot structure must be rigid and stable in the Operating Room (OR); the robot should have 3 DOF (1 translation and 2 rotations) which should cover the entire surgical field.

Starting from these requirements, a new parallel structure – PARAMIS has been developed, which could be used for surgical instruments positioning (Plitea et al., 2007), see Figure 1. The low-cost structure will allow a wider spread or the robot in the OR, an easier acceptance and a better feedback for further improvements. The structure has 3 DOF and it consists of three actuated joints (two of them are prismatic and one rotational one). The passive joints are two cylindrical joints, one prismatic joint and one Cardan joint. The particularity of the motion is the fact that

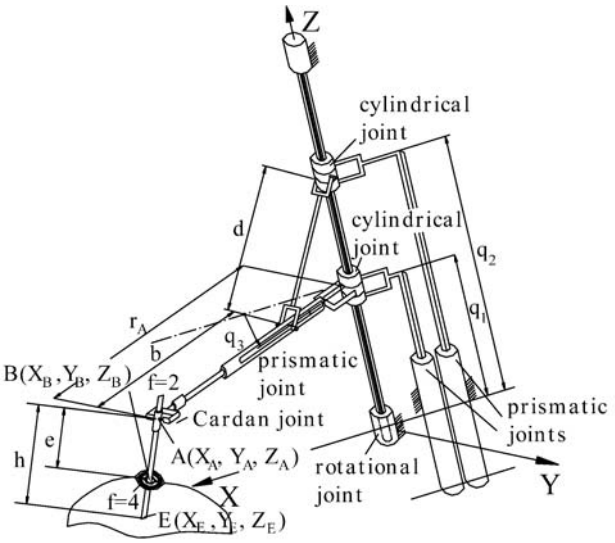


Fig. 1 Kinematic scheme of the parallel robot for MIS.

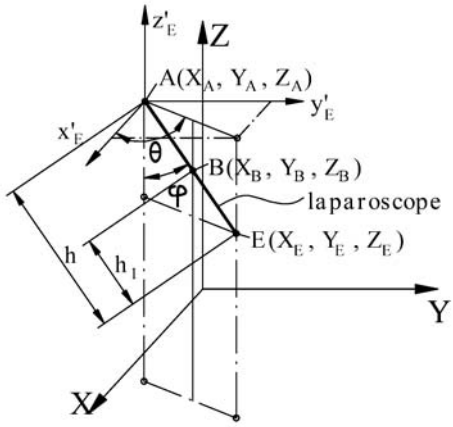


Fig. 2 The angles of the passive Cardan joint for PARAMIS robot.

the endoscope will move around a fixed point in space, which is the entrance point of the trocar in the abdominal wall of the patient (the point B whose coordinates are X_B, Y_B, Z_B).

The geometrical parameters of the parallel robot are represented by b, d, h; X_B, Y_B, Z_B, Figure 1; the angles φ, θ are also presented Figure 2. The endoscope can be positioned in any point of the abdominal area using 3 DOF of the robot to offer the surgeon the best possible details of the surgical field.

3 Kinematics

In Plitea et al. (2007) the inverse and direct geometric models of the structure were already presented. In order to obtain the kinematic equations, we start from this geometric model. The equations which define the coordinates of E point located on the laparoscope are:

$$\begin{cases} f_1(X_E, q_1, q_2, q_3) \equiv X_E + h\dfrac{X_B - X_E}{h_1} - \left[b + \sqrt{d^2 - (q_2 - q_1)^2}\right]\cos q_3 = 0, \\ f_2(Y_E, q_1, q_2, q_3) \equiv Y_E + h\dfrac{Y_B - Y_E}{h_1} - \left[b + \sqrt{d^2 - (q_2 - q_1)^2}\right]\sin q_3 = 0, \\ f_3(Z_E, q_1) \equiv Z_E + h\dfrac{Z_B - Z_E}{h_1} - q_1 = 0, \end{cases} \tag{1}$$

where

$$h_1 = \sqrt{(X_B - X_E)^2 + (Y_B - Y_E)^2 + (Z_B - Z_E)^2}, \tag{2}$$

see Figure 2.

