



robotics

Kinematics and Robot Design I, KaRD2018

Edited by

Raffaele Di Gregorio

Printed Edition of the Special Issue Published in *Robotics*

Kinematics and Robot Design I, KaRD2018

Kinematics and Robot Design I, KaRD2018

Editor

Raffaele Di Gregorio

MDPI • Basel • Beijing • Wuhan • Barcelona • Belgrade • Manchester • Tokyo • Cluj • Tianjin



Editor

Raffaele Di Gregorio
University of Ferrara
Italy

Editorial Office

MDPI
St. Alban-Anlage 66
4052 Basel, Switzerland

This is a reprint of articles from the Special Issue published online in the open access journal *Robotics* (ISSN 2218-6581) (available at: https://www.mdpi.com/journal/robotics/special_issues/KARD).

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

LastName, A.A.; LastName, B.B.; LastName, C.C. Article Title. <i>Journal Name</i> Year , <i>Volume Number</i> , Page Range.
--

ISBN 978-3-0365-1018-7 (Hbk)

ISBN 978-3-0365-1019-4 (PDF)

© 2021 by the authors. Articles in this book are Open Access and distributed under the Creative Commons Attribution (CC BY) license, which allows users to download, copy and build upon published articles, as long as the author and publisher are properly credited, which ensures maximum dissemination and a wider impact of our publications.

The book as a whole is distributed by MDPI under the terms and conditions of the Creative Commons license CC BY-NC-ND.

Contents

About the Editor	vii
Preface to “Kinematics and Robot Design I, KaRD2018”	ix
Alfredo Valverde and Panagiotis Tsiotras Spacecraft Robot Kinematics Using Dual Quaternions Reprinted from: <i>Robotics</i> 2018 , 7, 64, doi:10.3390/robotics7040064	1
Georg Nawratil Hexapods with Plane-Symmetric Self-Motions Reprinted from: <i>Robotics</i> 2018 , 7, 27, doi:10.3390/robotics7020027	25
Giovanni Legnani and Irene Fassi Kinematics Analysis of a Class of Spherical PKMs by Projective Angles Reprinted from: <i>Robotics</i> 2018 , 7, 59, doi:10.3390/robotics7040059	43
Ivan Giorgio and Dionisio Del Vescovo Non-Linear Lumped-Parameter Modeling of Planar Multi-Link Manipulators with Highly Flexible Arms Reprinted from: <i>Robotics</i> 2018 , 7, 60, doi:10.3390/robotics7040060	57
Liviu Ciupitu Adaptive Balancing of Robots and Mechatronic Systems Reprinted from: <i>Robotics</i> 2018 , 7, 68, doi:10.3390/robotics7040068	71
Ali Bin Junaid, Alejandro Diaz De Cerio Sanchez, Javier Betancor Bosch, Nikolaos Vitzilaios and Yahya Zweiri Design and Implementation of a Dual-Axis Tilting Quadcopter Reprinted from: <i>Robotics</i> 2018 , 7, 65, doi:10.3390/robotics7040065	83
Matteo Palpacelli, Luca Carbonari, Giacomo Palmieri and Massimo Callegari Design of a Lockable Spherical Joint for a Reconfigurable 3-URU Parallel Platform Reprinted from: <i>Robotics</i> 2018 , 7, 42, doi:10.3390/robotics7030042	103
Raffaele Di Gregorio, Mattia Cattai and Henrique Simas Performance-Based Design of the CRS-RRC Schoenflies-Motion Generator Reprinted from: <i>Robotics</i> 2018 , 7, 55, doi:10.3390/robotics7030055	111
Giovanni Boschetti and Alberto Trevisani Cable Robot Performance Evaluation by Wrench Exertion Capability Reprinted from: <i>Robotics</i> 2018 , 7, 15, doi:10.3390/robotics7020015	127
Guojun Liu Optimal Kinematic Design of a 6-UCU Kind Gough-Stewart Platform with a Guaranteed Given Accuracy Reprinted from: <i>Robotics</i> 2018 , 7, 30, doi:10.3390/robotics7020030	143
Marco Faroni, Manuel Beschi, Nicola Pedrocchi and Antonio Visioli Viability and Feasibility of Constrained Kinematic Control of Manipulators Reprinted from: <i>Robotics</i> 2018 , 7, 41, doi:10.3390/robotics7030041	153

Naim Md Lutful Huq, Md Raisuddin Khan, Amir Akramin Shafie, Md Masum Billah and Syed Masrur Ahmmad	
Motion Investigation of a Snake Robot with Different Scale Geometry and Coefficient of Friction	
Reprinted from: <i>Robotics</i> 2018 , 7, 18, doi:10.3390/robotics7020018	173
Hermes Giberti, Francesco La Mura, Gabriele Resmini and Marco Parmeggiani	
Fully Mechatronical Design of an HIL System for Floating Devices	
Reprinted from: <i>Robotics</i> 2018 , 7, 39, doi:10.3390/robotics7030039	185
Bozun Wang, Yefei Si, Charul Chadha, James T. Allison and Albert E. Patterson	
Nominal Stiffness of GT-2 Rubber-Fiberglass Timing Belts for Dynamic System Modeling and Design	
Reprinted from: <i>Robotics</i> 2018 , 7, 75, doi:10.3390/robotics7040075	215
Author Index	223

About the Editor

Raffaele Di Gregorio is currently Full Professor of Machine Mechanics at the Engineering Department of the University of Ferrara, Italy. He received an M.Sc. degree in nuclear engineering, an M.Sc. degree in mechanical engineering and an M.Sc. degree in automotive engineering from the Polytechnic University of Turin in 1982, 1985 and 1988, respectively, and a Ph.D. degree in applied mechanics from the University of Bologna, Italy, in 1992. In 1983, he received a grant of the Italian Automotive Technical Association (ATA) for spending one year as a research fellow at the FIAT Research Center, Orbassano (Italy). Between 1984 and 1992, he served as officer of the Technical Corp in the Italian Army; first, at the Military School of Turin, and successively, at the STAVECO (Military Vehicle Factory) of Bologna. In 1993, he taught at the ITIS O. Belluzzi of Bologna, Italy. Since 1994, he has been part of the Engineering Department of the University of Ferrara, Italy. His research interests include kinematics and dynamics of mechanisms and machines, biomechanics, robotics, vibration mechanics and vehicle mechanics. He has been author and co-author of about 80 technical papers published in refereed international journals, and as many papers published in conference proceedings. He has been “observer/member” of the IFToMM Technical Committee “Computational Kinematics” since 2007, and “member” of the IFToMM Permanent Commission “Standardization of Terminology” since 2009. Moreover, he is an ASME member and served as “general member” of the ASME Mechanisms and Robotics Committee between 2007 and 2012; since 2013, he has continued to serve this Committee as a past member. In these roles, he has actively participated in the organization of International Conferences on Mechanisms and/or Robotics and served as chair of the ASME—Mechanisms & Robotics’ Honors and Awards Sub-Committee from 2014 to 2016. As well as being editorial board member of many international journals, he serves as associate editor of “ACTAPRESS International Journal of Robotics and Automation”, and “Mechanism and Machine Theory”, and served as associate editor of the “ASME Journal of Mechanisms and Robotics” from July 2014 to October 2020. Furthermore, he has been “Guest Editor” of the Special Issue series “Kinematics and Robot Design” published by the journal “MDPI Robotics” since 2018 and “Editor in Chief” of the “Journal of Mechanical Engineering and Automation” since 2012. An updated list of his publication together with more details can be found on the websites:

- UNIFE: <http://docente.unife.it/raffaele.digregorio-en>
- ResearchGate: <http://www.researchgate.net/profile/Raffaele.Di.Gregorio>
- Scopus Author ID: <https://www.scopus.com/authid/detail.uri?authorId=7003465861>
- ORCID: <https://orcid.org/0000-0003-3925-3016>
- ResearchID: <http://www.researcherid.com/rid/F-9316-2014>
- Google Scholar: <http://scholar.google.it/citations?user=n.MI9FEAAAAJ&hl=it>
- A list of his professional links can be found at http://www.linkedin.com/profile/view?id=128075898&trk=tab_pro

Preface to “Kinematics and Robot Design I, KaRD2018”

Kinematics is intimately related to nearly all the design aspects of robotic/automatic systems. Even though the motion description is an ancient science, which received the name “kinematics” by Ampère [1] in 1834, its study never stopped [2] and is still an alive field of research whose results have direct effects on the design of many everyday devices. Topics such as analysis and synthesis of mechanisms, robot modeling and simulation, robot control, mobility and singularity analysis, performance measures, accuracy analysis, path planning and obstacle avoidance, collaborative robotics, novel manipulator architectures, metamorphic mechanisms, compliant mechanism analysis and synthesis, micro/nano-manipulator design, origami-based robotics, medical and rehabilitation robotics, bioinspired robotics, etc., deal with kinematics. All these topics have a deep social impact and somehow delineate future perspectives of human welfare, which attract big economic interest. Therefore, the presence of many serial conferences and publications devoted to mechanism kinematics and its applications, which involve a numerous scientific community, is no surprise. The Special Issue series on “Kinematics and Robot Design” (KaRD series) is one of these serial publications.

The KaRD series is hosted by the open access journal “MDPI Robotics” and aims at creating an open environment where researchers can present their works and discuss all the topics focused on the many aspects that involve kinematics in the design of robotic/automatic systems by using also supplementary multimedia materials uploaded during the submission. Even though the KaRD series publishes one Special Issue per year, all the received papers are peer-reviewed as soon as they are submitted and, if accepted, they are immediately published on MDPI Robotics and appear on the website of the KaRD issue. The open access nature of this series allows the authors to easily share their papers and the accompanying supplementary materials with the reference scientific community.

“Kinematics and Robot Design I, KaRD2018” (https://www.mdpi.com/journal/robotics/special_issues/KARD) is the first issue of the KaRD series. It received 22 papers and, after the peer-review process, accepted only 14 papers. The accepted papers cover some theoretical and many design/applicative aspects. This volume collects 14 such accepted papers and is organized as follows. The first paper [3] deals with the use of dual quaternion in spacecraft robotics; whereas, the successive two [4,5] deal with self-motion in PKMs and spherical PKMs, respectively. Then, the fourth paper [6] deals with the modeling of link flexibility in manipulators, and the fifth [7] with balancing. The sixth [8] and the seventh [9] address specific design issues, and the next three [10–12] deal with performance evaluation and optimal design of manipulators. Eventually, the last four papers [13–16] deal with particular applicative issues.

1. Ampère, A-M. 1834. *Essai sur la philosophie des sciences, or Exposition analytique d’une classification naturelle de toutes les connaissances humaines*. Bachelier, Paris.
2. Macagno, E. 1991. *History of kinematics: Inception of modern kinematics*. Iowa Institute of Hydraulic Research (IIHR), Monograph No. 112.
3. Valverde, A.; Tsiotras, P. Spacecraft Robot Kinematics Using Dual Quaternions. *Robotics* **2018**, *7*, 64, doi: 10.3390/robotics7040064.
4. Nawratil, G. Hexapods with Plane-Symmetric Self-Motions. *Robotics* **2018**, *7*, 27, doi:10.3390/robotics7020027.

5. Legnani, G.; Fassi, I. Kinematics Analysis of a Class of Spherical PKMs by Projective Angles. *Robotics* **2018**, *7*, 59, doi:10.3390/robotics7040059.
6. Giorgio, I.; Del Vecovo, D. Non-Linear Lumped-Parameter Modeling of Planar Multi-Link Manipulators with Highly Flexible Arms. *Robotics* **2018**, *7*, 60, doi:10.3390/robotics7040060.
7. Ciupitu, L. Adaptive Balancing of Robots and Mechatronic Systems. *Robotics* **2018**, *7*, 68, doi:10.3390/robotics7040068.
8. Bin Junaid, A.; Diaz De Cerio Sanchez, A.; Betancor Bosch, J.; Vitzilaios, N.; Zweiri, Y. Design and Implementation of a Dual-Axis Tilting Quadcopter. *Robotics* **2018**, *7*, 65, doi:10.3390/robotics7040065.
9. Palpacelli, M.; Carbonari, L.; Palmieri, G.; Callegari, M. Design of a Lockable Spherical Joint for a Reconfigurable 3-URU Parallel Platform. *Robotics* **2018**, *7*, 42, doi:10.3390/robotics7030042.
10. Di Gregorio, R.; Cattai, M.; Simas, H. Performance-Based Design of the CRS-RRC Schoenflies-Motion Generator. *Robotics* **2018**, *7*, 55, doi:10.3390/robotics7030055.
11. Boschetti, G.; Trevisani, A. Cable Robot Performance Evaluation by Wrench Exertion Capability. *Robotics* **2018**, *7*, 15, doi:10.3390/robotics7020015.
12. Liu, G. Optimal Kinematic Design of a 6-UCU Kind Gough-Stewart Platform with a Guaranteed Given Accuracy. *Robotics* **2018**, *7*, 30, doi:10.3390/robotics7020030.
13. Faroni, M.; Beschi, M.; Pedrocchi, N.; Visioli, A. Viability and Feasibility of Constrained Kinematic Control of Manipulators. *Robotics* **2018**, *7*, 41, doi:10.3390/robotics7030041.
14. Huq, N.; Khan, M.; Shafie, A.; Billah, M.; Ahmmad, S. Motion Investigation of a Snake Robot with Different Scale Geometry and Coefficient of Friction. *Robotics* **2018**, *7*, 18, doi:10.3390/robotics7020018.
15. Giberti, H.; La Mura, F.; Resmini, G.; Parmeggiani, M. Fully Mechatronical Design of an HIL System for Floating Devices. *Robotics* **2018**, *7*, 39, doi:10.3390/robotics7030039.
16. Wang, B.; Si, Y.; Chadha, C.; Allison, J.; Patterson, A. Nominal Stiffness of GT-2 Rubber-Fiberglass Timing Belts for Dynamic System Modeling and Design. *Robotics* **2018**, *7*, 75, doi:10.3390/robotics7040075.

Raffaele Di Gregorio

Editor

Article

Spacecraft Robot Kinematics Using Dual Quaternions

Alfredo Valverde ^{*,†} and Panagiotis Tsiotras [†]

School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30313, USA; tsiotras@gatech.edu

* Correspondence: avalverde3@gatech.edu

† These authors contributed equally to this work.

Received: 30 July 2018; Accepted: 10 October 2018; Published: 12 October 2018

Abstract: In recent years, there has been a growing interest in servicing orbiting satellites. In most cases, in-orbit servicing relies on the use of spacecraft-mounted robotic manipulators to carry out complicated mission objectives. Dual quaternions, a mathematical tool to conveniently represent pose, has recently been adopted within the space industry to tackle complex control problems during the stages of proximity operations and rendezvous, as well as for the dynamic modeling of robotic arms mounted on a spacecraft. The objective of this paper is to bridge the gap in the use of dual quaternions that exists between the fields of spacecraft control and fixed-base robotic manipulation. In particular, we will cast commonly used tools in the field of robotics as dual quaternion expressions, such as the Denavit-Hartenberg parameterization, or the product of exponentials formula. Additionally, we provide, via examples, a study of the kinematics of different serial manipulator configurations, building up to the case of a completely free-floating robotic system. We provide expressions for the dual velocities of the different types of joints that commonly arise in industrial robots, and we end by providing a collection of results that cast convex constraints commonly encountered by space robots during proximity operations in terms of dual quaternions.

Keywords: spacecraft; robotics; dual quaternions; kinematics

1. Introduction

Robots are increasingly present in our daily lives, with their many uses ranging from simple vacuuming devices to complex manufacturing robotic arms. This growth is sustained by the continuous development of faster and better software and hardware, as well as strong theoretical advances in the areas of kinematics, dynamics, computer vision, sensing, etc. The space industry, owing to obvious reasons having to do with the unfriendliness of the space environment to humans, relies heavily on the use of robotics systems. In fact, interplanetary robotic exploration is at the core of NASA's Jet Propulsion Laboratory, and a wide variety of companies and governmental agencies are currently developing space-rated robotic manipulation systems for on-orbit satellite servicing [1,2].

The field of robotics is a well-established one. Lately, in the field of fixed-base robotics, progress in the area of kinematics and dynamics has mainly focused on ease of use, and speed and performance improvements [3,4]. The combination between the study of robots and their use in space, i.e., space robotics, must find common ground between the techniques used in both. For example, quaternions are the representation of choice when it comes to attitude parameterization for spacecraft control and estimation, while $SE(2)/SE(3)$ and the Spatial Vector Algebra [5] are the dominant tools of choice in the fixed-base robotic community. Therefore, with the recent advent of dual quaternions, it is only natural to explore the use of a pose (i.e., position and attitude) representation tool for spacecraft control and estimation in order to study robotic systems mounted on a spacecraft.

Dual quaternion algebra is an extension of the well-known quaternion algebra. The former is used to study rigid body pose while the latter is used extensively to study just the attitude of a rigid body. Dual quaternions have recently seen a proliferation in their use for spacecraft

control [6–9]. Several factors have contributed to the recent interest in dual quaternions in spacecraft control. First, the similarities between quaternion and dual quaternion-based spacecraft controllers and estimators [10] make dual quaternions an appealing tool for the practitioner who is familiar with the (standard) quaternion algebra. Next, dual quaternions naturally encode position information, thus avoiding the artificial separation of rotational and translational motion during control, which becomes essential during proximity operations or robotic servicing missions. More recently, dual quaternions have been used extensively for the dynamic modeling of ground-based robotic manipulators, providing an even stronger argument towards their use for dynamic modeling of spacecraft-mounted robotic manipulators [11].

While dual quaternions have been used for serial robot kinematic design [12–14], and for kinematic manipulation of points, vectors, lines, screws and planes [15], few references incorporate velocity information into their study of kinematics. Leclercq et al. [16] studied robot kinematics in the context of human motion using dual quaternions, yielding one of the most complete references to study kinematic chains with dual quaternions. More recently, Quiroz-Omaña and Adorno [17] have made use of dual quaternions in the context of robotic manipulation on a non-holonomic base. The methodologies exhibited in [16,17], however, do not take advantage of the well-known and convenient dual quaternion expression for kinematics, which could avoid manually taking time derivatives of pose expressions. A possible reason for this is the lack of a systematic manner to represent the combined linear and angular velocities of joints in dual algebra, and instead relying on the explicit derivative of pose-like expressions.