Using the matrix representation, the kinematic model for velocities is:

$$A\dot{X} + B\dot{q} = 0, \tag{3}$$

where A is the direct Jacobian matrix and B is the inverse Jacobian matrix. Using (1), the direct Jacobian matrix A has the form:

$$A = \begin{bmatrix} 1 - \frac{h}{h_1} + \frac{h(X_B - X_E)^2}{h_1^3} & \frac{h(X_B - X_E)(Y_B - Y_E)}{h_1^3} & \frac{h(X_B - X_E)(Z_B - Z_E)}{h_1^3} \\ \frac{h(Y_B - Y_E)(X_B - X_E)}{h_1^3} & 1 - \frac{h}{h_1} + \frac{h(Y_B - Y_E)^2}{h_1^3} & \frac{h(Y_B - Y_E)(Z_B - Z_E)}{h_1^3} \\ \frac{h(Z_B - Z_E)(X_B - X_E)}{h_1^3} & \frac{h(Z_B - Z_E)(Y_B - Y_E)}{h_1^3} & 1 - \frac{h}{h_1} + \frac{h(Z_B - Z_E)^2}{h_1^3} \end{bmatrix} \tag{4}$$

and the inverse Jacobian matrix B has the form:

$$B = \begin{bmatrix} \cos q_3 \frac{q_1 - q_2}{\sqrt{d^2 - (q_2 - q_1)^2}} & \cos q_3 \frac{q_2 - q_1}{\sqrt{d^2 - (q_2 - q_1)^2}} & \sin q_3 \left[b + \sqrt{d^2 - (q_2 - q_1)^2}\right] \\ \sin q_3 \frac{q_1 - q_2}{\sqrt{d^2 - (q_2 - q_1)^2}} & \sin q_3 \frac{q_2 - q_1}{\sqrt{d^2 - (q_2 - q_1)^2}} & -\cos q_3 \left[b + \sqrt{d^2 - (q_2 - q_1)^2}\right] \\ -1 & 0 & 0 \end{bmatrix}. \tag{5}$$

In the case of direct kinematic model for velocities the driving velocities $\dot{q} = [\dot{q}_1, \dot{q}_2, \dot{q}_3]^T$ are given and the end-effector velocities $\dot{X} = [\dot{X}_E, \dot{Y}_E, \dot{Z}_E]^T$ are computed. As input data for this kinematic model we have the inverse and geometric model and the geometric parameters of the PARAMIS parallel mechanism. Using (3) we obtain:

$$\dot{X} = -A^{-1}B\dot{q}. \tag{6}$$

The inverse of A matrix has been computed:

$$A^{-1} = \frac{h_1}{(h_1 - h)} \tag{7}$$

$$\times \begin{bmatrix} 1 - \frac{h}{h_1^3}(X_B - X_E)^2 & -\frac{h}{h_1^3}(Y_B - Y_E)(X_B - X_E) & -\frac{h}{h_1^3}(Z_B - Z_E)(X_B - X_E) \\ -\frac{h}{h_1^3}(X_B - X_E)(Y_B - Y_E) & 1 - \frac{h}{h_1^3}(Y_B - Y_E)^2 & -\frac{h}{h_1^3}(Z_B - Z_E)(Y_B - Y_E) \\ -\frac{h}{h_1^3}(X_B - X_E)(Z_B - Z_E) & -\frac{h}{h_1^3}(Y_B - Y_E)(Z_B - Z_E) & 1 - \frac{h}{h_1^3}(X_B - X_E)^2 \end{bmatrix}.$$

Using (5) and (7), the vector $\dot{X}$ components are computed.