Works in the fields of dynamics and spacecraft control have settled on an understanding of the construction of dual velocities [18,19], which can be extended to provide generic expressions for the dual velocities of rigid bodies, or even of the different types of joints that may appear in a serial kinematic chain. In fact, Özgür and Mezouar [20] make use of said representation of dual velocity, commonly given by an expression of the form $\omega = \omega + \epsilon v$, to perform kinematic control on a robotic arm, yielding a clever representation of the Jacobian matrix that uses dual quaternion screws. Their approach, however, has a fixed base and requires the use of base-frame coordinates—as opposed to body-frame coordinates, which are commonly used in the study of spacecraft motion—to describe the Plücker lines associated to the different joints of the system.

Given the significant interest that dual quaternions have garnered in the last decade in the realm of space applications, it is pertinent to contribute to the literature a straightforward treatment of kinematics with an emphasis on space-based robotic operations. In this paper, we aim to extend the study of robot kinematics using dual quaternions, mainly by lifting the condition that the robotic base must be fixed, allowing it instead to move freely in the three-dimensional space. Additionally, we consider the possibility of incorporating different types of joints, and provide the formulas for the dual velocity of each different type of joint. Along the way, we provide some important well-known results, such as the derivation of the famous quaternion kinematic law, and the aforementioned dual quaternion equivalent, as well as a collection of results that capture convex constraints using dual quaternions. While the latter expressions have been used in the field of Entry, Descent, and Landing (EDL), their incorporation in robotic manipulation for in-orbit servicing missions is also extremely beneficial in order to ensure safety and robustness.

This paper is structured as follows: Section 2 provides the mathematical tools necessary to use quaternion and dual quaternion algebras. Section 3 provides an overview of the most common kinematic tools in the robotics fields in dual quaternion form. Section 4 provides the development of the kinematic equations of motion using dual quaternions, and in Section 5 we provide a brief summary of some important constraint expressions for robotic manipulation cast using dual quaternions. Such constraints arise naturally in many in-orbit servicing missions. Addressing these constraints in a numerically efficient manner (e.g., casting them as convex constraints) leads to safe and elegant solutions of the in-orbit servicing problem.

2. Mathematical Preliminaries

In this section we give an introduction to quaternion and dual quaternion algebras, which provide convenient mathematical frameworks for attitude and pose representations respectively. Next, we provide the theoretical foundations required to study the kinematics of rigid bodies, and in particular how they pertain to serial manipulators.

2.1. Quaternions

The group of quaternions, as defined by Hamilton in 1843, extends the well-known imaginary unit j , which satisfies $j^2 = -1$. This non-abelian group is defined by $\mathbb{Q}_8 \triangleq \{-1, i, j, k : i^2 = j^2 = k^2 = ijk = -1\}$. The algebra constructed from \mathbb{Q}_8 over the field of real numbers is the quaternion algebra defined as $\mathbb{H} \triangleq \{q = q_0 + q_1i + q_2j + q_3k : i^2 = j^2 = k^2 = ijk = -1, q_0, q_1, q_2, q_3 \in \mathbb{R}\}$. This defines an associative, non-commutative, division algebra.

In practice, quaternions are often referred to by their scalar and vectors parts as $q = (q_0, \bar{q})$, where $q_0 \in \mathbb{R}$ and $\bar{q} = [q_1, q_2, q_3]^T \in \mathbb{R}^3$. The properties of the quaternion algebra are summarized in Table 1. Filipe and Tsiotras [7] also conveniently define a multiplication between real 4-by-4 matrices and quaternions, denoted by the $*$ operator, which resembles the well-known matrix-vector multiplication by simply representing the quaternion coefficients as a vector in \mathbb{R}^4 . In other words, given $a = (a_0, \bar{a}) \in \mathbb{H}$ and a matrix $M \in \mathbb{R}^{4 \times 4}$ defined as

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \quad (1)$$

where $M_{11} \in \mathbb{R}, M_{12} \in \mathbb{R}^{1 \times 3}, M_{21} \in \mathbb{R}^{3 \times 1}$ and $M_{22} \in \mathbb{R}^{3 \times 3}$, then

$$M * a \triangleq (M_{11}a_0 + M_{12}\bar{a}, M_{21}a_0 + M_{22}\bar{a}) \in \mathbb{H}. \quad (2)$$

Table 1. Quaternion Operations.

Operation	Definition
Addition	$a + b = (a_0 + b_0, \bar{a} + \bar{b})$
Scalar multiplication	$\lambda a = (\lambda a_0, \lambda \bar{a})$
Multiplication	$ab = (a_0b_0 - \bar{a} \cdot \bar{b}, a_0\bar{b} + b_0\bar{a} + \bar{a} \times \bar{b})$
Conjugate	$a^* = (a_0, -\bar{a})$
Dot product	$a \cdot b = (a_0b_0 + \bar{a} \cdot \bar{b}, 0_{3 \times 1}) = \frac{1}{2}(a^*b + b^*a)$
Cross product	$a \times b = (0, a_0\bar{b} + b_0\bar{a} + \bar{a} \times \bar{b}) = \frac{1}{2}(ab - b^*a^*)$
Norm	$\ a\ = \sqrt{a \cdot a}$
Scalar part	$\text{sc}(a) = (a_0, 0_{3 \times 1})$
Vector part	$\text{vec}(a) = (0, \bar{a})$

Since any rotation can be described by three parameters, the unit norm constraint is imposed on quaternions for attitude representation. *Unit* quaternions are closed under multiplication, but not under addition. A quaternion describing the orientation of frame X with respect to frame Y, denoted by $q_{X/Y}$, satisfies $q_{X/Y}^*q_{X/Y} = q_{X/Y}q_{X/Y}^* = 1$, where $1 \triangleq (1, \bar{0}_{3 \times 1})$. This quaternion can be constructed as $q_{X/Y} = (\cos(\phi/2), \bar{n} \sin(\theta/2))$, where \bar{n} and θ are the *unit* Euler axis, and Euler angle of the rotation respectively. It is worth emphasizing that $q_{Y/X}^* = q_{X/Y}$, and that $q_{X/Y}$ and $-q_{X/Y}$ represent the same rotation. Furthermore, given quaternions $q_{Y/X}$ and $q_{Z/Y}$, the quaternion describing the rotation from X to Z is given by $q_{Z/X} = q_{Y/X}q_{Z/Y}$. For completeness purposes, we define $0 \triangleq (0, \bar{0}_{3 \times 1})$.

Three-dimensional vectors can also be interpreted as special cases of quaternions. Specifically, given $\bar{s}^X \in \mathbb{R}^3$, the coordinates of a vector expressed in frame X, its quaternion representation is given

by $s^x = (0, \bar{s}^x) \in \mathbb{H}^v$, where \mathbb{H}^v is the set of *vector* quaternions defined as $\mathbb{H}^v \triangleq \{(q_0, \bar{q}) \in \mathbb{H} : q_0 = 0\}$ (see Reference [19] for further information). The change of the reference frame for a vector quaternion is achieved by the adjoint operation, and is given by $s^y = q_{V/X}^* s^x q_{V/X}$. Additionally, given $s \in \mathbb{H}^v$, we can define the operation $[\cdot]^\times : \mathbb{H}^v \rightarrow \mathbb{R}^{4 \times 4}$ as

$$[s]^\times = \begin{bmatrix} 0 & 0_{1 \times 3} \\ 0_{3 \times 1} & [\bar{s}]^\times \end{bmatrix}, \quad \text{where} \quad [\bar{s}]^\times = \begin{bmatrix} 0 & -s_3 & s_2 \\ s_3 & 0 & -s_1 \\ -s_2 & s_1 & 0 \end{bmatrix}. \quad (3)$$

For quaternions $a = (a_0, \bar{a})$ and $b = (b_0, \bar{b})$, the left and right quaternion multiplication operators $[\cdot]_L, [\cdot]_R : \mathbb{H} \rightarrow \mathbb{R}^{4 \times 4}$ will be defined as

$$[a]_L * b \triangleq [b]_R * a \triangleq ab, \quad (4)$$

where

$$[a]_L = \begin{bmatrix} a_0 & -a_1 & -a_2 & -a_3 \\ \hline a_1 & a_0 & -a_3 & a_2 \\ a_2 & a_3 & a_0 & -a_1 \\ a_3 & -a_2 & a_1 & a_0 \end{bmatrix} = \begin{bmatrix} a_0 & -\bar{a}^r \\ \bar{a} & a_0 \mathbb{I}_3 + [\bar{a}]^\times \end{bmatrix}, \quad (5)$$

$$[b]_R = \begin{bmatrix} b_0 & -b_1 & -b_2 & -b_3 \\ \hline b_1 & b_0 & b_3 & -b_2 \\ b_2 & -b_3 & b_0 & b_1 \\ b_3 & b_2 & -b_1 & b_0 \end{bmatrix} = \begin{bmatrix} b_0 & -\bar{b}^r \\ \bar{b} & b_0 \mathbb{I}_3 - [\bar{b}]^\times \end{bmatrix}. \quad (6)$$

2.2. Dual Quaternions

We define the dual quaternion group as

$$\mathbb{Q}_d := \{-1, i, j, k, \epsilon, \epsilon i, \epsilon j, \epsilon k : i^2 = j^2 = k^2 = ijk = -1, \epsilon i = i\epsilon, \epsilon j = j\epsilon, \epsilon k = k\epsilon, \epsilon \neq 0, \epsilon^2 = 0\}. \quad (7)$$

The dual quaternion algebra arises as the algebra of the dual quaternion group \mathbb{Q}_d over the field of real numbers, and is denoted as \mathbb{H}_d . When dealing with the modeling of mechanical systems, it is convenient to present this algebra as $\mathbb{H}_d = \{q = q_r + \epsilon q_d : q_r, q_d \in \mathbb{H}\}$, where ϵ is the dual unit. We call q_r the real part, and q_d the dual part of the dual quaternion q .

Filipe and Tsiotras [7,8,19,21] have laid out much of the groundwork in terms of the notation and basic properties of dual quaternions for spacecraft problems. The main properties of the dual quaternion algebra are listed in Table 2. Filipe and Tsiotras [7] also conveniently define a multiplication between matrices and dual quaternions, denoted by the $*$ operator, that resembles the well-known real matrix-vector multiplication by simply representing the dual quaternion coefficients as a vector in \mathbb{R}^8 . In other words, given $a = a_r + \epsilon a_d \in \mathbb{H}_d$ and a matrix $M \in \mathbb{R}^{8 \times 8}$ defined as

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \quad (8)$$

where $M_{11}, M_{12}, M_{21}, M_{22} \in \mathbb{R}^{4 \times 4}$, then

$$M \star a \triangleq (M_{11} \star a_r + M_{12} \star a_d) + \epsilon(M_{21} \star a_r + M_{22} \star a_d) \in \mathbb{H}_d. \tag{9}$$

Table 2. Dual Quaternion Operations.

Operation	Definition
Addition	$\mathbf{a} + \mathbf{b} = (a_r + b_r) + \epsilon(a_d + b_d)$
Scalar multiplication	$\lambda \mathbf{a} = (\lambda a_r) + \epsilon(\lambda a_d)$
Multiplication	$\mathbf{a}\mathbf{b} = (a_r b_r) + \epsilon(a_d b_r + a_r b_d)$
Conjugate	$\mathbf{a}^* = (a_r^*) + \epsilon(a_d^*)$
Dot product	$\mathbf{a} \cdot \mathbf{b} = (a_r \cdot b_r) + \epsilon(a_d \cdot b_r + a_r \cdot b_d) = \frac{1}{2}(\mathbf{a}^* \mathbf{b} + \mathbf{b}^* \mathbf{a})$
Cross product	$\mathbf{a} \times \mathbf{b} = (a_r \times b_r) + \epsilon(a_d \times b_r + a_r \times b_d) = \frac{1}{2}(\mathbf{a}\mathbf{b} - \mathbf{b}^* \mathbf{a}^*)$
Circle product	$\mathbf{a} \circ \mathbf{b} = (a_r \cdot b_r + a_d \cdot b_d) + \epsilon 0$
Swap	$\mathbf{a}^s = a_d + \epsilon a_r$
Norm	$\ \mathbf{a}\ = \sqrt{\mathbf{a} \circ \mathbf{a}}$
Scalar part	$\text{sc}(\mathbf{a}) = \text{sc}(a_r) + \epsilon \text{sc}(a_d)$
Vector part	$\text{vec}(\mathbf{a}) = \text{vec}(a_r) + \epsilon \text{vec}(a_d)$

Analogous to the set of vector quaternions \mathbb{H}^v , we can define the set of vector dual quaternions as $\mathbb{H}_d^v \triangleq \{\mathbf{q} = q_r + \epsilon q_d : q_r, q_d \in \mathbb{H}^v\}$. For vector dual quaternions we will define the skew-symmetric operator $[\cdot]^\times : \mathbb{H}_d^v \rightarrow \mathbb{R}^{8 \times 8}$,

$$[\mathbf{s}]^\times = \begin{bmatrix} [s_r]^\times & 0_{4 \times 4} \\ [s_d]^\times & [s_r]^\times \end{bmatrix}. \tag{10}$$

For dual quaternions $\mathbf{a} = a_r + \epsilon a_d$ and $\mathbf{b} = b_r + \epsilon b_d \in \mathbb{H}_d$, the left and right dual quaternion multiplication operators $[\cdot]_L, [\cdot]_R : \mathbb{H}_d \rightarrow \mathbb{R}^{8 \times 8}$ are defined as

$$\mathbf{a}\mathbf{b} \triangleq [\mathbf{a}]_L \star \mathbf{b} \triangleq [\mathbf{b}]_R \star \mathbf{a}, \tag{11}$$

where

$$[\mathbf{a}]_L = \begin{bmatrix} [a_r]_L & 0_{4 \times 4} \\ [a_d]_L & [a_r]_L \end{bmatrix} \quad \text{and} \quad [\mathbf{b}]_R = \begin{bmatrix} [b_r]_R & 0_{4 \times 4} \\ [b_d]_R & [b_r]_R \end{bmatrix}. \tag{12}$$

Since rigid body motion has six degrees of freedom, a dual quaternion needs two constraints to parameterize it. The dual quaternion describing the relative pose of frame B relative to frame I is given by $\mathbf{q}_{B/I} = q_{B/I,r} + \epsilon q_{B/I,d} = q_{B/I} + \epsilon \frac{1}{2} q_{B/I}^B r_{B/I}^B$ where $r_{B/I}^B$ is the position quaternion describing the location of the origin of frame B relative to that of frame I, expressed in B-frame coordinates. It can be easily observed that $q_{B/I,r} \cdot q_{B/I,r} = 1$ and $q_{B/I,r} \cdot q_{B/I,d} = 0$, where $0 = (0, \vec{0})$, providing the two necessary constraints. Thus, a dual quaternion representing a pose transformation is a *unit* dual quaternion, since it satisfies $\mathbf{q} \cdot \mathbf{q} = \mathbf{q}^* \mathbf{q} = \mathbf{1}$, where $\mathbf{1} \triangleq 1 + \epsilon 0$. Additionally, we also define $\mathbf{0} \triangleq 0 + \epsilon 0$.

Similar to the standard quaternion relationships, the frame transformations laid out in Table 3 can be easily verified.

Table 3. Unit Dual Quaternion Operations.

Composition of transformations	$\mathbf{q}_{Z/X} = \mathbf{q}_{Y/X} \mathbf{q}_{Z/Y}$
Inverse, Conjugate	$\mathbf{q}_{Y/X}^* = \mathbf{q}_{X/Y}$

In Reference [19] it was proven that for a dual unit quaternion $\mathbf{q} \in \mathbb{H}_d$, \mathbf{q} and $-\mathbf{q}$ represent the same frame transformation, property inherited from the space of quaternions. Therefore, as is done in practice for quaternions, dual quaternions can be subjected to properization, which is the action

of redefining a dual quaternion so that the scalar part of the quaternion is always positive. Formally, we can define the properization of a dual quaternion $q = q_r + \epsilon q_d$ as

$$q := -q \quad \text{if} \quad (q_r)_0 < 0, \tag{13}$$

where $(q_r)_0$ is the scalar part of q_r . Just like in the case of quaternions, dual quaternions also inherit the so-called *unwinding phenomenon*, first described in [22], which is most important in control applications.

A useful equation is the generalization of the velocity of a rigid body in dual form, which contains both the linear and angular velocity components. The dual velocity of the Y-frame with respect to the Z-frame, expressed in X-frame coordinates, is defined as

$$\omega_{Y/Z}^x = q_{X/Y}^* \omega_{Y/Z}^y q_{X/Y} = \omega_{Y/Z}^x + \epsilon(v_{Y/Z}^x + \omega_{Y/Z}^x \times r_{X/Y}^x), \tag{14}$$

where $\omega_{Y/Z}^x = (0, \bar{\omega}_{Y/Z}^x)$ and $v_{Y/Z}^x = (0, \bar{v}_{Y/Z}^x)$, $\bar{\omega}_{Y/Z}^x$ and $\bar{v}_{Y/Z}^x \in \mathbb{R}^3$ are respectively the angular and linear velocity of the Y-frame with respect to the Z-frame expressed in X-frame coordinates, and $r_{X/Y}^x = (0, \bar{r}_{X/Y}^x)$, where $\bar{r}_{X/Y}^x \in \mathbb{R}^3$ is the position vector from the origin of the Y-frame to the origin of the X-frame expressed in X-frame coordinates. In particular, from Equation (14) we observe that the dual velocity of a rigid body assigned to frame B with respect to an inertial frame I, expressed in B-frame coordinates is given as $\omega_{B/I}^b = \omega_{B/I}^b + \epsilon v_{B/I}^b$. However, if we wanted to express this same dual velocity in inertial frame coordinates, as per Equation (14) we would get $\omega_{B/I}^b = \omega_{B/I}^b + \epsilon(v_{B/I}^b + \omega_{B/I}^b \times r_{I/B}^b)$. We will formally introduce frame transformations next.