In the case of inverse kinematic model for velocities, the end-effector velocities $\dot{X} = [\dot{X}_E, \dot{Y}_E, \dot{Z}_E]^T$ are given and the driving velocities $\dot{q} = [\dot{q}_1, \dot{q}_2, \dot{q}_3]^T$ are computed. As input data for this kinematic model we have the inverse and geometric model and the geometric parameters of the PARAMIS parallel mechanism. Using (3) we obtain:

$$\dot{q} = -B^{-1} A \dot{X}. \tag{8}$$

The inverse of the B matrix has been computed:

$$B^{-1} = \begin{bmatrix} 0 & 0 & -1 \\ \cos q_3 \frac{\sqrt{d^2 - (q_2 - q_1)^2}}{q_2 - q_1} & \sin q_3 \frac{\sqrt{d^2 - (q_2 - q_1)^2}}{q_2 - q_1} & -1 \\ \sin q_3 \frac{1}{b + \sqrt{d^2 - 9q_2 - q_1)^2}} & -\cos q_3 \frac{1}{b + \sqrt{d^2 - (q_2 - q_1)^2}} & 0 \end{bmatrix}. \tag{9}$$

Using (4) and (9) the vector $\dot{q}$ components are computed.

Through the differentiation of Equation (3), we obtain

$$A\ddot{X} + \dot{A}\dot{X} + B\ddot{q} + \dot{B}\dot{q} = 0. \tag{10}$$

From (10), if the end-effector velocities $\ddot{X} = [\ddot{X}_E, \ddot{Y}_E, \ddot{Z}_E]^T$ are given, the driving accelerations $\ddot{q} = [\ddot{q}_1, \ddot{q}_2, \ddot{q}_3]^T$ could be computed. As input data in this case, we have the inverse and geometric model, the inverse and direct kinematic model for velocities and the geometric parameters of the PARAMIS parallel mechanism.

4 Workspace Generation

For the geometrical generation of the workspace of point E (the laparoscope extremity where the camera is placed) the first step is the determination of the workspace of point A, keeping in mind that point B is fixed in space and AE has a constant length. The workspace of point A can be easily determined, by computing the cross-sections within the workspace in a plane parallel to OXY which for the limits imposed for q_1 and q_2 will have the same shape for any position of the Z axis.

For the workspace generation for point E, it has to be considered that the laparoscope (or any other surgical instrument) has a constant length and will always

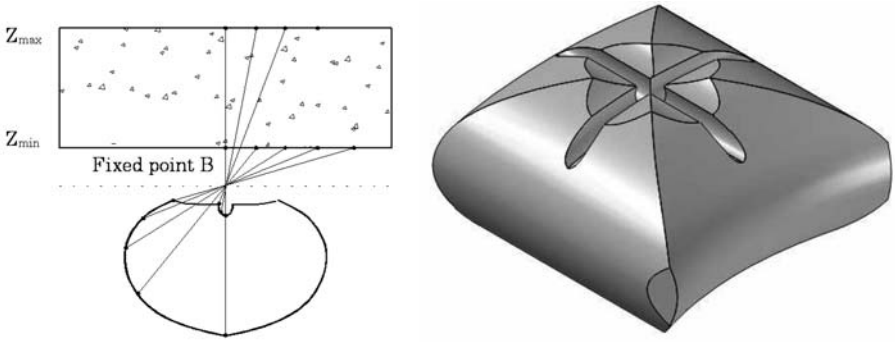


Fig. 3 Workspace generation of E point for PARAMIS.

pass through a fixed point in space. The workspace of point E is obtained by generating two intermediate workspaces whose intersection will represent the effective workspace.

The first intermediate workspace is generated by projecting the point A from sections parallel with OXY plane. The second one is generated from sections of the workspace of point A perpendicular on the plane OXY. The intersection of the two intermediary volumes will result in the workspace of point E presented in Figure 3.

5 Numerical and Simulation Results

The kinematic modeling has been implemented in the kinematic module (kinematics) of the developed simulation system for parallel robots (Pisla, 2001, 2005). In the simulation is included the parallel robot with the instrument and the virtual human body (Figure 4).

The simulation enabled the validation of the structure from the points of view of both engineers and surgeons and opened the way for the next step in the PARAMIS robotic design evolution, the construction of an experimental model (Figure 5).

For the parallel robot constructive design, an optimal solution for the building of the robot column (base) was proposed, which concentrates all the actuation motors.

For the experimental model building, the following aspects have been taken into account: the robot should be modular, the actuation system should be identical for all motors and with the same control interface, in our case electrical actuation; as it is possible for the robot to suffer some small modifications, the components should be simple and cheap regarding the manufacturing costs and processing technologies.