2.3. Frame Transformations Using Dual Quaternions

As is common in the study of kinematics, frame transformations are vital for the determination of velocities and accelerations with respect to different frames. A dual velocity, or dual acceleration, can be described by a dual vector quaternion $s^x \in \mathbb{H}_d^v$ expressed in X-frame coordinates as $s^x \triangleq s_r^x + \epsilon s_d^x$, where $s_r^x, s_d^x \in \mathbb{H}^v$. As noted for Equation (14), frame transformations are given by the adjoint operation as

$$\begin{aligned} s^y &= q_{Y/X}^* s^x q_{Y/X} \\ &= (q_{Y/X} + \epsilon \frac{1}{2} r_{Y/X}^x q_{Y/X})^* (s_r^x + \epsilon s_d^x) (q_{Y/X} + \epsilon \frac{1}{2} r_{Y/X}^x q_{Y/X}) \\ &= (q_{Y/X}^* + \epsilon \frac{1}{2} q_{Y/X}^* r_{Y/X}^{x*}) (s_r^x + \epsilon s_d^x) (q_{Y/X} + \epsilon \frac{1}{2} r_{Y/X}^x q_{Y/X}) \\ &= (q_{Y/X}^* - \epsilon \frac{1}{2} q_{Y/X}^* r_{Y/X}^x) (s_r^x + \epsilon s_d^x) (q_{Y/X} + \epsilon \frac{1}{2} r_{Y/X}^x q_{Y/X}) \\ &= (q_{Y/X}^* - \epsilon \frac{1}{2} q_{Y/X}^* r_{Y/X}^x) (s_r^x q_{Y/X} + \epsilon (s_d^x q_{Y/X} + s_r^x \frac{1}{2} r_{Y/X}^x q_{Y/X})) \\ &= q_{Y/X}^* s_r^x q_{Y/X} - \epsilon (\frac{1}{2} q_{Y/X}^* r_{Y/X}^x s_r^x q_{Y/X}) + \epsilon (q_{Y/X}^* s_d^x q_{Y/X} + q_{Y/X}^* s_r^x \frac{1}{2} r_{Y/X}^x q_{Y/X}) \\ &= s_r^y + \epsilon (s_d^y + \frac{1}{2} q_{Y/X}^* s_r^x q_{Y/X} q_{Y/X}^* r_{Y/X}^x q_{Y/X} - \frac{1}{2} q_{Y/X}^* r_{Y/X}^x q_{Y/X} q_{Y/X}^* s_r^x q_{Y/X}) \\ &= s_r^y + \epsilon (s_d^y + \frac{1}{2} s_r^y r_{Y/X}^y - \frac{1}{2} r_{Y/X}^y s_r^y) \\ &= s_r^y + \epsilon (s_d^y + \frac{1}{2} s_r^y r_{Y/X}^y - \frac{1}{2} (r_{Y/X}^y)^* (s_r^y)^*). \end{aligned}$$

By the definition of the cross product of two quaternion quantities given in Table 1, we get that

$$\begin{aligned} s^y &= q_{Y/X}^* s^x q_{Y/X} \\ &= s_r^y + \epsilon (s_d^y + s_r^y \times r_{Y/X}^y) \\ &= s_r^y + \epsilon (s_d^y + r_{X/Y}^y \times s_r^y). \end{aligned} \tag{15}$$

Analogously, the transformation of a dual vector $s^y \triangleq s_r^y + \epsilon s_d^y$ can be easily derived using the procedure described above to be:

$$\begin{aligned}
 \mathbf{s}^x &= \mathbf{q}_{Y/X} \mathbf{s}^y \mathbf{q}_{Y/X}^* \\
 &= \mathbf{s}_r^x + \epsilon (\mathbf{s}_d^x + \mathbf{s}_r^x \times \mathbf{r}_{X/Y}^x) \\
 &= \mathbf{s}_r^x + \epsilon (\mathbf{s}_d^x + \mathbf{r}_{Y/X}^x \times \mathbf{s}_r^x).
 \end{aligned} \tag{16}$$

As is standard notation, we can define the group adjoint operation for *unit* dual quaternions as

$$\text{Ad}_q \mathbf{s} \triangleq \mathbf{q} \mathbf{s} \mathbf{q}^{-1} = \mathbf{q} \mathbf{s} \mathbf{q}^* \tag{17}$$

Therefore, using this notation, the frame transformations derived above can be cast as

$$\mathbf{s}^x = \text{Ad}_{\mathbf{q}_{Y/X}} \mathbf{s}^y \tag{18}$$

$$\mathbf{s}^y = \text{Ad}_{\mathbf{q}_{Y/X}^*} \mathbf{s}^x = \text{Ad}_{\mathbf{q}_{X/Y}} \mathbf{s}^x \tag{19}$$

The power of dual quaternions goes beyond the ability to represent pose and transform dual velocities and accelerations. In fact, dual quaternions can natively—without constructs that fall outside the algebra—encode the most typical geometric objects such as points, lines and planes. The reader is referred to the literature to find such parameterizations and the correct dual quaternion transformation [15,16].

2.4. Derivation of Fundamental Kinematic Laws

In this section we will derive both the quaternion and dual quaternion kinematic laws. We will make the time dependence explicit only when necessary for clarity.

The three-dimensional attitude kinematics evolve as

$$\dot{\mathbf{q}}_{X/Y} = \frac{1}{2} \mathbf{q}_{X/Y} \boldsymbol{\omega}_{X/Y}^x = \frac{1}{2} \boldsymbol{\omega}_{X/Y}^y \mathbf{q}_{X/Y} \tag{20}$$

where $\boldsymbol{\omega}_{X/Y}^z \triangleq (0, \overline{\boldsymbol{\omega}}_{X/Y}^z) \in \mathbb{H}^v$ and $\overline{\boldsymbol{\omega}}_{X/Y}^z \in \mathbb{R}^3$ is the angular velocity of frame X with respect to frame Y expressed in Z-frame coordinates. On the other hand, the dual quaternion kinematics can be expressed as [7]

$$\dot{\mathbf{q}}_{X/Y} = \frac{1}{2} \mathbf{q}_{X/Y} \boldsymbol{\omega}_{X/Y}^x = \frac{1}{2} \boldsymbol{\omega}_{X/Y}^y \mathbf{q}_{X/Y} \tag{21}$$

Lemma 1. *The attitude of a rigid body evolves as $\dot{\mathbf{q}}_{X/Y} = \frac{1}{2} \mathbf{q}_{X/Y} \boldsymbol{\omega}_{X/Y}^x$, as stated in Equation (20).*

Proof. Denote the infinitesimal rotation about axis \hat{u} by $\Delta\theta$. The quaternion that represents this rotation is constructed as $\delta \mathbf{q}_{X/Y}(\Delta t) \triangleq (\cos(\Delta\theta/2), \hat{u} \sin(\Delta\theta/2))$. Therefore, $\mathbf{q}_{X/Y}(t + \Delta t) = \mathbf{q}_{X/Y}(t) \delta \mathbf{q}_{X/Y}(\Delta t)$. Then, for a small rotation angle, $\delta \mathbf{q}_{X/Y}(\Delta t) = (1, \hat{u} \Delta\theta/2)$. Substituting into the previous expression for $\mathbf{q}_{X/Y}(t + \Delta t)$, we obtain

$$\begin{aligned}
 \mathbf{q}_{X/Y}(t + \Delta t) &= \mathbf{q}_{X/Y}(t) (1, \hat{u} \Delta\theta/2) \\
 &= \mathbf{q}_{X/Y}(t) (1 + (0, \hat{u} \Delta\theta/2)) \\
 &= \mathbf{q}_{X/Y}(t) + \frac{1}{2} \mathbf{q}_{X/Y}(t) \hat{u} \Delta\theta.
 \end{aligned} \tag{22}$$

Manipulating the expression and dividing by Δt , we obtain

$$\frac{\mathbf{q}_{X/Y}(t + \Delta t) - \mathbf{q}_{X/Y}(t)}{\Delta t} = \frac{1}{2} \mathbf{q}_{X/Y}(t) \hat{u} \frac{\Delta\theta}{\Delta t} \tag{23}$$

and invoking the limit as $\Delta t \rightarrow 0$ yields

$$\dot{q}_{x/y}(t) = \frac{1}{2}q_{x/y}(t)\omega_{x/y}^x(t), \quad (24)$$

where we have defined the angular velocity as $\omega_{x/y}^x \triangleq \hat{u}\dot{\theta}$, i.e., the rate of rotation about the instantaneous Euler axis. \square

Lemma 2. *The pose of a rigid body evolves as $\dot{q}_{x/y} = \frac{1}{2}q_{x/y}\omega_{x/y}^x$, as stated in Equation (21).*

Proof. Taking the derivative of $q_{x/y} = q_{x/y} + \epsilon \frac{1}{2}q_{x/y}r_{x/y}^x$ we get

$$\begin{aligned} \dot{q}_{x/y} &= \dot{q}_{x/y} + \epsilon \left(\frac{1}{2}\dot{q}_{x/y}r_{x/y}^x + \frac{1}{2}q_{x/y}\dot{r}_{x/y}^x \right) \\ &= \frac{1}{2}q_{x/y}\omega_{x/y}^x + \epsilon \left(\frac{1}{4}q_{x/y}\omega_{x/y}^x r_{x/y}^x + \frac{1}{2}q_{x/y}\dot{r}_{x/y}^x \right) \\ &= \frac{1}{2}q_{x/y}\omega_{x/y}^x + \epsilon \left(\frac{1}{4}q_{x/y}\omega_{x/y}^x r_{x/y}^x + \frac{1}{2}q_{x/y}(\dot{v}_{x/y}^x - \omega_{x/y}^x \times r_{x/y}^x) \right) \\ &= \frac{1}{2}q_{x/y}\omega_{x/y}^x + \epsilon \left(\frac{1}{2}q_{x/y}\dot{v}_{x/y}^x + \frac{1}{4}q_{x/y}\omega_{x/y}^x r_{x/y}^x - \frac{1}{2}q_{x/y}(\omega_{x/y}^x \times r_{x/y}^x) \right), \end{aligned} \quad (25)$$

where we used the fact that $\dot{v}_{x/y}^x \triangleq dr_{x/y}^x/dt = \dot{r}_{x/y}^x + \omega_{x/y}^x \times r_{x/y}^x$. Using the definition of the cross product, we know that $\omega_{x/y}^x \times r_{x/y}^x = \frac{1}{2}(\omega_{x/y}^x r_{x/y}^x - (r_{x/y}^x)^*(\omega_{x/y}^x)^*) = \frac{1}{2}(\omega_{x/y}^x r_{x/y}^x - r_{x/y}^x \omega_{x/y}^x)$. Evaluating this cross product into the above expression yields

$$\begin{aligned} &= \frac{1}{2}q_{x/y}\omega_{x/y}^x + \epsilon \left(\frac{1}{2}q_{x/y}\dot{v}_{x/y}^x + \frac{1}{4}q_{x/y}\omega_{x/y}^x r_{x/y}^x - \frac{1}{4}q_{x/y}(\omega_{x/y}^x r_{x/y}^x - r_{x/y}^x \omega_{x/y}^x) \right) \\ &= \frac{1}{2}q_{x/y}\omega_{x/y}^x + \epsilon \frac{1}{2}q_{x/y}\dot{v}_{x/y}^x + \epsilon \frac{1}{4}q_{x/y}r_{x/y}^x \omega_{x/y}^x \\ &= \frac{1}{2}q_{x/y}\omega_{x/y}^x + \epsilon \frac{1}{2}q_{x/y}\dot{v}_{x/y}^x + \epsilon \frac{1}{4}q_{x/y}r_{x/y}^x \omega_{x/y}^x + \epsilon^2 \frac{1}{4}q_{x/y}r_{x/y}^x \dot{v}_{x/y}^x, \text{ since } \epsilon^2 = 0 \\ &= \frac{1}{2}q_{x/y}(\omega_{x/y}^x + \epsilon \dot{v}_{x/y}^x) + \frac{1}{2}\epsilon \frac{1}{2}q_{x/y}r_{x/y}^x (\omega_{x/y}^x + \epsilon \dot{v}_{x/y}^x) \\ &= \frac{1}{2}(q_{x/y} + \epsilon \frac{1}{2}q_{x/y}r_{x/y}^x)(\omega_{x/y}^x + \epsilon \dot{v}_{x/y}^x) \\ &= \frac{1}{2}q_{x/y}\omega_{x/y}^x, \end{aligned} \quad (26)$$

proving the desired result. \square

Remark 1. *The spatial kinematic equation $\dot{q}_{x/y} = \frac{1}{2}\omega_{x/y}^y q_{x/y}$ can be immediately derived as a direct consequence of the adjoint transformation equation $\omega_{x/y}^x = q_{x/y}^* \omega_{x/y}^y q_{x/y}$, which implies $q_{x/y}\omega_{x/y}^x = \omega_{x/y}^y q_{x/y}$.*

Remark 2. *The spatial kinematic equation $\dot{q}_{x/y} = \frac{1}{2}\omega_{x/y}^x q_{x/y}$ can be immediately derived as a direct consequence of the adjoint transformation equation $\omega_{x/y}^x = q_{x/y}^* \omega_{x/y}^y q_{x/y}$, which implies $q_{x/y}\omega_{x/y}^x = \omega_{x/y}^y q_{x/y}$.*

3. Robot Kinematics Using Dual Quaternions

3.1. Dual Quaternion Notation

The forward kinematics of a robot can be easily laid out in dual quaternion form. In general, a dual quaternion encoding the relationship between two frames A and B is given as

$$q_{B/A} = q_{B/A} + \epsilon \frac{1}{2}q_{B/A}r_{B/A}^B \quad (27)$$

$$q_{B/A} = q_{B/A} + \epsilon \frac{1}{2}r_{B/A}^A q_{B/A} \quad (28)$$

where $q_{B/A}$ is the quaternion that represents the attitude change in going from reference frame A, to reference frame B. The position vectors $r_{B/A}^B$ and $r_{B/A}^A$ represent the position vector from the origin of frame A to the origin of frame B expressed in frame B, and frame A coordinates, respectively. Notice that Equations (27) and (28) can be equivalently expressed as follows:

$$\text{Rotation First: } \mathbf{q}_{B/A} = (q_{B/A} + \epsilon 0)(1 + \epsilon \frac{1}{2} r_{B/A}^B), \tag{29}$$

$$\text{Translation First: } \mathbf{q}_{B/A} = (1 + \epsilon \frac{1}{2} r_{B/A}^A)(q_{B/A} + \epsilon 0), \tag{30}$$

leading to an intuitive decomposition of the underlying operations. In the forward kinematics, Equation (29) implies that the frame rotation is carried out first, and then a translation is carried out relative to the new frame. Equation (30) denotes a translation in the base frame, followed by an attitude change of the resulting frame. Throughout this work we will use the *translation first* approach.

3.2. Product of Exponentials Formula in Dual-Quaternion Form

The product of exponentials formula has been long used to study the forward kinematics of robots. Reference [23] has a thorough introduction to the topic, with many examples. In this section we lay out the main results that cast the product of exponentials (POE) formula in dual quaternion form. In particular, [20] has made use of the dual quaternion formalism to perform geometric control on a fixed-base robotic arm, where the forward kinematics of the robot are expressed using the POE formula.

As commonly used in robotics, the exponential operation takes an element of the Lie algebra for a given Lie group, and renders a group element. For the dual quaternion case, let the set of parameters $(\theta, s) \in D \times \mathbb{H}_d^v$, where $D = \{a + \epsilon a_d : a, a_d \in \mathbb{R} \text{ and } \epsilon^2 = 0\}$ is the set of dual numbers, parametrize a screw motion as shown in Figure 1. In particular, θ and s are given by

$$\theta = \theta + \epsilon d, \quad \theta \in D, \quad \theta, d \in \mathbb{R}, \tag{31}$$

$$s = \ell + \epsilon m, \quad s \in \mathbb{H}_d^v, \quad \ell, m \in \mathbb{H}^v, \tag{32}$$

where θ is the angle of the screw motion, d is the translation along the screw axis, ℓ is the unit screw axis of the joint, and m is the moment vector of the screw axis of direction ℓ with respect to the origin of the local inertial frame. This implies that

$$m = r_{P/I} \times \ell, \tag{33}$$

where the point P lies on the screw axis. In robotic systems, the exponential mapping is commonly used to evaluate the forward kinematics of fixed-base robotic systems. We summarize the dual quaternion exponential mapping in the following lemma [14,20].

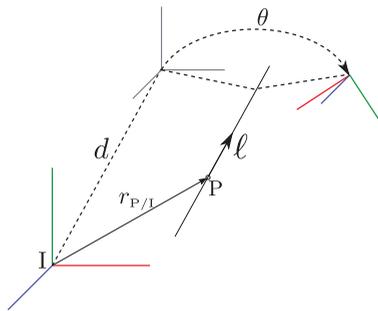


Figure 1. Screw motion parametrized by θ and s .

Lemma 3. The exponential operation, $\exp : \mathbb{D} \times \mathbb{H}_d^v \rightarrow \mathbb{H}_d$ for a given pair $(\theta, \mathbf{s}) \in \mathbb{D} \times \mathbb{H}_d^v$ defined as in Equations (31) and (32) is given as

$$\begin{aligned} \mathbf{q} &= \exp\left(\frac{1}{2}\theta\mathbf{s}\right), \quad \mathbf{q} \in \mathbb{H}_d \\ &= \cos\left(\frac{1}{2}\theta\right) + \mathbf{s} \sin\left(\frac{1}{2}\theta\right) \\ &= \left(\cos\left(\frac{1}{2}\theta\right), \ell \sin\left(\frac{1}{2}\theta\right)\right) + \epsilon \left(-\frac{1}{2}d \sin\left(\frac{1}{2}\theta\right), \frac{1}{2}d\ell \cos\left(\frac{1}{2}\theta\right) + m \sin\left(\frac{1}{2}\theta\right)\right). \end{aligned} \quad (34)$$

Proof. Since $\theta = \theta + \epsilon d \in \mathbb{D}$, we have that

$$\cos\left(\frac{1}{2}\theta\right) = \cos\left(\frac{1}{2}\theta\right) + \epsilon \frac{d}{2} \left(-\sin\left(\frac{1}{2}\theta\right)\right) \quad (35)$$

$$\sin\left(\frac{1}{2}\theta\right) = \sin\left(\frac{1}{2}\theta\right) + \epsilon \frac{d}{2} \cos\left(\frac{1}{2}\theta\right). \quad (36)$$

It follows that

$$\mathbf{q} = \cos\left(\frac{1}{2}\theta\right) + \mathbf{s} \sin\left(\frac{1}{2}\theta\right) \quad (37)$$

$$= \cos\left(\frac{1}{2}\theta\right) - \epsilon \frac{d}{2} \sin\left(\frac{1}{2}\theta\right) + (\ell + \epsilon m) \left(\sin\left(\frac{1}{2}\theta\right) + \epsilon \frac{d}{2} \cos\left(\frac{1}{2}\theta\right)\right), \quad (38)$$

which yields the desired result upon expansion. \square

Remark 3. By comparing Equations (28) and (34), it can be deduced that the effect of a joint motion can be characterized by an equivalent rotation and a translation. In particular, by equating the real parts of the dual quaternions, we have that

$$q_{B/A} = \left(\cos\left(\frac{1}{2}\theta\right), \ell \sin\left(\frac{1}{2}\theta\right)\right), \quad (39)$$

and from the dual parts

$$\frac{1}{2}r_{B/A}^\Lambda q_{B/A} = \left(-\frac{1}{2}d \sin\left(\frac{1}{2}\theta\right), \frac{1}{2}d\ell \cos\left(\frac{1}{2}\theta\right) + m \sin\left(\frac{1}{2}\theta\right)\right). \quad (40)$$

Equivalently, $r_{B/A}^\Lambda$ can be described as

$$r_{B/A}^\Lambda = (0, d\ell + m \sin(\theta) + (\cos(\theta) - 1)m \times \ell). \quad (41)$$

The inverse to the exponential mapping is the logarithmic mapping, $\ln : \mathbb{H}_d \rightarrow \mathbb{H}_d$, which is defined as

$$\ln \mathbf{q} = \frac{1}{2}\theta\mathbf{s} = \frac{1}{2}\theta\ell + \epsilon \frac{1}{2}(\theta m + d\ell). \quad (42)$$

Appendix A.6. of [20] explains how to retrieve $\{\theta, d, \ell, m\}$ given a dual quaternion, \mathbf{q} .

Given the dual quaternion from the inertial (base) frame to the end effector, at the robots' home configuration, $\mathbf{q}_{e,0/1'}$ and parameter s_i for each of the n joints of a robot at its home configuration, the product of exponentials formula yields

$$\mathbf{q}_{e/1} = \exp\left(\frac{1}{2}\theta_1\mathbf{s}_1\right) \dots \exp\left(\frac{1}{2}\theta_n\mathbf{s}_n\right) \mathbf{q}_{e,0/1'} \quad (43)$$

where joint 1 is closest to the base and joint n is closest to the end-effector. The exponential formula is effectively changing the spatial frame, as opposed to the body frame of the end-effector. Besides its simplicity to compute forward kinematics, the POE formula is straightforward to compute for a given configuration once the type of joint is known and the geometric properties of the robot are selected.

As this point, it is worth emphasizing that in regards to the moving frames used in space operations, the use of an inertial frame with respect to which one can perform spatial kinematics,

as was done in [20], is impractical. Since the satellite base is constantly in motion, local-frame parameterizations of pose transformations across the links of the manipulator are preferred. Therefore, we favor the use of the forward-moving pose representations to express the location of the end-effector frame. We show next how to use the Denavit-Hartenberg parameterization in dual quaternions to capture such a transformation.

3.3. Denavit-Hartenberg Parameters in Dual Quaternion Form

The Denavit-Hartenberg parameters, commonly referred to as DH parameters, are four geometric quantities that allow identifying the relative pose of a joint with respect to another in a systematic manner. We will denote a set of DH parameters as $\{d_i, \theta_i, a_i, \alpha_i\}$ for joint i . The parameters d_i and θ_i are commonly referred to as *joint* parameters, while a_i and α_i are known as the *link* parameters. A complete description of the DH parameters for R and P joint types, and several examples of their use are provided in [24]. In [24] a thorough description of the orientation of the frames is also provided, to which the reader is referred. In [25], Gan et al. have used dual quaternions in combination with the DH parameter convention to capture the pose transformation between joints. For completeness, we provide these equations herein, making use of Figure 2.

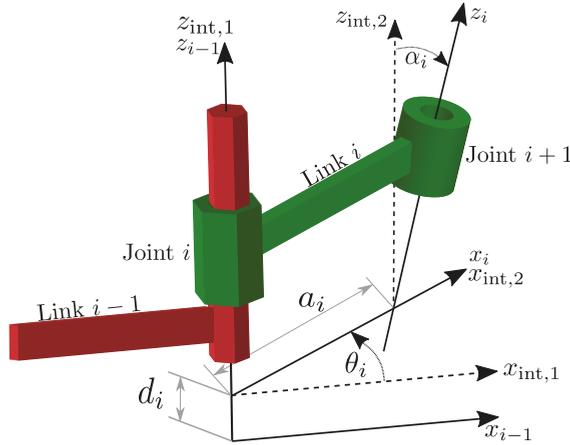


Figure 2. Denavit-Hartenberg parameters.

In words, the transformation from the reference frame assigned to the proximal joint (i.e., closer to the base of the robot) of a given link i , to the reference frame assigned to its distal joint (i.e., closer to the end effector), is described in terms of the DH parameters as:

1. From the origin O_{i-1} , displace along the Z_{i-1} (joint) axis by an amount d_i . Define this intermediate frame as $\{int, 1\}$.
2. Rotate about the Z_{i-1} axis by θ_i until axis X_{i-1} is superimposed to X_i
3. Translate along X_i by a distance of a_i . Define this intermediate frame as $\{int, 2\}$.
4. Rotate about the X_i axis by α_i

Mathematically, we can write this as the composition of four elementary dual quaternion operations, and summarize it further into two composite dual quaternions as

$$q_{i/i-1} = (1 + \epsilon r_{int,1/i-1}^{int,1})(q_{int,2/int,1} + \epsilon 0)(1 + \epsilon r_{int,2/int,1}^{int,2})(q_{i/int,2} + \epsilon 0) \quad (44)$$

$$= (q_{int,2/int,1} + \epsilon r_{int,1/i-1}^{int,1} q_{int,2/int,1})(q_{i/int,2} + \epsilon r_{int,2/int,1}^{int,2} q_{i/int,2}) \quad (45)$$

where

$$r_{\text{int},1/\text{i},1}^{\text{int},1} = (0, [0, 0, d_i]^T) \quad (46)$$

$$q_{\text{int},2/\text{int},1} = (\cos \theta_i/2, [0, 0, \sin \theta_i/2]^T) \quad (47)$$

$$r_{\text{int},2/\text{int},1}^{\text{int},2} = (0, [a_i, 0, 0]^T) \quad (48)$$

$$q_{\text{i}/\text{int},2} = (\cos \alpha_i/2, [\sin \alpha_i/2, 0, 0]^T) \quad (49)$$

Notice that while this is compact and readable up to multiplication of the dual quaternions, the same cannot be said about the end result compared to its homogeneous transformation matrix (HTM) counterpart. In fact, if we express $q_{\text{i}/\text{i}-1}$ component-wise, and cast it as a vector in \mathbb{R}^8 which is the typical representation of dual quaternions for numerical purposes, and compute the equivalent HTM, we get the following:

$$q_{\text{i}/\text{i}-1} = \begin{bmatrix} \cos(\alpha/2) \cos(\theta/2) \\ \sin(\alpha/2) \cos(\theta/2) \\ \sin(\alpha/2) \sin(\theta/2) \\ \cos(\alpha/2) \sin(\theta/2) \\ -\frac{1}{2}a_i \sin(\alpha_i/2) \cos(\theta_i/2) - \frac{1}{2}d_i \cos(\alpha_i/2) \sin(\theta_i/2) \\ \frac{1}{2}a_i \cos(\alpha_i/2) \cos(\theta_i/2) - \frac{1}{2}d_i \sin(\alpha_i/2) \sin(\theta_i/2) \\ \frac{1}{2}a_i \cos(\alpha_i/2) \sin(\theta_i/2) + \frac{1}{2}d_i \sin(\alpha_i/2) \cos(\theta_i/2) \\ \frac{1}{2}d_i \cos(\alpha_i/2) \cos(\theta_i/2) - \frac{1}{2}a_i \sin(\alpha_i/2) \sin(\theta_i/2) \end{bmatrix} \quad (50)$$

$$T_{\text{i}/\text{i}-1} = \left[\begin{array}{ccc|c} \cos \theta_i & \sin \theta_i & 0 & -a_i \\ -\cos \alpha_i \sin \theta_i & \cos \alpha_i \cos \theta_i & \sin \alpha_i & -d_i \sin \alpha_i \\ \sin \alpha_i \sin \theta_i & -\sin \alpha_i \cos \theta_i & \cos \alpha_i & -d_i \cos \alpha_i \\ \hline 0 & 0 & 0 & 1 \end{array} \right]. \quad (51)$$

While the HTM is more readable and faster to code, it uses 16 doubles and a multi-dimensional array to store the information and operate in the underlying algebra.

Remark 4. Since the transformations associated to θ_i and d_i are about z_{i-1} and the operations associated to α_i and a_i happen about x_i , both stages of the DH transformation can be interpreted in the context of screw theory. Hence, the operation described by Equation (44) can be equivalently expressed as the composition of exponential operations given by

$$q_{\text{i}/\text{i}-1} = \exp(\frac{1}{2}\theta_1 s_1) \exp(\frac{1}{2}\theta_2 s_2), \quad (52)$$

where $\theta_1 = \theta_i + \epsilon d_i$ and $s_1 = (0, [0, 0, 1]^T) + \epsilon 0$ and $\theta_2 = \alpha_i + \epsilon a_i$ and $s_2 = (0, [1, 0, 0]^T) + \epsilon 0$.

4. Manipulator Kinematics Using Dual Quaternions

In this section we provide examples to demonstrate how one can develop the kinematic equations for different types of serial manipulators using dual quaternions.

4.1. Example: Forward Kinematics with an Inertially Fixed Base

The serial RR configuration in Figure 3 will be used as an example of how to use dual quaternions for forward kinematics. Notice that the pose of the end effector with respect to the inertial frame is given by

$$q_{e/1} = q_{1/1} q_{2/1} q_{e/2}. \quad (53)$$

For the sake of exposition, these are given by

$$q_{1/i} = (1 + \epsilon \frac{1}{2} r_{1/i}^1)(q_{1/i} + \epsilon 0), \tag{54}$$

$$q_{2/i} = (1 + \epsilon \frac{1}{2} r_{2/i}^1)(q_{2/i} + \epsilon 0), \tag{55}$$

$$q_{e/2} = (1 + \epsilon \frac{1}{2} r_{e/2}^2)(q_{e/2} + \epsilon 0), \tag{56}$$

where the *translation-first* approach has been used. Each of these quantities can be easily determined from the geometry of the problem. The position quaternions are given by $r_{x/Y}^y = (0, \bar{r}_{x/Y}^y)$, and

$$\bar{r}_{1/i}^1 = [0, 0, 0]^T, \tag{57}$$

$$\bar{r}_{2/i}^1 = [l_1, 0, 0]^T, \tag{58}$$

$$\bar{r}_{e/2}^2 = [l_2, 0, 0]^T, \tag{59}$$

while the quaternions are given by

$$q_{1/i} = (\cos \alpha_1/2, [0, 0, \sin \alpha_1/2]^T), \tag{60}$$

$$q_{2/i} = (\cos \alpha_2/2, [0, 0, \sin \alpha_2/2]^T), \tag{61}$$

$$q_{e/2} = 1. \tag{62}$$

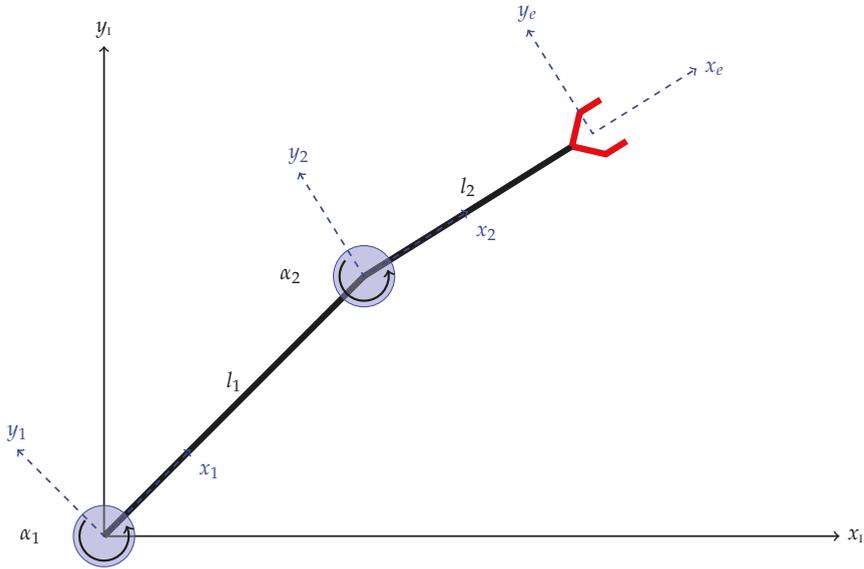


Figure 3. Robot arm configuration.

The time derivative of the dual quaternion yields information about the angular and linear velocity of the end-effector. In particular, we have that for a dual quaternion:

$$\dot{q}_{x/y} = \frac{1}{2} q_{x/y} \omega_{x/y}^x = \frac{1}{2} \omega_{x/y}^y q_{x/y} \quad (63)$$

where the first equality in Equation (63) is associated with a body-frame time derivative, and the second equality in Equation (63) is associated with a spatial-frame time derivative.

With these definitions in mind, we compute the time-rate of change of the pose of the end-effector as

$$\dot{q}_{e/1} = \dot{q}_{1/1} q_{2/1} q_{e/2} + q_{1/1} \dot{q}_{2/1} q_{e/2} + q_{1/1} q_{2/1} \dot{q}_{e/2} \quad (64)$$

$$= \frac{1}{2} q_{1/1} \omega_{1/1}^1 q_{2/1} q_{e/2} + q_{1/1} \frac{1}{2} q_{2/1} \omega_{2/1}^2 q_{e/2} + q_{1/1} q_{2/1} \frac{1}{2} q_{e/2} \omega_{e/2}^e. \quad (65)$$

Then, using Equation (63), we get that the dual velocity of the end effector with respect to the inertial frame is given by

$$\begin{aligned} \omega_{e/1}^e &= 2q_{e/1}^* \dot{q}_{e/1} \\ &= q_{e/1}^* q_{1/1} \omega_{1/1}^1 q_{2/1} q_{e/2} + q_{e/1}^* q_{1/1} q_{2/1} \omega_{2/1}^2 q_{e/2} + q_{e/1}^* q_{1/1} q_{2/1} q_{e/2} \underbrace{\omega_{e/2}^e}_{=0} \\ &= q_{e/2}^* q_{2/1}^* q_{1/1}^* q_{1/1} \omega_{1/1}^1 q_{2/1} q_{e/2} + q_{e/2}^* q_{2/1}^* q_{1/1}^* q_{1/1} q_{2/1} \omega_{2/1}^2 q_{e/2} \\ &= q_{e/2}^* q_{2/1}^* \omega_{1/1}^1 q_{2/1} q_{e/2} + q_{e/2}^* \omega_{2/1}^2 q_{e/2} \\ &= \text{Ad}_{q_{e/2}^* q_{2/1}^*} \omega_{1/1}^1 + \text{Ad}_{q_{e/2}^*} \omega_{2/1}^2 \\ &= \text{Ad}_{(q_{2/1} q_{e/2})^*} \omega_{1/1}^1 + \text{Ad}_{(q_{e/2})^*} \omega_{2/1}^2 \\ &= \begin{bmatrix} \text{Ad}_{(q_{2/1} q_{e/2})^*} \xi_{1/1}^1 & \text{Ad}_{(q_{e/2})^*} \xi_{2/1}^2 \end{bmatrix} \dot{\alpha} \quad (66) \end{aligned}$$

$$= J^B(q, \xi) \dot{\alpha}, \quad (67)$$

where $J^B(q, \xi)$ is the Jacobian expressed in the body frame and

$$\tilde{\alpha} = \begin{bmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \end{bmatrix} \quad \text{and} \quad \hat{\alpha} = \begin{bmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \end{bmatrix}. \quad (68)$$

The elements ξ_i are the dual quaternion screws for each of the joints. In general, the screws for revolute and prismatic joints are listed in Table 4 for each of the three axes, and these are independent of the current robot configuration.

Table 4. Screw (ξ_i) for revolute and prismatic joints.

	Revolute Joint	Prismatic Joint
X-axis	$(0, [1, 0, 0]^T) + \epsilon 0$	$0 + \epsilon(0, [1, 0, 0]^T)$
Y-axis	$(0, [0, 1, 0]^T) + \epsilon 0$	$0 + \epsilon(0, [0, 1, 0]^T)$
Z-axis	$(0, [0, 0, 1]^T) + \epsilon 0$	$0 + \epsilon(0, [0, 0, 1]^T)$

4.2. Example: Forward Kinematics of a Floating Double Pendulum with End-Effector

Given the floating double pendulum shown in Figure 4, we want to model its kinematics. The difference with respect to the one shown in Figure 3 is that the first revolute joint is free to translate in 2D space.

The kinematic equations of motion can thus be derived as follows using a geometric description of the forward kinematics

$$q_{e/1} = q_{1/1} q_{2/1} q_{e/2} \quad (69)$$

where $q_{1/1}$, $q_{2/1}$, $q_{e/2}$ are given by Equations (54)–(56). However, $p_{1/1} = [u, v, 0]^T$ determines the

translation of the first revolute joint in 2D. It is clear that

$$\frac{d}{dt} \bar{r}_{1/1}^1 = \dot{\bar{r}}_{1/1}^1 = \bar{v}_{1/1}^1 = [\dot{u}, \dot{v}, 0]^T \quad (70)$$

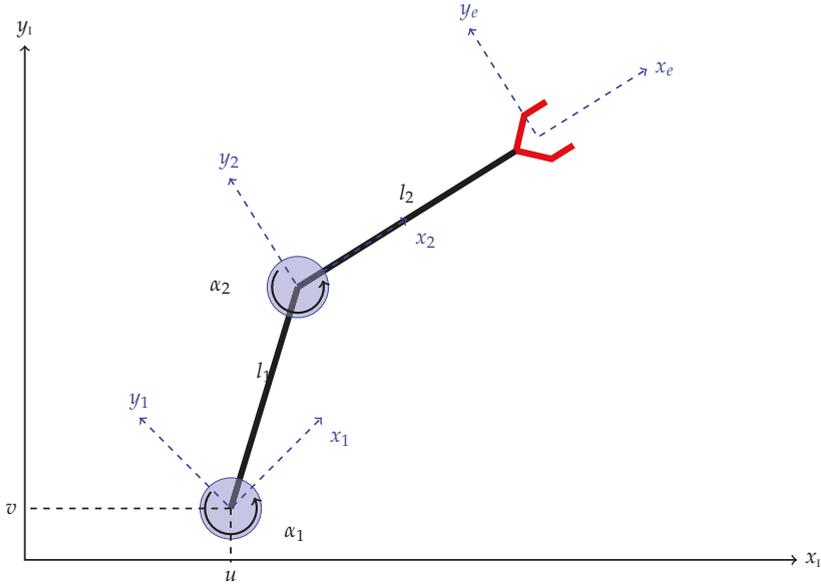


Figure 4. Robot arm configuration.