For the numerical simulation, the experimental model was considered: $b = 256.4$ mm; $h = 270$ mm; $d = 545.4$ mm. The coordinates of the point B are: $X_B = 679.2$ mm; $Y_B = 0$ mm; $Z_B = 400.1$ mm. The starting point of the moving path are: $X_E = 679.2$ mm; $Y_E = 0$; $Z_E = 281.1$ mm. The inverse geometric model delivers the starting joint coordinates: $q_1 = 551.1$ mm; $q_2 = 847.7$ mm; $q_3 = 0$.

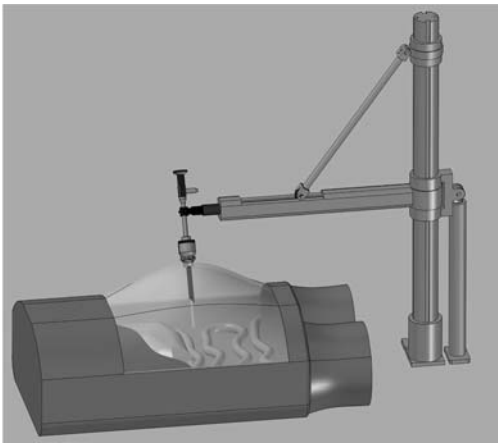


Fig. 4 The virtual robotic system.

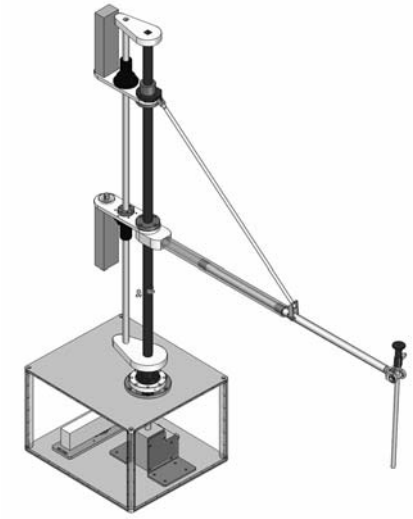


Fig. 5 The robot experimental model

For the validation of the kinematic model of the robot, direct and inverse kinematic models have been numerically solved.

For inverse kinematic problem, the selected trajectory of the end-effector is a straight line parallel to the Z axis starting from $X_E = 679.2$ mm; $Y_E = 0$ mm and ending with $X_E = 679.21$ mm; $Y_e = 0$; $Z_1 = 331.1$ mm, the maximum velocity being $v_{\max} = 10$ mm/s and the maximum acceleration $a_{\max} = 5$ mm/s^2. The kinematic diagrams valid for $s = X_E$, $v = \dot{X}_E$, $a = \ddot{X}_E$ are represented in Figure 6.

For the case where $t_1 < t_2$ (Figure 6a), the following equations are valid:

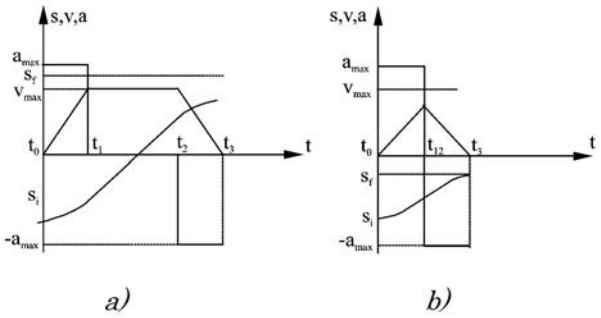


Fig. 6 Kinematic diagrams of the test trajectories for two time conditions.