In this case, the time evolution of $q_{1/1}$ is given by

$$\dot{q}_{1/1} = \frac{1}{2} \omega_{1/1}^1 q_{1/1} \quad (71)$$

as before, but we redefine the dual velocity as dictated by the definition in Equation (14) as

$$\omega_{1/1}^1 = \omega_{1/1}^1 + \epsilon(v_{1/1}^1 - \omega_{1/1}^1 \times r_{1/1}^1). \quad (72)$$

The relationship derived earlier

$$\omega_{e/1}^e = \text{Ad}_{q_{e/2}^* q_{2/1}^*} \omega_{1/1}^1 + \text{Ad}_{q_{e/2}^*} \omega_{2/1}^2$$

still holds. However, $\omega_{1/1}^1$ must be computed from our knowledge of $\omega_{1/1}^1$. While in quaternion and vector notation this might be troublesome, the expression using dual quaternions is simple and given by

$$\omega_{1/1}^1 = q_{1/1}^* \omega_{1/1}^1 q_{1/1} = \text{Ad}_{q_{1/1}^*} \omega_{1/1}^1. \quad (73)$$

4.3. Manipulator on an Orbiting Spacecraft

For the general case, the robot base can move with six degrees of freedom, reinforcing the need for a convenient pose representation tool such as dual quaternions. Following an approach analogous to that proposed by Adorno in [26], in this section we will provide explicit expressions for the Jacobian matrix for different types of joints.

The kinematics of the robotic base, attached to frame B, would still be governed by the equation

$$\dot{q}_{B/I} = \frac{1}{2} q_{B/I} \omega_{B/I}^B \quad (74)$$

while the kinematics of the joints depend on the type of joint. The dual velocities for the joints depend on frame positioning and on the selection of the generalized coordinates. In Table 5, we provide example generalized coordinates and their corresponding dual velocities. Simple numerical derivatives can yield the dual acceleration of the joint. It is worth emphasizing that for a $3 - 2 - 1/\psi - \theta - \phi$ rotation, the matrix $M(\phi_{i/i-1}, \theta_{i/i-1}, \psi_{i/i-1})$ will correspond to

$$M(\phi_{i/i-1}, \theta_{i/i-1}, \psi_{i/i-1}) = \begin{bmatrix} 1 & 0 & -\sin(\theta_{i/i-1}) \\ 0 & \cos(\phi_{i/i-1}) & \cos(\theta_{i/i-1}) \sin(\phi_{i/i-1}) \\ 0 & -\sin(\phi_{i/i-1}) & \cos(\theta_{i/i-1}) \cos(\phi_{i/i-1}) \end{bmatrix}. \quad (75)$$

In general, we can identify the pose of a satellite-mounted end-effector by

$$q_{e/I} = q_{B/I} q_{O/B} q_{0/O} \left(\prod_{i=1}^{n-1} q_{i/i-1} \right) q_{e/n-1} \quad (76)$$

where $q_{O/B}$ represents the pose transformation from the body frame of the satellite, B, to the frame at the base of the satellite manipulator, denoted by O; frame i represents the joint frame attached to the i-th link, one of the n bodies composing the manipulator, at the location of the proximal joint; and e is the end-effector frame that is rigidly attached to the last link of the serial manipulator, $n - 1$. For clarity, an example of the product operator \prod used on dual quaternions is given by

$$\prod_{i=1}^2 q_{i/i-1} = q_{1/0} q_{2/1}. \quad (77)$$

Additionally, using the definition of frames described above, the first link of the manipulator is link 0 and its connecting joint to the satellite base is frame 0.

Then, since $q_{O/B}$ and $q_{e/n-1}$ are constant, the kinematics can be derived following the procedure of previous sections as

$$\begin{aligned} \dot{q}_{e/I} &= \dot{q}_{B/I} q_{O/B} q_{0/O} \left(\prod_{i=1}^{n-1} q_{i/i-1} \right) q_{e/n-1} + q_{B/I} q_{O/B} \dot{q}_{0/O} \left(\prod_{i=1}^{n-1} q_{i/i-1} \right) q_{e/n-1} \\ &\quad + q_{B/I} q_{O/B} q_{0/O} \sum_{k=1}^{n-1} (q_{k-1/0} \dot{q}_{k/k-1} q_{n-1/k}) q_{e/n-1} \\ &= \frac{1}{2} q_{B/I} \omega_{B/I}^B q_{O/B} q_{0/O} \left(\prod_{i=1}^{n-1} q_{i/i-1} \right) q_{e/n-1} + q_{B/I} q_{O/B} \frac{1}{2} q_{0/O} \omega_{0/O}^0 \left(\prod_{i=1}^{n-1} q_{i/i-1} \right) q_{e/n-1} \\ &\quad + q_{B/I} q_{O/B} q_{0/O} \sum_{k=1}^{n-1} (q_{k-1/0} \frac{1}{2} q_{k/k-1} \omega_{k/k-1}^k q_{n-1/k}) q_{e/n-1}. \end{aligned} \quad (78)$$

Multiplying by $2q_{e/I}^*$ on the left, we get

$$\begin{aligned} \omega_{e/I}^e &= q_{e/I}^* q_{B/I} \omega_{B/I}^B q_{O/B} q_{0/O} \left(\prod_{i=1}^{n-1} q_{i/i-1} \right) q_{e/n-1} + q_{e/I}^* q_{B/I} q_{O/B} q_{0/O} \omega_{0/O}^0 \left(\prod_{i=1}^{n-1} q_{i/i-1} \right) q_{e/n-1} \\ &\quad + q_{e/I}^* q_{B/I} q_{O/B} q_{0/O} \sum_{k=1}^{n-1} (q_{k-1/0} q_{k/k-1} \omega_{k/k-1}^k q_{n-1/k}) q_{e/n-1} \end{aligned} \quad (79)$$

and carrying out the dual quaternion multiplications to simplify the expression yields

$$\omega_{e/l}^e = q_{e/B}^* \omega_{B/l}^B q_{e/B} + q_{e/O}^* \omega_{O/O}^0 q_{e/O} + \sum_{k=1}^{n-1} q_{e/k}^* \omega_{k/k-1}^k q_{e/k}. \quad (80)$$

Table 5. Generalized coordinates and dual velocities for different joint types.

Joint Type	Generalized Coordinate Parametrization, $\tilde{\alpha}_i$	Dual Velocity, $\omega_{i/i-1}^i$
Revolute	$\theta_{i/i-1} \in \mathbb{R}^1$	$(0, [0, 0, \dot{\theta}_{i/i-1}]^T) + \epsilon 0$
Prismatic	$z_{i/i-1} \in \mathbb{R}^1$	$0 + \epsilon(0, [0, 0, \dot{z}_{i/i-1}]^T)$
Spherical	$[\phi_{i/i-1}, \theta_{i/i-1}, \psi_{i/i-1}]^T \in \mathbb{R}^3$	$(0, M(\phi_{i/i-1}, \theta_{i/i-1}, \psi_{i/i-1})[\dot{\phi}_{i/i-1}, \dot{\theta}_{i/i-1}, \dot{\psi}_{i/i-1}]^T) + \epsilon 0$
Cylindrical	$[\theta_{i/i-1}, z_{i/i-1}]^T \in \mathbb{R}^2$	$(0, [0, 0, \dot{\theta}_{i/i-1}]^T) + \epsilon(0, [0, 0, \dot{z}_{i/i-1}]^T)$
Cartesian	$[x_{i/i-1}, y_{i/i-1}, z_{i/i-1}]^T \in \mathbb{R}^3$	$0 + \epsilon(0, [\dot{x}_{i/i-1}, \dot{y}_{i/i-1}, \dot{z}_{i/i-1}]^T)$

In this form, it is straightforward to identify that in Equation (80), the first term yields the motion of the end-effector due to the motion of the base. The second and third terms provide the effect of the motion of the end-effector due to joint motion. We can now manipulate Equation (80) towards a more familiar structure

$$\omega_{e/l}^e = \llbracket q_{e/B}^* \rrbracket_L \llbracket q_{e/B} \rrbracket_R \star \omega_{B/l}^B + \llbracket q_{e/O}^* \rrbracket_L \llbracket q_{e/O} \rrbracket_R \star \omega_{O/O}^0 + \sum_{k=1}^{n-1} \llbracket q_{e/k}^* \rrbracket_L \llbracket q_{e/k} \rrbracket_R \star \omega_{k/k-1}^k. \quad (81)$$

Defining the vector of generalized coordinates as the vertical concatenation of the individual joint generalized coordinates, we can write

$$\begin{aligned} \omega_{e/l}^e &= \llbracket q_{e/B}^* \rrbracket_L \llbracket q_{e/B} \rrbracket_R \star \omega_{B/l}^B + \llbracket q_{e/O}^* \rrbracket_L \llbracket q_{e/O} \rrbracket_R \zeta_0 \dot{\alpha}_0 + \sum_{k=1}^{n-1} \llbracket q_{e/k}^* \rrbracket_L \llbracket q_{e/k} \rrbracket_R \zeta_k \dot{\alpha}_k \\ &= \llbracket q_{e/B}^* \rrbracket_L \llbracket q_{e/B} \rrbracket_R \star \omega_{B/l}^B + J(q, \zeta) \dot{\alpha}. \end{aligned} \quad (82)$$

Here, we have defined the body-frame Jacobian associated to joint motion as

$$J(q, \zeta) \triangleq \left[\llbracket q_{e/O}^* \rrbracket_L \llbracket q_{e/O} \rrbracket_R \zeta_0, \dots, \llbracket q_{e/k}^* \rrbracket_L \llbracket q_{e/k} \rrbracket_R \zeta_k, \dots, \llbracket q_{e/n-1}^* \rrbracket_L \llbracket q_{e/n-1} \rrbracket_R \zeta_{n-1} \right]. \quad (83)$$

The general term of the Jacobian mapping matrix, $\llbracket q_{e/k}^* \rrbracket_L \llbracket q_{e/k} \rrbracket_R \zeta_k$, where ζ_k is a screw matrix as defined in Table 6, is an improvement upon the more typical, adjoint-based methodology due to the ability of ζ_k to capture more than one degree of freedom in each of its different columns. For the case in which the adjoint formula $\text{Ad}_{q_{e/k}^*} \zeta_k$ is used, as in Equation (67), then ζ_k necessarily corresponds to one single generalized coordinate. In other words, the screws for the cylindrical, spherical and Cartesian joints would need to be separated into different columns, each of which has its adjoint operation applied independently.

For example, it would be easy to demonstrate that for a cylindrical ($d = 2$), spherical ($d = 3$) or Cartesian joints ($d = 3$),

$$\frac{\partial \omega_{i/i-1}^i}{\partial \dot{\alpha}_k} \equiv \llbracket q_{e/k}^* \rrbracket_L \llbracket q_{e/k} \rrbracket_R \zeta_k \in \mathbb{R}^{8 \times d}, \quad (84)$$

but

$$\frac{\partial \omega_{i/i-1}^i}{\partial \dot{\alpha}_k} \neq \text{Ad}_{q_{e/k}^*} \zeta_k \in \mathbb{H}_d, \quad (85)$$

for any physically intuitive $\zeta_k \in \mathbb{H}_d^q$.

Table 6. Screw matrix for different joint types.

Joint Type	Screw Matrix, $\zeta_i = \partial\omega_{i/i-1}^i/\partial\tilde{\mathbf{x}}_i$
Revolute	$[0, 0, 0, 1, 0, 0, 0, 0]^T$
Prismatic	$[0, 0, 0, 0, 0, 0, 0, 1]^T$
Spherical	$\begin{bmatrix} 0_{1 \times 3} \\ M(\phi_{i/i-1}, \theta_{i/i-1}, \psi_{i/i-1}) \\ 0_{4 \times 4} \end{bmatrix}^T$
Cylindrical	$\begin{bmatrix} 0, 0, 0, 1, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 0, 0, 1 \end{bmatrix}^T$
Cartesian	$\begin{bmatrix} 0_{5 \times 3} \\ \mathbb{I}_3 \end{bmatrix}$

5. Convex Constraints Using Dual Quaternions

When performing robotic operations, the incorporation of constraints is important where the safety of users, or a payload, is concerned. The dual quaternion framework is amenable to the incorporation of several convex constraints, which are of particular interest due to the availability of specialized codes to solve convex problems efficiently. The following presentation of convex constraints could be used in combination with a control approach such as the one proposed in [11], which is based on the differential dynamic programming algorithm.

In [27], the authors use dual quaternions as a pose parametrization representation to model convex state constraints for a powered landing scenario. In this section, we repurpose these same constraints for a space robotic servicing mission. The dual quaternion-based constraints will be provided without proof of convexity, since this is done in [27]. However, some properties of quaternions and some definitions are in order for a proper description of the results.

Lemma 4. *Given the quaternion $q \in \mathbb{H}$ and quaternions $r = (0, \bar{r}) \in \mathbb{H}^v$ and $y = (0, \bar{y}) \in \mathbb{H}^v$, the following equalities hold:*

$$(rq) \cdot (yq) = r \cdot y = (qr) \cdot (qy) \tag{86}$$

Proof. Using the definition of the quaternion dot product given in Table 1, the expression on the left becomes

$$\begin{aligned} (rq) \cdot (yq) &= \frac{1}{2} [(rq)^*yq + (yq)^*rq] \\ &= \frac{1}{2} [q^*r^*yq + q^*y^*rq] \\ &= \frac{1}{2}q^* [r^*y + y^*r]q \\ &= q^*(r \cdot y)q, \text{ and since } r \cdot y = (\bar{r} \cdot \bar{y}, 0_{3 \times 1}) = (\bar{r} \cdot \bar{y})\mathbf{1} \\ &= (\bar{r} \cdot \bar{y})q^*q \\ &= (\bar{r} \cdot \bar{y})\mathbf{1} \\ &= r \cdot y. \end{aligned} \tag{87}$$

The second equality can be proven in the same manner. \square

For the following facts, let us define

$$E_u \triangleq \begin{bmatrix} \mathbb{I}_4 & 0_{4 \times 4} \\ 0_{4 \times 4} & 0_{4 \times 4} \end{bmatrix} \quad (88)$$

and

$$E_d \triangleq \begin{bmatrix} 0_{4 \times 4} & 0_{4 \times 4} \\ 0_{4 \times 4} & \mathbb{I}_4 \end{bmatrix}. \quad (89)$$

Lemma 5. Consider the dual quaternion $\mathbf{q}_{B/A} = q_{B/A} + \epsilon \frac{1}{2} q_{B/A} r_{B/A}^B$. Then, $\mathbf{q}_{B/A} \circ \mathbf{q}_{B/A} = (1 + \frac{1}{4} \|r_{B/A}^B\|^2, 0_{3 \times 1}) + \epsilon 0$

Proof. By definition, $\mathbf{q}_{B/A} \circ \mathbf{q}_{B/A} = (q_{B/A} + \epsilon \frac{1}{2} q_{B/A} r_{B/A}^B) \circ (q_{B/A} + \epsilon \frac{1}{2} q_{B/A} r_{B/A}^B) = q_{B/A} \cdot q_{B/A} + (\frac{1}{2} q_{B/A} r_{B/A}^B) \cdot (\frac{1}{2} q_{B/A} r_{B/A}^B) + \epsilon 0$. By the unit norm constraint of the unit quaternions and applying Lemma 4 on the second summand, $\mathbf{q}_{B/A} \circ \mathbf{q}_{B/A} = (1 + \frac{1}{4} r_{B/A}^B \cdot r_{B/A}^B, 0_{3 \times 1}) + \epsilon 0$, from which the result follows. \square

Lemma 6. Consider the dual quaternion $\mathbf{q}_{B/A} = q_{B/A} + \epsilon \frac{1}{2} q_{B/A} r_{B/A}^B$. Then, $\mathbf{q}_{B/A} \circ (E_u \star \mathbf{q}_{B/A}) = \mathbf{1}$.

Proof. Using the definition of E_u , we have $\mathbf{q}_{B/A} \circ (E_u \star \mathbf{q}_{B/A}) = \mathbf{q}_{B/A} \circ (q_{B/A} + \epsilon 0) = q_{B/A} \cdot q_{B/A} + \epsilon 0$. The result follows from the unit constraint of a unit quaternion. \square

Lemma 7. Consider the dual quaternion $\mathbf{q}_{B/A} = q_{B/A} + \epsilon \frac{1}{2} q_{B/A} r_{B/A}^B$. Then, $\mathbf{q}_{B/A} \circ (E_d \star \mathbf{q}_{B/A}) = \frac{1}{4} \|r_{B/A}^B\|^2 + \epsilon 0$.

Proof. Using the definition of E_d , we have $\mathbf{q}_{B/A} \circ (E_d \star \mathbf{q}_{B/A}) = \mathbf{q}_{B/A} \circ (0 + \epsilon \frac{1}{2} q_{B/A} r_{B/A}^B) = (\frac{1}{2} q_{B/A} r_{B/A}^B) \cdot (\frac{1}{2} q_{B/A} r_{B/A}^B) + \epsilon 0$. The result follows from application of Lemma 4. \square

Lemma 8. Consider $\|r_{B/A}^B\| \leq \delta$. Then, $\mathbf{q}_{B/A} \circ \mathbf{q}_{B/A} \leq 1 + \frac{1}{4} \delta^2$.

Proof. From Lemma 5, it follows that $\mathbf{q}_{B/A} \circ \mathbf{q}_{B/A} = 1 + \frac{1}{4} \|r_{B/A}^B\|^2 \leq 1 + \frac{1}{4} \delta^2$. \square

Corollary 1. Given the bound $\|r_{B/A}^B\| \leq \delta$, it follows that $\mathbf{q}_{B/A} \circ \mathbf{q}_{B/A} \in [1, 1 + \frac{1}{4} \delta^2]$, which is a closed and bounded set.

It is worth emphasizing that in Lemmas 5 and 8 the bijective mapping between the circle product and the real-line is implied. In other words, since the circle product between two dual quaternions $a \circ b = s\mathbf{1}$ for some $s \in \mathbb{R}$, it will be commonly interpreted as $a \circ b = s$ for simplicity of exposition.

We are now ready to introduce three types of constraints in terms of dual quaternions:

1. Line-of-sight constraints.
2. Approach slope angle constraints, of which upper-and-lower bound constraints is a re-interpretation of the geometry.
3. Body attitude constraint with respect to an inertial direction.

For this, we will use notation consistent with [27]. Additionally, we require two auxiliary frames. We will define G as fixed on a gripper, and A as fixed on the target (say, an asteroid, or an object of interest) to be captured.