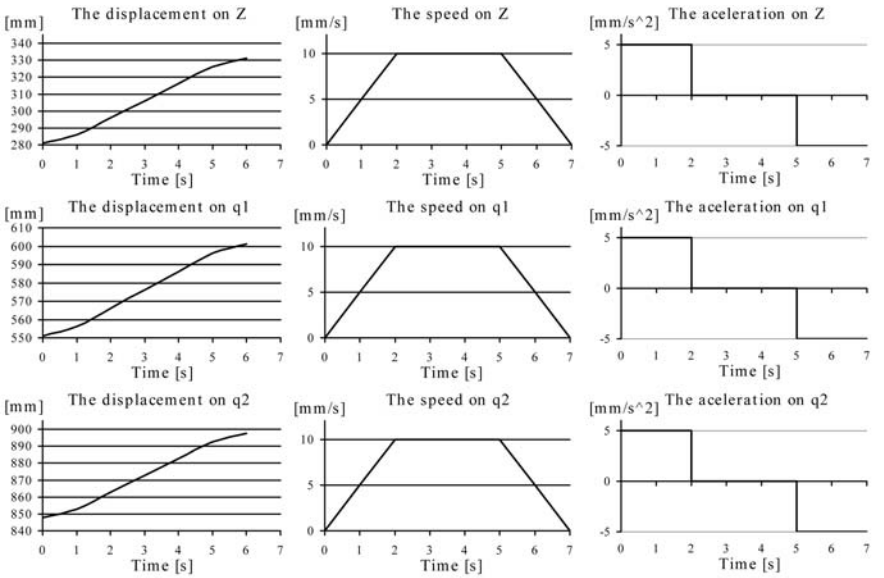


Fig. 7 Kinematic results of the inverse kinematic model for PARAMIS robot.

$$t_1 = \frac{v_{\max}}{a_{\max}}, \quad t_2 = \frac{|s_f - s_i|}{v_{\max}}, \quad t_3 = \frac{|s_f - s_i|}{v_{\max}} + \frac{v_{\max}}{a_{\max}}, \tag{11}$$

whereas for the case where $t_1 > t_2$ (Figure 6b), we take:

$$t_{12} = \sqrt{\frac{|s_f - s_i|}{a_{\max}}}, \quad t_3 = 2t_{12}. \tag{12}$$

The results from the inverse kinematics are presented in Figure 7.

The diagrams represent a linear motion, along the Z axis, of the point E of the laparoscope, which is obtained by the simultaneous motions of the active joints q_1 and q_2. As shown in the graphics the accelerations and speeds for all components

are equal as well as the variation of the displacement. For this type of motion the joint coordinate $q_3 = 0$.

6 Conclusions

The kinematics of a new developed parallel robot for camera positioning in MIS was presented. An analytical approach for kinematics solving was suggested. A geometrical method for the workspace generation of the parallel robot was proposed. The obtained numerical and simulation results have shown that the kinematic models could be successfully implemented in the control algorithms of the experimental model, which will be further build. The results obtained are useful for the designers not only to understand the distribution of characteristics of the workspaces for various geometrical parameters of parallel structure, but also to optimize the PARAMIS parallel robot.

The future research will be directed to the dynamic modeling of this new parallel structure for MIS, the building of its prototype and experimental testing under laboratory conditions.

Acknowledgements

The authors gratefully acknowledge the financial support provided by the research grants awarded by the Romanian Ministry of Education and Research and the "Institutional Academic Cooperation Research Grant" between the Technical University Braunschweig, Germany and the Technical University of Cluj-Napoca, Romania, awarded by Alexander von Humboldt foundation.

References

Beasley, R.A., Howe, R.D. and Dupont, P.E. (2004), Kinematic error correction for minimally invasive surgical robots, in *Proceedings of ICRA '04*, New Orleans, USA, Vol. 1, pp. 358–364.

Ben-Porat, O., Shoham, M. and Meyer, J. (2000), Control design and task performance in endoscopic teleoperation, Presence, *Massachusetts Institute of Technology* **9**(3), 56–267.

Brown University, Division of Biology and Medicine (2005).

Glozman, D., Shoham, M. and Ficher, A. (2001), A surface-matching technique for robot-assisted registration, *Computer Aided Surgery* **6**, 259–269.

Grace, K.W., Colgate, J.E., Glucksberg, M.R. and Chun, J.H. (1993), A six degree of freedom micromanipulator for ophthalmic surgery, in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 630–635.

Lum, M.J.H., Rosen, J., Sinanan, M.N. and Hannaford, B. (2004), Kinematic optimization of a spherical mechanism for a minimally invasive surgical robot, in *Proceedings of the 2004 IEEE ICRA 2004*, New Orleans, USA, pp. 829–834.