Proposition 1. Consider the domain $\mathcal{D} = \{\mathbf{q}_{G/A} \in \mathbb{H}_d : \mathbf{q}_{G/A} \circ \mathbf{q}_{G/A} \leq 1 + \frac{1}{4}\delta^2\}$. The line of sight constraint depicted in Figure 5 can be encoded as

$$r_{A/G}^G \cdot \hat{\mathbf{y}}^G \geq \|r_{A/G}^G\| \cos \theta, \quad (90)$$

and it requires that the angle between $r_{A/G}^G$ and $\hat{\mathbf{y}}^G$ remains less than θ . Using dual quaternions, this constraint can be equivalently expressed as

$$-\mathbf{q}_{G/A} \circ (M_H \star \mathbf{q}_{G/A}) + 2\|E_d \mathbf{q}_{G/A}\| \cos \theta \leq 0, \quad (91)$$

where

$$M_H = \begin{bmatrix} 0_{4 \times 4} & \llbracket \hat{\mathbf{y}}^G \rrbracket_{\mathbb{R}}^T \\ \llbracket \hat{\mathbf{y}}^G \rrbracket_{\mathbb{R}} & 0_{4 \times 4} \end{bmatrix}, \quad (92)$$

and it is convex over \mathcal{D} .

Proposition 2. Consider the domain $\mathcal{D} = \{\mathbf{q}_{G/A} \in \mathbb{H}_d : \mathbf{q}_{G/A} \circ \mathbf{q}_{G/A} \leq 1 + \frac{1}{4}\delta^2\}$. The approach slope constraint depicted in Figure 6, and the upper-and-lower bounded approach constraint depicted in Figure 7, can be encoded as

$$r_{G/A}^A \cdot \hat{\mathbf{z}}^A \geq \|r_{G/A}^A\| \cos \phi, \quad (93)$$

and it requires that the angle between $r_{G/A}^A$ and $\hat{\mathbf{z}}^A$ remains less than ϕ . Using dual quaternions, this constraint can be equivalently expressed as

$$-\mathbf{q}_{G/A} \circ (M_G \star \mathbf{q}_{G/A}) + 2\|E_d \mathbf{q}_{G/A}\| \cos \phi \leq 0, \quad (94)$$

where

$$M_G = \begin{bmatrix} 0_{4 \times 4} & \llbracket \hat{\mathbf{z}}^A \rrbracket_{\mathbb{L}}^T \\ \llbracket \hat{\mathbf{z}}^A \rrbracket_{\mathbb{L}} & 0_{4 \times 4} \end{bmatrix}, \quad (95)$$

and it is convex over \mathcal{D} .

Proposition 3. Consider the domain $\mathcal{D} = \{\mathbf{q}_{B/I} \in \mathbb{H}_d : \mathbf{q}_{B/I} \circ \mathbf{q}_{B/I} \leq 1 + \frac{1}{4}\delta^2\}$. The attitude constraint depicted in Figure 8 can be encoded as

$$\hat{\mathbf{n}}^I \cdot (\mathbf{q}_{B/I} \hat{\mathbf{n}}^B \mathbf{q}_{B/I}^*) \geq \cos \psi, \quad (96)$$

and it requires that the angle between the inertially fixed vector $\hat{\mathbf{n}}^I$ and the body fixed vector $\hat{\mathbf{n}}^B$ remains less than ψ . Using dual quaternions, this constraint can be equivalently expressed as

$$\mathbf{q}_{B/I} \circ (M_A \star \mathbf{q}_{B/I}) + \cos \psi \leq 0, \quad (97)$$

where

$$M_A = \begin{bmatrix} \llbracket \hat{\mathbf{z}}^I \rrbracket_{\mathbb{L}} & \llbracket \hat{\mathbf{z}}^B \rrbracket_{\mathbb{R}} & 0_{4 \times 4} \\ 0_{4 \times 4} & 0_{4 \times 4} & 0_{4 \times 4} \end{bmatrix}, \quad (98)$$

and it is convex over \mathcal{D} .

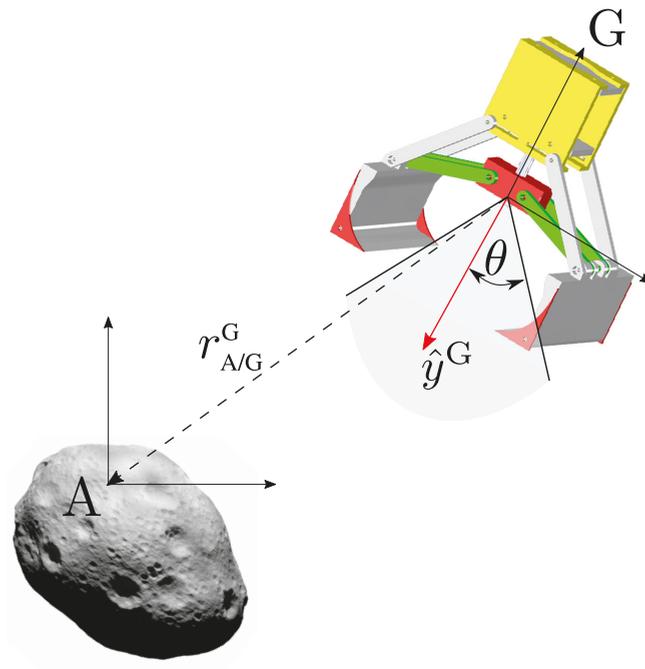


Figure 5. Line-of-sight constraint during grappling.

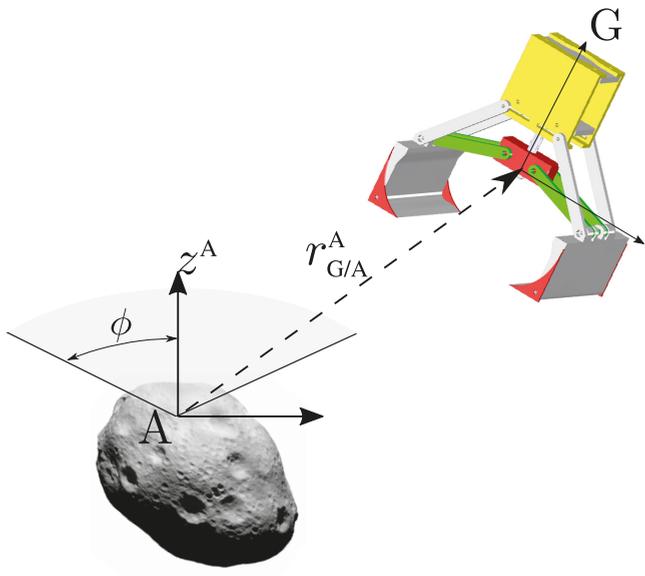


Figure 6. Approach slope constraint.

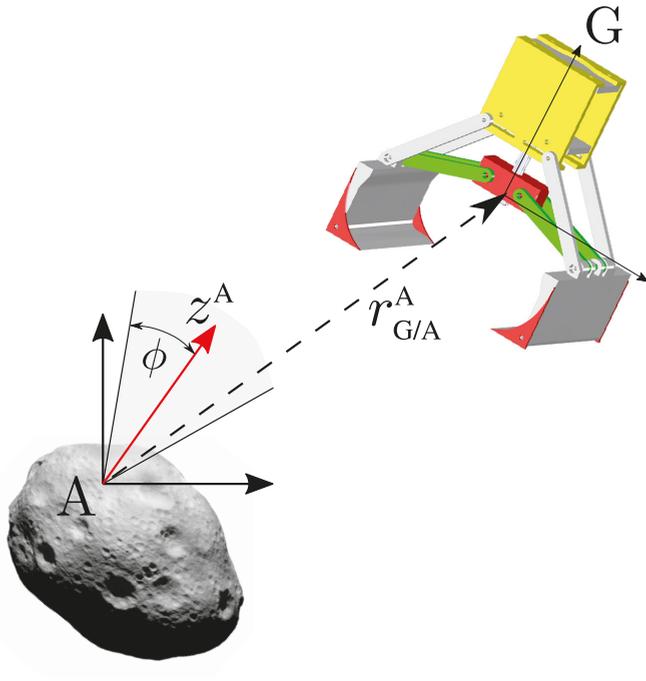


Figure 7. Upper-and-lower bounds constraint.

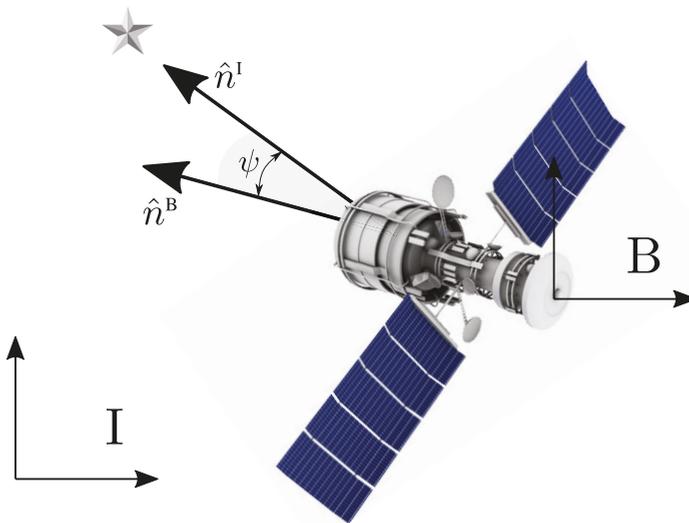


Figure 8. General attitude constraint with respect to inertial directions.

6. Conclusions

In this paper we have explored the use of dual quaternions for robot modeling. In particular, the main contribution of this paper is a generalizable framework to capture the kinematics of

spacecraft-mounted robotic manipulators using a dual quaternion approach. We took a bare-bones approach that built up to a convenient description of the end-effector's dual velocity, making use of a more intuitive forward kinematics methodology than the existing methods in the literature. Previous works on robot kinematics using dual quaternions provided either strict geometry-dependent approaches or were only applicable to fixed-base robots. The work presented herein is highly relevant in combination with the latest literature in dynamic modeling of robot manipulators using dual quaternions. Additionally, in our study of kinematics, we developed a convenient and simple-to-implement representation of the body-frame Jacobian matrix. The proposed form of the Jacobian exploits a convenient matrix representation of the adjoint dual quaternion transformation so that, in combination with the newly proposed form of the screw matrix, it avoids the artificial separation of the contribution by the generalized speeds of a given joint. Finally, we have provided a summary, and re-interpretation, of several existing results on the topic of dual quaternions, emphasizing their applicability on spacecraft-mounted robots. These included results on the exponential and logarithmic maps, an exposition on the use of the DH parameters, and finally the casting of the dual quaternion representation of constraints (originally developed for EDL purposes) interpreted in the context of a gripper-target system on-board a spacecraft.

Future work in this area will aim at implementing kinematic control laws for end-effector pose control when the based is not fixed to an inertial reference frame. This should be possible by following the steps in Özgür and Mezouar [20], and through the use of the Generalized Jacobian Matrix [28].

Author Contributions: Both authors contributed equally in the preparation of the manuscript.

Funding: This research was performed, in part, at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration, and funded through the internal Research and Technology Development program under award number SP.17.0004.008 RSA. This research was also performed under National Science Foundation award NRI1426945.

Acknowledgments: The authors would like to thank David S. Bayard, from the G&C section at the Jet Propulsion Laboratory for valuable conversations that improved the content of this article.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DQ Dual quaternions
EDL Entry, Descent and Landing

References

1. Reed, B.B.; Smith, R.C.; Naasz, B.J.; Pellegrino, J.F.; Bacon, C.E. The Restore-L Servicing Mission. *AIAA Space Forum* **2016**. [[CrossRef](#)]
2. NASA Goddard Space Flight Center. On-Orbit Satellite Servicing Study, Project Report. In *National Aeronautics and Space Administration, Goddard Space Flight Center*; Technical Report; NASA: Greenbelt, MD, USA, 2010.
3. Saha, S.K.; Shah, S.V.; Nandihal, P.V. Evolution of the DeNOC-based dynamic modelling for multibody systems. *Mech. Sci.* **2013**, *4*, 1–20. [[CrossRef](#)]
4. Todorov, E.; Erez, T.; Tassa, Y. MuJoCo: A physics engine for model-based control. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Portugal, 7–12 October 2012.
5. Featherstone, R. *Rigid Body Dynamics Algorithms*; Springer: Berlin/Heidelberg, Germany, 2008.
6. Wang, J.; Liang, H.; Sun, Z.; Zhang, S.; Liu, M. Finite-Time Control for Spacecraft Formation with Dual-Number-Based Description. *J. Guid. Control Dyn.* **2012**, *35*, 950–962. [[CrossRef](#)]
7. Filipe, N.; Tsiotras, P. Simultaneous Position and Attitude Control Without Linear and Angular Velocity Feedback Using Dual Quaternions. In Proceedings of the 2013 American Control Conference, Washington, DC, USA, 17–19 June 2013; pp. 4815–4820.

8. Filipe, N.; Tsiotras, P. Rigid Body Motion Tracking Without Linear and Angular Velocity Feedback Using Dual Quaternions. In Proceedings of the European Control Conference, Zurich, Switzerland, 17–19 July 2013; pp. 329–334.
9. Seo, D. Fast Adaptive Pose Tracking Control for Satellites via Dual Quaternion Upon Non-Certainty Equivalence Principle. *Acta Astronaut.* **2015**, *115*, 32–39. [[CrossRef](#)]
10. Tsiotras, P.; Valverde, A. Dual Quaternions as a Tool for Modeling, Control, and Estimation for Spacecraft Robotic Servicing Missions. In Proceedings of the Texas A&M University / AAS John L. Junkins Astrodynamics Symposium, College Station, TX, USA, 20–21 May 2018.
11. Valverde, A.; Tsiotras, P. Modeling of Spacecraft-Mounted Robot Dynamics and Control Using Dual Quaternions. In Proceedings of the 2018 American Control Conference, Milwaukee, WI, USA, 27–29 June 2018.
12. Perez, A.; McCarthy, J. Dual Quaternion Synthesis of Constrained Robotic Systems. *J. Mech. Des.* **2004**, *126*, 425–435. [[CrossRef](#)]
13. Perez, A. Dual Quaternion Synthesis of Constrained Robotic Systems. Ph.D. Thesis, University of California, Irvine, CA, USA, 2003.
14. Pennestri, E.; Stefanelli, R. Linear algebra and numerical algorithms using dual numbers. *Multibody Syst. Dyn.* **2007**, *18*, 323–344. [[CrossRef](#)]
15. Radavelli, L.A.; De Pieri, E.R.; Martins, D.; Simoni, R. Points, Lines, Screws and Planes in Dual Quaternions Kinematics. In *Advances in Robot Kinematics*; Lenarcic, J., Khatib, O., Eds.; Springer: Berlin, Germany, 2014; pp. 285–293.
16. Leclercq, G.; Lefèvre, P.; Blohm, G. 3-D Kinematics Using Dual Quaternions: Theory and Applications in Neuroscience. *Front. Behav. Neurosci.* **2013**, *7*, 1–7. [[CrossRef](#)] [[PubMed](#)]
17. Quiroz-Omaña, J.J.; Adorno, B.V. Whole-Body Kinematic Control of Nonholonomic Mobile Manipulators Using Linear Programming. *J. Intell. Robot. Syst.* **2017**, *91*, 263–278. [[CrossRef](#)]
18. Brodsky, V.; Shoham, M. Dual numbers representation of rigid body dynamics. *Mech. Mach. Theory* **1999**, *34*, 693–718. [[CrossRef](#)]
19. Filipe, N. Nonlinear Pose Control and Estimation for Space Proximity Operations: An Approach Based on Dual Quaternions. Ph.D. Thesis, Georgia Institute of Technology, Atlanta, GA, USA, 2014.
20. Özgür, E.; Mezouar, Y. Kinematic Modeling and Control of a Robot Arm Using Unit Dual Quaternions. *Robot. Autom. Syst.* **2016**, *77*, 66–73. [[CrossRef](#)]
21. Filipe, N.; Tsiotras, P. Adaptive Position and Attitude-Tracking Controller for Satellite Proximity Operations Using Dual Quaternions. *J. Guid. Control Dyn.* **2014**, *38*, 566–577. [[CrossRef](#)]
22. Bhat, S.; Bernstein, D. A topological obstruction to global asymptotic stabilization of rotational motion and the unwinding phenomenon. In Proceedings of the 1998 American Control Conference, Philadelphia, PA, USA, 26 June 1998; doi:10.1109/acc.1998.688361.
23. Murray, R.M.; Li, Z.; Sastry, S.S. *A Mathematical Introduction to Robotic Manipulation*; CRC Press: Boca Raton, FL, USA, 1994.
24. Jazar, R.N. *Theory of Applied Robotics: Kinematics, Dynamics, and Control*; Springer: Berlin, Germany, 2010.
25. Gan, D.; Liao, Q.; Wei, S.; Dai, J.; Qiao, S. Dual Quaternion-Based Inverse Kinematics of the General Spatial 7R mechanism. *Proc. Inst. Mech. Eng. C J. Mech. Eng.* **2008**, *222*, 1593–1598. [[CrossRef](#)]
26. Adorno, B.V. Two-Arm Manipulation: From Manipulators to Enhanced Human-Robot Collaboration. Ph.D. Thesis, Université Montpellier II Sciences et Techniques du Languedoc, Montpellier, France, 2011.
27. Lee, U.; Mesbahi, M. Constrained Autonomous Precision Landing via Dual Quaternions and Model Predictive Control. *J. Guid. Control Dyn.* **2017**, *40*, 292–308. [[CrossRef](#)]
28. Umetani, Y.; Yoshida, K. Resolved motion rate control of space manipulators with generalized Jacobian matrix. *IEEE Trans. Robot. Autom.* **1989**, *5*, 303–314. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Hexapods with Plane-Symmetric Self-Motions

Georg Nawratil

Institute of Discrete Mathematics and Geometry, Vienna University of Technology,
Wiedner Hauptstrasse 8-10/104, 1040 Vienna, Austria; nawratil@geometrie.tuwien.ac.at;
Tel.: +43-1-58801-104362; Fax: +43-1-58801-9104362

Received: 30 April 2018; Accepted: 5 June 2018; Published: 8 June 2018

Abstract: A hexapod is a parallel manipulator where the platform is linked with the base by six legs, which are anchored via spherical joints. In general, such a mechanical device is rigid for fixed leg lengths, but, under particular conditions, it can perform a so-called self-motion. In this paper, we determine all hexapods possessing self-motions of a special type. The motions under consideration are so-called plane-symmetric ones, which are the straight forward spatial counterpart of planar/spherical symmetric rollings. The full classification of hexapods with plane-symmetric self-motions is achieved by formulating the problem in terms of algebraic geometry by means of Study parameters. It turns out that besides the planar/spherical symmetric rollings with circular paths and two trivial cases (butterfly self-motion and two-dimensional spherical self-motion), only one further solution exists, which is the so-called Duporcq hexapod. This manipulator, which is studied in detail in the last part of the paper, may be of interest for the design of deployable structures due to its kinematotropic behavior and total flat branching singularities.

Keywords: hexapod; self-motion; spatial symmetric rolling; plane-symmetric motion; Duporcq manipulator

1. Introduction

In planar kinematics, the instantaneous pole P traces the so-called fixed/moving polode in the fixed/moving system during the constrained motion of a given mechanism. It is well known that this motion can also be generated by the rolling of the moving polode ϕ along the fixed polode ϕ_0 without sliding. If the polodes are symmetric with respect to the pole tangent t , then the motion is called planar symmetric rolling (cf. Figure 1, left). In 1826, this motion was first (with the exception of the already known symmetric circle rolling yielding the limacons of Pascal) studied by Quetelet [1], who pointed out the following property (cf. [2]): *The path x of a point X under this special planar motion can be generated by the reflexion of a point X_0 of the fixed system on each tangent of ϕ_0 .* This can also be reformulated as follows: *x can be obtained by a central dilation with center X_0 and scale factor 2 (i.e., central doubling) of X_0 's pedal-curve f with respect to ϕ_0 .* A detailed study of the planar symmetric rolling was done by Bereis [3], Bottema [4] and Tölke (cf. [2] and the references given therein).

The spherical counterpart of this motion is called spherical symmetric rolling and was extensively studied by Tölke in a series of papers, which are summarized and referenced in [2]. The spherical version of the above given characterization also holds true for the spherical symmetric rolling (cf. Figure 1, right).

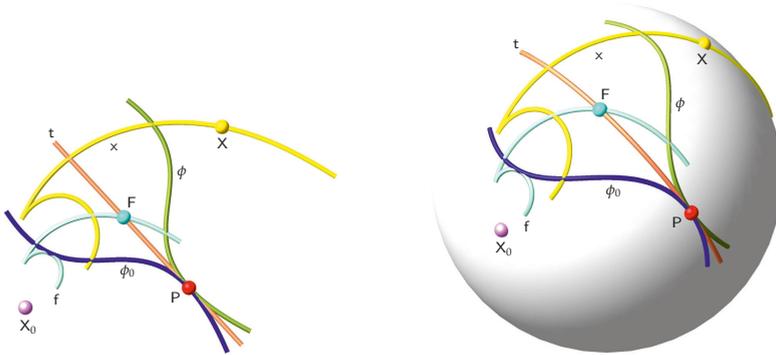


Figure 1. Sketch of the planar symmetric rolling (left) and the spherical symmetric rolling (right). The pedal-point of the fixed point X_0 with respect to the pol-tangent t is denoted by F .

From another perspective, a planar/spherical symmetric rolling can also be generated by reflecting the fixed system in a 1-parametric continuous set of lines/great circles. This point of view is of importance for the spatial generalization of symmetric rollings, which can be done in multiple ways:

1. Darboux noted in [5] (No. 61) a 2-parametric spatial motion, which is generated by the rolling of a moving surface Φ on an indirect congruent fixed surface Φ_0 . It also holds that the path-surface of a point X can be generated by the reflexion of a point X_0 of the fixed system on each tangent-plane of Φ_0 ; for example, the path-surface can be obtained by a central doubling of X_0 's pedal-surface with respect to Φ_0 's tangent-planes.
2. Krames [6] considered the so-called line-symmetric motion as the 1-parametric spatial analogue of the planar/spherical symmetric rolling. These motions are obtained by reflecting the moving system in a 1-parametric continuous set of lines, which form the so-called *basic surface* Γ (cf. Figure 2, left). Krames reasoned this by the fact that the path x of a point X under a line-symmetric motion can be generated by the reflexion of a point X_0 on each generator g of Γ ; for example, x can be obtained by a central doubling of X_0 's pedal-curve f with respect to Γ 's rulings. However, it should be pointed out that Γ differs from the fixed axode Φ_0 (generated by the central tangents of Γ). However, Φ_0 and the moving axode Φ are at each time instant symmetric with respect to the axis p of the instantaneous screw, which is in general not an instantaneous rotation. For further details and references on this motion type, please see [7,8] (§7 of Ch. 4) and [9].
3. It is astonishing that neither Tölke [2] (Section 3.1) nor Krames [6] (p. 394) mentioned the more apparent generalization by reflecting the fixed system in a 1-parametric continuous set of planes. Less attention was paid to these so-called *plane-symmetric motions* in the literature until now. We summarize the known results in the next section.

Remark 1. Note that the term *plane-symmetric motion* was also used in [10] (§3.3) for a superset of the above described motions, which is characterized by the sole property that "the same equation describes the motion and its inverse, but with respect to reference systems that are a reflection of each other". In order to avoid confusions, we point out that we do not mean this superset by using this wording.

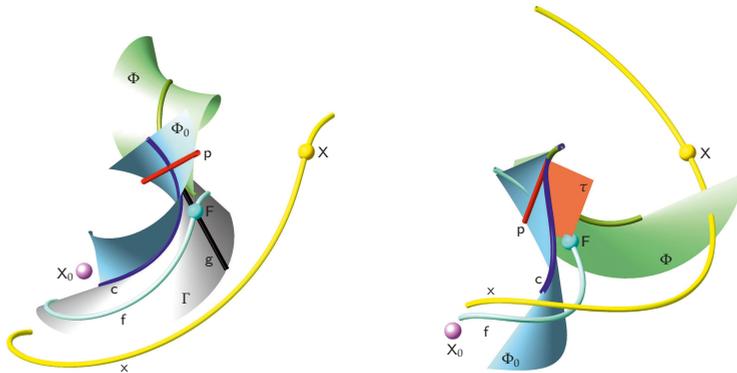


Figure 2. Sketch of the line-symmetric motion (left) and the spatial symmetric rolling (right). For the illustrations, the basic surface Γ of the line-symmetric motion and the fixed axode Φ_0 of the spatial symmetric rolling have been chosen as tangent-surfaces of a straight cubic circle c . F denotes the pedal-point of the fixed point X_0 with respect to (left) the generator g of Γ and (right) the tangent-plane τ along the instantaneous axis p of rotation, respectively.

1.1. Review on Plane-Symmetric Motions

The basic properties of this motion type are reported in [8] (§8 of Ch. 4). Given is a 1-parametric continuous set of planes $\tau(t)$, where the parameter t can be seen as time. By reflecting the fixed frame δ_0 on the plane $\tau(t)$, we obtain the pose δ_0^t of the plane-symmetric motion.

Let us consider to infinitesimal neighboring poses δ_0^t and $\delta_0^{t+\Delta t}$ of the plane-symmetric motion. Now, one can transform δ_0^t into $\delta_0^{t+\Delta t}$ by a reflexion on $\tau(t)$ followed by a further reflexion on $\tau(t + \Delta t)$. It is well known that this is a pure rotation about the line of intersection of $\tau(t)$ and $\tau(t + \Delta t)$. Moreover, this is exactly a torsal ruling of the developable surface enveloped by the given 1-parametric set of planes. As a consequence, the fixed axode Φ_0 is a developable surface (It is well known (e.g., [11] (Thms. 5.1.7 and 6.1.3)) that every developable surface is composed of cylindrical, conical or tangent-surfaces) and the corresponding moving axode Φ is obtained by reflecting Φ_0 in Φ_0 's tangent-plane τ along the instantaneous axis p of rotation (cf. Figure 2, right). Now, the path x of a point X under a plane-symmetric motion can be generated by the reflexion of a point X_0 on each tangent-plane τ of Φ_0 ; i.e., x can be obtained by a central doubling of X_0 's pedal-curve f with respect to Φ 's tangent-planes.

Due to all these properties, the plane-symmetric motion seems to be the straightforward spatial counterpart of the planar/spherical symmetric rolling. Therefore, we call a plane-symmetric motion also a *spatial symmetric rolling*.

As far as the author knows, these spatial symmetric rollings are only explicitly mentioned in a practical example by Kunze and Stachel [12], who pointed out that the relative motion of opposite systems of a threefold-symmetric Bricard linkage (e.g., the *invertible cube* of Schatz) is a plane-symmetric one. Clearly, this also holds for the more general class of plane-symmetric Bricard linkages [13], where the two opposite systems not containing a rotation-axis spanning the plane of symmetry also possess a plane-symmetric relative motion during the overconstrained motion of the closed 6R-chain.

1.2. Motivation and Outline

One of the author's main research interests are hexapods with self-motions, i.e., overconstrained parallel manipulators where the platform is linked with the base by six legs, which are anchored via spherical red joints (Due to the spherical joints at the platform and the base, each leg can rotate about its carrier line without changing the pose of the platform. These uncontrolled leg-movements are not

meant by the term *self-motion*). All these mechanical devices are solutions to the still unsolved problem posed by the French Academy of Science for the *Prix Vaillant* of the year 1904, which is also known as Borel–Bricard problem and reads as follows [14]: “Determine and study all displacements of a rigid body in which distinct points of the body move on spherical paths.” In order to avoid trivial solutions of the problem, the following assumption should hold for the remainder of the article.

Assumption 1. *The platform anchor points m_1, \dots, m_6 of the hexapod as well as the corresponding base anchor points M_1, \dots, M_6 should span in each case at least a plane.*

It is well known that so-called architecturally singular hexapods (A hexapod is called architecturally singular if the six legs belong in each relative pose of the platform with respect to the base to a linear line complex) possess self-motions in each pose (over \mathbb{C}). These special solutions to the Borel–Bricard problem are already well studied (A review on this topic is given in [15] (Section 3.1)). The approaches for the determination of non-architecturally singular hexapods recorded in the literature (Note that we do not claim that the following list of given references is complete), can roughly be divided into the following two groups:

1. Assumptions on the geometry of the platform and base; e.g.,

- (a) linear mapping between platform and base [16–22],
- (b) symmetry properties of platform and base [20–24],
- (c) special topology (e.g., octahedral structure [25]),

or a combination of these assumptions (e.g., [20–22])

2. Assumptions on the self-motion; e.g.,

- (a) line-symmetric self-motion [9],
- (b) type II Darboux–Mannheim self-motion [26],
- (c) Schoenflies self-motion [27],
- (d) translational self-motion [28],
- (e) self-motion of maximal degree [29],

or more generally characterizations like *linear relations between direction cosines* [30–33].

Note that these assumptions are done in order to reduce the complexity of the problem, as one has to deal with 30 design parameters (24 for the geometry and six leg lengths, whereby the number of 30 can be reduced by one due to the freedom of scaling) and six degrees of freedom.

We want to follow the second approach by assuming that the self-motions are symmetric rollings. Therefore, this paper closes a gap as line-symmetric self-motions and point-symmetric (Point-symmetric motions are obtained by reflecting the fixed system in a 1-parametric continuous set of points and according to [7] (Section 8), these motions are pure translations) self-motions are already well-studied [9,28]. In addition, this motion-type seems to be a good candidate for self-motions, due to the following property implied by the symmetry of the motion:

Theorem 1. *If a point A of the moving system traces a spherical curve with center B_0 during a plane-symmetric motion, then also the point B of the moving system has a spherical trajectory about the point A_0 , where A and A_0 as well as B and B_0 , are plane-symmetric points of the moving and fixed frame with respect to the tangent-plane τ along the instantaneous axis ρ of rotation. As a consequence, the set of points with spherical trajectories is indirectly congruent to the set of corresponding sphere centers.*

In the remainder of the paper, we call the replacement of the point pair (A, B_0) by (B, A_0) the “*symmetric leg-replacement*”.

Remark 2. Clearly, the lower dimensional version of Theorem 1 is also true for the planar/spherical symmetric rolling. Moreover, Theorem 1 also holds for point-symmetric motions, if “plane-reflection” is substituted by “point-reflection”. A similar result holds for line-symmetric motions; one only has to replace “plane-reflection” by “line-reflection” and “indirectly congruent” by “directly congruent” (see e.g., [9]).

The paper is structured as follows: We start with the discussion of planar/spherical symmetric rolling motions with circular paths in Section 2.1. In Section 2.2, we formulate the problem of determining hexapods with plane-symmetric self-motions in terms of algebraic geometry by means of Study parameters. Based on this description, the problem is solved in Section 3. One of the obtained solutions is the so-called *Duporcq hexapod*, which is discussed in more detail in Section 4. The paper is closed by a conclusion (cf. Section 5).

2. Preliminary Considerations and Preparatory Work

As far as the author knows, no hexapods with plane-symmetric self-motions are reported in the literature so far. From known results in planar/spherical kinematics, which are reviewed in the next subsection, we can immediately construct such hexapods.

2.1. Planar/Spherical Symmetric Rollings with Circular Paths

Clearly, a pure rotation is a planar/spherical symmetric rolling where every point of the moving system traces a circle. Besides this trivial case, which we meet again under the notation of a so-called butterfly self-motions (cf. later given Theorem 4), the following planar/spherical symmetric rollings with circular trajectories exist:

- The planar symmetric rolling motions with points running on circular paths are well known due to the study of Bereis [3]. In this case, the polodes are either ellipses or hyperbolas and the focals (two real, two complex) of the moving ellipse/hyperbola are running on circles. They are the Burmester points of this motion. These motion can be realized by the mechanisms illustrated in Figure 3.

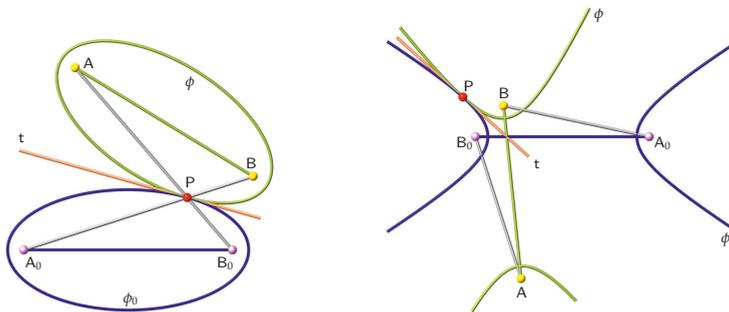


Figure 3. Twin-crank mechanisms with non-counter-rotating cranks (left): In this case, the polodes are ellipses. Twin-crank mechanism with counter-rotating cranks (right): In this case, the polodes are hyperbolas.

- Unfortunately, the considerations of Bereis cannot be generalized straightforward to the sphere (cf. [2] (p. 195)), as in spherical kinematics six Burmester points exist (e.g., [8] (p. 216)). However, we can do the reasoning in a different way. Due to [28] (Theorem 6), one can assume without loss of generality that only two points of a moving body can have spherical trajectories. According to the spherical version of Theorem 1, a second point is also running on a circle due to the symmetric leg-replacement (With the exceptional case that the first leg is orthogonal to the pole tangent, but

this will not yield a closed loop; i.e., a spherical parallel manipulator). Thus, we can only end up with a spherical isogram illustrated in Figure 4, which is studied in more detail in [34].

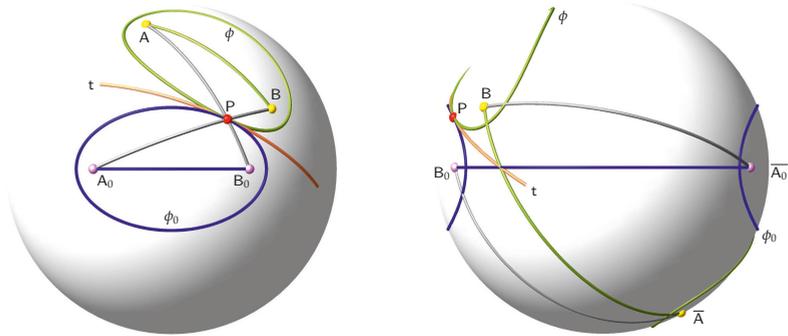


Figure 4. As on the sphere points can be replaced by their antipodes, it can easily be seen that every spherical conic can be interpreted as a spherical ellipse (e.g., [35] (Section 10.1)). The left and the right figure show the same symmetric rolling motion. If we replace A and A₀ by their antipodal points \bar{A} and \bar{A}_0 , respectively, and look on the sphere from the right side, then we get the figure illustrated on the right-hand side.

From the discussed planar and spherical case, one can easily construct hexapods with plane-symmetric self-motions (see Figure 5).

Remark 3. Note that the hexapods of Figure 5 do not only possess the illustrated plane-symmetric self-motions, but also the already mentioned butterfly self-motions (cf. later given Theorem 4).

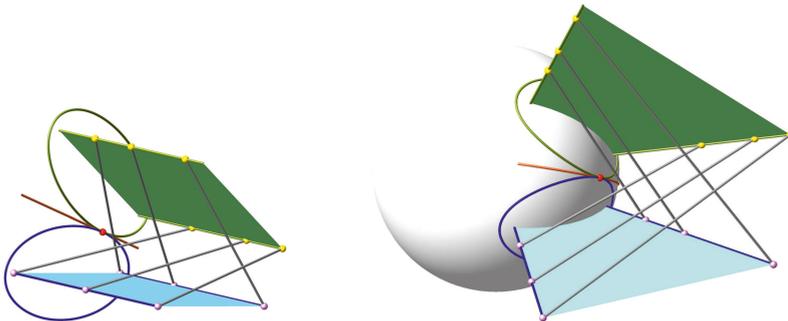


Figure 5. Hexapods with plane-symmetric self-motions, where the platform (green) and the base (blue) are both planar. The axodes of the self-motions are cylinders (left) and cones (right), respectively, but we only illustrated the planar/spherical directrices of these singular quadrics to see better their connection to the planar/spherical symmetric rolling displayed in Figures 3 and 4, respectively.

2.2. Mathematical Framework

For the algebraic formulation of our problem, we want to use Study parameters $(e_0 : e_1 : e_2 : e_3 : f_0 : f_1 : f_2 : f_3)$, which are nothing else than homogenized dual unit-quaternions $\mathfrak{E} + \varepsilon\mathfrak{F}$ with

$$\mathfrak{E} = e_0 + e_1\mathbf{i} + e_2\mathbf{j} + e_3\mathbf{k} \quad \text{and} \quad \mathfrak{F} = f_0 + f_1\mathbf{i} + f_2\mathbf{j} + f_3\mathbf{k}, \tag{1}$$

where $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are the well-known quaternionic units and ε the dual unit with the property $\varepsilon^2 = 0$.