Merlet, J-P. (2000), *Parallel Robots*, 2nd Edition, Kluwer, Dordrecht.

Ortmaier, T., Weiss, H. and Falk, V. (2004), Design requirements for a new robot for minimally invasive surgery, *Industrial Robot, An International Journal* **31**(6), 493–498.

Pisla, D. (2005), *Kinematic and Dynamic Modeling of Parallel Robots* Casa carti de stiinta Publishing House, Cluj-Napoca [in Romanian].

Pisla, D. and Pisla, A. (2001), Effiziente dynamische Rechnersimulation für Parallelroboter, *ZAMM* **81**(Suppl. 5), 277–278.

Plitea, N., Hesselbach, J., Pisla, D., Raatz, A., Vaida, C., Wrege, J. and Burisch, A. (2006), Innovative development of parallel robots and microrobots, *Acta Tehnica Napocensis, Series of Applied Mathematics and Mecanics* **5**(49), 5–26.

Plitea, N., Hesselbach, J., Vaida, C., Raatz, A., Pisla, D., Budde, C., Vlad, L., Burisch, A. and Senner, R. (2007), Innovative development of surgical parallel robots, *Acta Electronica, Mediamira Science, Cluj-Napoca*, 201–206.

Simaan, N., Taylor, R. and Flint, P. (2004), A dexterous system for laryngeal surgery, in *Proceedings of ICRA '04*, New Orleans, USA, pp. 351–357.

Author Index

Alba-Gómez, O.G., 253
Altuzarra, O., 301
Ambike, S., 205
Angeles, J., 225, 353
Barthélemy, S., 399
Bayat, J., 419
Ben-Horin, P., 49, 111
Berselli, G., 291
Bidaud, P., 399
Bonev, I.A., 345
Boudreau, R., 243
Briot, S., 345
Budde, C., 371
Cardou, P., 353
Caro, S., 49, 311
Carretero, J.A., 243, 381
Carricato, M., 3
Ceccarelli, M., 411
Chablat, D., 49, 59, 311
Company, O., 71
Conconi, M., 3
Crane III, C.D., 419
De Santos, A., 429
De Sapio, V., 263
Demircan, E., 263
Di Gregorio, R., 361
Doncieux, S., 127
Donelan, P., 147, 185
Dubowsky, S., 391
Ebrahimi, I., 243
Gauthier, J.-F., 225
Gogu, G., 21
Gonzalez-Vega, L., 283
Gosselin, C.M., 331
Gouttefarde, M., 71
Grand, Ch., 127
Hernandez, A., 301
Hervé, J.M., 439
Hesselbach, J., 371
Hubert, J., 13
Husty, M., 411
Janiak, M., 137
Jüttler, B., 283
Kanaan, D., 59
Karger, A., 275
Khan, M.A., 119
Khatib, O., 263
Kong, X., 29
Krut, S., 71
Lahouar, S., 235
Larochelle, P., 79
Last, P., 371
Ledezma Rubio, Y., 89
Lee, C.-C., 439
Le Vey, G., 177
Macho, E., 301
Małek, L., 155
Martinelli, P., 127
McCarthy, J.M., 101
Merlet, J.-P., 13
Molfino, R., 119
Mouret, J.-B., 127
Müller, A., 215
Nawratil, G., 321
Nokleby, S., 225
O'Brien, S.M., 381
Ottaviano, E., 411
Pamanes, J.A., 253
Parenti Castelli, V., 291
Parkin, I.A., 193

Pierrot, F., 71
Pinto, C., 301
Pisla, D., 459
Plante, J.-S., 391
Plitea, N., 459
Pott, A., 451
Quennouelle, C., 331
Raatz, A., 371
Rabl, M., 283
Rakotomanga, N., 311
Ramdani, N., 71
Rico, J.M., 215
Roberts, R., 419
Rojas Salgado, A., 89
Romdhane, L., 235
Schütz, D., 371
Schmiedeler, J.P., 205
Selig, J.M., 147
Sentis, L., 263
Shoham, M., 49
Siciliano, B., 429
Soh, G.S., 101
Tanev, T.K., 39
Tchoń, K., 137, 155
Thomas, F., 111
Vaida, C., 459
Vassura, G., 291
Vertechy, R., 291
Vikas, V., 419
Wenger, P., 49, 59, 253
Wohlhart, K., 165
Zeghloul, S., 235
Zoppi, M., 119

Subject Index

accelerometer, 353
accuracy analysis, 345
actuation schemes, 243
angular velocity, 353
approximate motion synthesis, 79
approximation, 137
architectural singularity, 275
architecture singular manipulators, 321
arm kinematics, 205
assembly modes, 301
Assur groups, 165
binary actuation, 391
biomechanics, 353
cable-driven robot, 71
calibration, 371, 381, 391
Cardanic motion, 29
Cartesian trajectory planning, 225
characteristic curve, 283
compliant mechanisms, 291
compliant parallel mechanisms, 331
configuration space, 215, 301
connectivity, 21
constraint singularity, 21
control, 127
convergence, 155
curvature theory, 205
cusp points, 301
cylindrical singularity surface, 321
cylindroids, 439
Delassus' linkages, 439
design, 71, 89
determinants, 147
dielectric elastomer, 391
dielectric elastomer actuators, 291
distance metrics, 345
distribution, 137
dynamical singularities, 177
energy saving, 89
envelope, 283
equilibrium robustness, 399
Euler angles, 345
exact motion synthesis, 79
extended Jacobian inverse, 137
finite displacement screw, 193
flapping-wing, 127
forward kinematics, 29, 451
frame design, 291
generic manipulator, 185
geometric algebra, 39
Grassmann–Cayley algebra, 49, 59
higher order analysis 215
hinged parallelogram, 439
human motion, 399
human motion synthesis, 263
instantaneous kinematics, 3
inverse kinematics, 177, 429
Jacobian inverse kinematics, 155
Jacobian pseudo-inverse, 137
Jacobians, 147
kinematic redundancy, 243
kinematic synthesis, 101
kinematics, 39, 193, 381, 459
kinemato-static model, 331
leading joint hypothesis, 205
level set, 411
Lie bracket, 111
Lie group, 185
manipulators, 89
marker space, 263
mechanism synthesis, 361

metrics, 361
minimally invasive surgery, 119, 459
mobile manipulator, 155
mobility, 21, 215
modeling, 127
motor control, 205
multiple-point control, 429
nonholonomic joint, 111
operational space formulation, 263
optimal control, 177
optimization, 225, 253
overconstraint, 21
parallel, 391
parallel kinematics, 371
parallel manipulator(s), 29, 39, 59, 243, 253, 381
parallel mechanism(s), 3, 119, 127, 311
parallel robot(s), 49, 71, 111, 235, 459
parallel singularities, 89
parallelogram, 439
parameterization, 283
path planning, 235, 243
performance, 155
pitch, 193
planar mechanisms, 419
planar parallel manipulator(s), 321, 301
planar parallel robots, 13
position analysis, 165
positioning precision, 361
postural stability, 399
posture change, 411
prioritization, 263
product of exponentials, 185
pseudo-planar motion, 439
redundancy, 21, 137
regular dextrous workspace, 311
residual radius, 399
rigid-body acceleration field, 353
robot kinematics, 137
robotic surgery, 119
rotational errors, 345
SCARA, 225
screw theory, 59, 193
SE(3), 361
self-motion(s), 29, 275
serial manipulators, 411
singular posture, 439
singularity(ies), 3, 13, 29, 39, 49, 59, 147, 185, 215, 235, 301, 371, 411
singularity robust algorithm, 155
skeleton algorithm, 429
software, 49
spherical linkage synthesis, 101
spherical mechanisms, 79
spherical parallel manipulator, 101
static workspace, 13
Stewart–Gough platforms, 275, 321
stiffness matrix, 331
strap-down inertial measurement unit, 353
support function, 283
Sylvester's elimination method, 165
task accessibility, 253
task/posture decomposition, 263
ten-bar spherical linkage, 101
tensegrity, 419
trajectory planning, 253
transversality 185
twist, 439
UAV, 127
underactuated robot, 111
variable actuated mechanism, 311
whole-body modelling, 429
wire robot, 451
working mode, 371
workspace, 71, 411, 451, 459