Now, all real points of the seven-dimensional Study parameter space P^7 , which are located on the so-called Study quadric $\Psi : \sum_{i=0}^3 e_i f_i = 0$, correspond to a Euclidean displacement with exception of the three-dimensional subspace E of Ψ given by $e_0 = e_1 = e_2 = e_3 = 0$, as its points cannot fulfill the condition $N \neq 0$ with $N = e_0^2 + e_1^2 + e_2^2 + e_3^2$. The translation vector $\mathbf{t} := (t_1, t_2, t_3)^T$ and the rotation matrix $\mathbf{R} := (r_{ij})$ of the corresponding Euclidean displacement $\mathbf{x} \mapsto \mathbf{R}\mathbf{x} + \mathbf{t}$ are given by:

$$t_1 = 2(e_0 f_1 - e_1 f_0 + e_2 f_3 - e_3 f_2), \quad t_2 = 2(e_0 f_2 - e_2 f_0 + e_3 f_1 - e_1 f_3), \quad t_3 = 2(e_0 f_3 - e_3 f_0 + e_1 f_2 - e_2 f_1),$$

and

$$\mathbf{R} = \begin{pmatrix} e_0^2 + e_1^2 - e_2^2 - e_3^2 & 2(e_1 e_2 - e_0 e_3) & 2(e_1 e_3 + e_0 e_2) \\ 2(e_1 e_2 + e_0 e_3) & e_0^2 - e_1^2 + e_2^2 - e_3^2 & 2(e_2 e_3 - e_0 e_1) \\ 2(e_1 e_3 - e_0 e_2) & 2(e_2 e_3 + e_0 e_1) & e_0^2 - e_1^2 - e_2^2 + e_3^2 \end{pmatrix},$$

if the normalizing condition $N = 1$ is fulfilled.

Clearly, the reflection on a plane is an orientation-reversing congruence transformation, which cannot be described directly by the Study parameters. Therefore, we follow the approach of Selig and Husty [7] (Section 8), which is as follows: We start with a reflexion on a fixed plane; say the xy -plane of the fixed frame \mathfrak{S}_0 . By this plane-reflection of \mathfrak{S}_0 , we obtain $\tilde{\mathfrak{S}}_0$. In addition, we apply the reflexion on the plane $\tau(t)$, which finally yields the pose $\tilde{\mathfrak{S}}_0^+$. As the composition of two plane-reflexions is again a direct congruence transformation, we can describe the plane-symmetric motions in this way. If $\tau(t)$ and the xy -plane of \mathfrak{S}_0 are

- not parallel, then the composition is a rotation about the line of intersection,
- parallel, then the composition is a translation orthogonal to these planes.

This yields that the plane-symmetric motions are given by $e_3 = f_0 = f_1 = f_2 = 0$. Moreover, it should be noted that the Study condition is fulfilled identically, thus the set of plane-symmetric motions corresponds to a three-dimensional generator space P of Ψ which intersects E in a line. Based on this description, we analyze the relation between plane-symmetric motions and line-symmetric ones in the next theorem:

Theorem 2. *A plane-symmetric motion is also a line-symmetric one if and only if there exists a linear relation $\alpha e_0 + \beta e_1 + \gamma e_2 + \delta f_3 = 0$ with $(\alpha, \beta, \gamma, \delta) \neq (0, 0, 0, 0)$ between the remaining Study parameters.*

Proof. For the proof, we need an algebraic characterization of line-symmetric motions in terms of Study parameters. It is well-known that there always exist, a Cartesian frame in the moving system in a way that $e_0 = f_0 = 0$ holds for a line-symmetric motion. Then, $(e_1 : e_2 : e_3 : f_1 : f_2 : f_3)$ are the Plücker coordinates of the generators of the basic surface with respect to the fixed frame.

A change of the moving system can be achieved by a so-called right multiplication; i.e., $(\mathfrak{E} + \varepsilon \delta) \circ (\mathfrak{R} + \varepsilon \mathfrak{G})$ where \circ stands for the quaternionic multiplication. If we denote this product by $\mathfrak{G} + \varepsilon \mathfrak{H}$, the corresponding entries g_0 and h_0 read as follows (under consideration of $e_3 = f_0 = f_1 = f_2 = 0$):

$$g_0 := r_0 e_0 - r_1 e_1 - r_2 e_2, \quad h_0 := s_0 e_0 - s_1 e_1 - s_2 e_2 - r_3 f_3. \tag{2}$$

If $\delta = 0$ holds, then we set $r_0 = \alpha, r_1 = -\beta, r_2 = -\gamma$ and $s_0 = s_1 = s_2 = r_3 = 0$. For $\delta \neq 0$, we set $s_0 = \alpha, s_1 = -\beta, s_2 = -\gamma, r_3 = -\delta$ and $r_0 = r_1 = r_2 = 0$. For both cases, we get $g_0 = h_0 = 0$, which finishes the sufficiency of the linear relation between e_0, e_1, e_2, f_3 .

Its necessity can also be seen from Equation (2), as without such a linear relation, the condition $g_0 = h_0 = 0$ can only be fulfilled for $r_0 = r_1 = r_2 = r_3 = 0$, which yields a contradiction as \mathfrak{R} has to differ from the zero-quaternion. \square

A further important theorem in this context is the following:

Theorem 3. *A plane-symmetric motion is also a line-symmetric one if and only if it is a planar motion or a spherical motion.*

Proof. If the linear relation equals $f_3 = \alpha e_0 + \beta e_1 + \gamma e_2$, then it can easily be checked by direct computations that the point $(\gamma, -\beta, -\alpha)$ is mapped to the point $(\gamma, -\beta, \alpha)$ for all e_0, e_1, e_2 fulfilling $N = 1$. Therefore, $(\gamma, -\beta, \alpha)$ is the center of the spherical motion.

If the linear relation equals $\alpha e_0 + \beta e_1 + \gamma e_2 = 0$, then it can easily be checked by direct computations that the direction $(\gamma, -\beta, -\alpha)$ is mapped to the direction $(\gamma, -\beta, \alpha)$ for all e_0, e_1, e_2 fulfilling $N = 1$. Therefore, the direction $(\gamma, -\beta, \alpha)$ remains fixed under the motion. Moreover, the translation vector (t_1, t_2, t_3) is orthogonal to this direction, which already proves that the motion is planar. \square

These two theorems imply the following statement:

Corollary 1. *If we embed the planar and spherical symmetric rollings into $SE(3)$, then they can also be seen as line-symmetric motions.*

Therefore, the self-motions of the hexapods illustrated in Figure 5 are plane-symmetric and line-symmetric at the same time. This raises also the question of whether self-motions exist, which are plane-symmetric but not line-symmetric. The answer is given within the next section.

3. Plane-Symmetric Self-Motions

The coordinate vector of the base point M_i with respect to the fixed system is given by $\mathbf{M}_i = (A_i, B_i, C_i)^T$. The position of the corresponding platform anchor point $m_i(t)$ is obtained by reflecting a point $m_{i,0}$ with fixed coordinates $\mathbf{m}_i = (a_i, b_i, c_i)^T$ in a 1-parametric continuous set of planes $\tau(t)$. Instead of these reflexions, we use direct isometries based on the Study representation described in Section 2.2 (i.e., $e_3 = f_0 = f_1 = f_2 = 0$). Therefore, the locus of the corresponding platform anchor point m_i with respect to the fixed frame can be parametrized as $\mathbf{R}\bar{\mathbf{m}}_i + \mathbf{t}$ with $\bar{\mathbf{m}}_i = (a_i, b_i, -c_i)^T$.

The condition that the point m_i is located on a sphere centered in M_i with radius d_i is a quadratic homogeneous equation in the Study parameters according to Husty [36]. For our setup, this so-called sphere condition Λ_i has the following form:

$$\begin{aligned} \Lambda_i : & (a_i^2 + b_i^2 + c_i^2 + A_i^2 + B_i^2 + C_i^2 - d_i^2)N - 4(c_i + C_i)e_0f_3 + 4(b_i + B_i)e_1f_3 - 4(a_i + A_i)e_2f_3 \\ & - 2(a_iA_i + b_iB_i - c_iC_i)e_0^2 - 2(a_iA_i - b_iB_i + c_iC_i)e_1^2 + 2(a_iA_i - b_iB_i - c_iC_i)e_2^2 \\ & - 4(c_iB_i + b_iC_i)e_0e_1 + 4(c_iA_i + a_iC_i)e_0e_2 - 4(b_iA_i + a_iB_i)e_1e_2 + 4f_3^2 = 0. \end{aligned} \quad (3)$$

It corresponds to a quadric in the three-dimensional projective space P^3 with homogenous coordinates $(e_0 : e_1 : e_2 : f_3)$. The symmetric leg-replacement (cf. Theorem 1) can also easily be seen within this formula, as it is invariant under the following permutations: $A_i \leftrightarrow a_i, B_i \leftrightarrow b_i, C_i \leftrightarrow c_i$. Due to this symmetry, we only have to find spatial rolling motions where three points have a spherical trajectory. This means that the corresponding three quadrics Λ_1, Λ_2 and Λ_3 of P^3 have to have a curve in common, which can be a

1. straight line,
2. conic section,
3. cubic curve,
4. quartic curve.

In the following subsections these cases are discussed separately.

3.1. Intersection Curve Is a Straight Line

It is well-known that straight lines in the Study quadric correspond with either rotations about a line or straight translations. As the second option is not possible due to the sphere condition, we are only left with the rotation case. In the first step, we ask under which conditions two quadrics Λ_1 and Λ_2 have a straight line in common.

- General Case:** Let us assume that $M_1 \neq M_2$ and $m_1(t) \neq m_2(t)$ hold. Clearly, the straight line in P^3 has to correspond with a rotation about the line G spanned by M_1 and M_2 . Therefore, the line $g(t)$ spanned by $m_1(t)$ and $m_2(t)$ generates either a hyperboloid, cone or cylinder of revolution with axis G . Moreover, all these poses of the platform points have to be obtained by plane-reflexions of the points $m_{1,0}$ and $m_{2,0}$, respectively. This already implies that the 1-parametric set of planes $\tau(t)$ has to be a pencil of planes with axis G . Therefore, the leg lengths d_1 and d_2 are given by

$$d_i = \text{dist}(M_i, m_{i,0}) = \sqrt{(A_i - a_i)^2 + (B_i - b_i)^2 + (C_i - c_i)^2}, \tag{4}$$

which is already the necessary and sufficient condition for the two quadrics Λ_1 and Λ_2 to have a straight line in common.

- Special Case:** As the case $M_1 = M_2$ and $m_1(t) = m_2(t)$ cannot arise (legs are identical), we only have to discuss one further case due to the symmetric leg-replacement. Without loss of generality, we can assume $M_1 \neq M_2$ and $m_1(t) = m_2(t)$. Now, $m_1(t) = m_2(t)$ has to trace a circle about the line G , which in fact implies the same condition given in Equation (4) for $i = 1, 2$.

Under consideration of the notation that (M_i, m_i) and (M_{i+3}, m_{i+3}) are coupled by the symmetric leg-replacement (for $i = 1, 2, 3$), we can immediately formulate the following theorem.

Theorem 4. *Up to symmetric leg-replacements, the three quadrics Λ_1, Λ_2 and Λ_3 have a line in common if and only if Equation (4) holds for $i = 1, 2, 3$ and M_1, M_2, M_3 are collinear. The corresponding self-motion of the hexapod is a butterfly self-motion about the line spanned by M_1, M_2, M_3 , where $M_i = m_{i+3}$ holds for $i = 1, 2, 3$.*

As these butterfly self-motions (cf. Figure 6, left) are trivial, they are not of further interest.

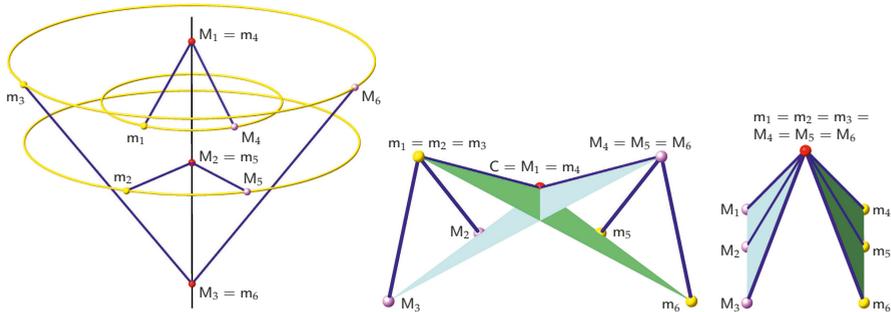


Figure 6. In all three illustrations, the plane of symmetry is always a vertical projecting plane; **left:** butterfly self-motion of a hexapod. Note that not necessarily the three legs obtained by the symmetric leg-replacements have to be added, but any legs where neither M_i or $m_i(t)$ is collinear with M_1, M_2, M_3 for $i = 4, 5, 6$; **center:** situation after performing the Δ -transform; **right:** two-dimensional spherical self-motion.

3.2. Intersection Curve Is a Conic

As the conic is a planar curve, there has to exist a linear relation between the homogenous coordinates $(e_0 : e_1 : e_2 : f_3)$ of P^3 . Therefore, we can apply the Theorems 2 and 3, which imply that we can only end up with planar/spherical symmetric rollings already discussed in Section 2.1.

3.3. Intersection Curve Is Cubic

A necessary condition that Λ_1, Λ_2 and Λ_3 have a cubic curve in common is that the intersection of two quadrics split up into a line and this cubic. Therefore, condition Equation (4) has to hold. It can easily be checked that Λ_i splits up into two planes:

$$\Lambda_i : 4(C_i e_0 - B_i e_1 + A_i e_2 - f_0)(c_i e_0 - b_i e_1 + a_i e_2 - f_0) = 0 \quad (5)$$

under consideration of Equation (4). Therefore, the cubic has to split up into three lines, which all correspond to plane-symmetric butterfly self-motions already described in Theorem 4. As a consequence, no further discussion of this case is necessary.

3.4. Intersection Curve Is Quartic

We start with the following lemma, which helps to exclude the discussion of special cases arising.

Lemma 1. *If M_1, M_2, M_3 are collinear and $m_1 = m_2 = m_3$ holds (under consideration of symmetric leg-replacements), then the hexapod can only have the following plane-symmetric self-motions:*

1. butterfly self-motion,
2. two-dimensional spherical self-motion,
3. planar/spherical symmetric rollings of Section 2.1.

Proof. If the carrier line G of M_1, M_2, M_3 is always identical with the reflected carrier line g of m_4, m_5, m_6 , then it is clear that the motion can only be a butterfly motion (cf. Figure 6, left).

Moreover, it is trivial, that the motion can only be a planar one if G is always parallel to g (\Rightarrow planar symmetric rolling of Section 2.1).

Now, we discuss the remaining case that, during the plane-symmetric self-motion, one configuration exists, where G and g intersect in one point C . As the first three legs are always in a pencil of lines, one can make a so-called Δ -transform [37] (without changing the self-motion) such that $M_1 = C$ holds. This results in the following relations (cf. Figure 6, center):

$$\overline{M_1 M_4} = \overline{m_4 M_4} \quad \text{and} \quad \overline{m_1 m_4} = \overline{M_1 m_4}. \quad (6)$$

Under consideration of the plane-symmetric setup, these conditions can only be fulfilled if

- $M_1 = m_4$ holds, which yields the spherical symmetric rolling (with center $M_1 = m_4$) of Section 2.1,
- $M_4 = m_1$ holds, which implies a two-dimensional spherical self-motion (with center $M_4 = m_1$; cf. Figure 6, right).

This finishes the proof. \square

Remark 4. *Note that, for the two-dimensional spherical self-motion, the collinearity condition of M_1, M_2, M_3 is not necessary. For the leg lengths of Equation (4) and $m_1 = m_2 = m_3$, the three quadrics Λ_1, Λ_2 and Λ_3 have already a plane in common due to Equation (5).*

If Λ_1, Λ_2 and Λ_3 have a quadric curve in common, they are contained within a pencil of quadrics, which is already spanned by two of them. Therefore, we make the following ansatz:

$$\Sigma : \lambda_1 \Lambda_1 + \lambda_2 \Lambda_2 + \Lambda_3 = 0 \quad \text{with} \quad \lambda_1 \lambda_2 \neq 0. \quad (7)$$

In order to simplify the resulting direct computations, we can select the fixed frame in a clever way based on the following lemma:

Lemma 2. *By applying symmetric leg-replacements, we can assume that M_1, M_2, M_3 span a plane (under consideration of Assumption 1).*

Proof. If M_1, M_2, M_3 are collinear (span the line G), we apply the symmetric leg-replacement to the i th leg for $i \in \{1, 2, 3\}$. Due to Assumption 1, at least one of the M_{i+3} are not located on G , thus, after a reenumeration of anchor points, the lemma holds. \square

Due to Lemma 2, we can assume without loss of generality that the origin of the fixed frame equals M_i , that M_j is located on the positive x -axis of the fixed frame and that M_k is located in the xy -plane of the fixed frame for pairwise distinct $i, j, k \in \{1, 2, 3\}$. As M_1, M_2, M_3 is a triangle there always exist at least four (This number results from the fact that each triangle has at least two acute angles, whose two vertices can be used as M_i) choices for i, j, k in a way that M_k is located in the 1st quadrant of the xy -plane. After a may necessary reenumeration, we can assume:

$$\mathbf{M}_1 = (0, 0, 0)^T, \quad \mathbf{M}_2 = (A_2, 0, 0)^T, \quad \mathbf{M}_3 = (A_3, B_3, 0)^T, \quad (8)$$

with $A_2 > 0, A_3 > 0$ and $B_3 > 0$. Moreover, by selecting the unit-length in a suitable way, we can achieve $A_2 = 1$.

Based on this choice of the fixed frame, we inspect the coefficients of the linear combination Σ given in Equation (7) with respect to the Study parameters. We denote the coefficient of $e_0^i e_1^j e_2^k f_3^l$ by Σ_{ijkl} . From $\Sigma_{1100} = -4c_3 B_3$, we get $c_3 = 0$. Moreover, we can compute d_3^2 from Σ_{2000} . Then, Σ_{0200} equals $4b_3 B_3$, which implies $b_3 = 0$. From $\Sigma_{1100} = 4\lambda_2 c_2$, we get $c_2 = 0$. Now, $\Sigma_{1001} = 4\lambda_1 c_1$ yields $c_1 = 0$. Then, we express A_3 and B_3 from Σ_{0101} and Σ_{0011} , which results in

$$A_3 = -a_3 - \lambda_1 a_1 - \lambda_2 a_2 - \lambda_2, \quad B_3 = -\lambda_1 b_1 - \lambda_2 b_2. \quad (9)$$

Moreover, we can set $\lambda_1 = -1 - \lambda_2$ due to Σ_{0002} . Then, we are only left with the following two conditions arising from Σ_{0110} and Σ_{0020} , respectively:

$$-a_3 b_1 - a_3 b_1 \lambda_2 + a_3 \lambda_2 b_2 - \lambda_2 b_2 = 0, \quad -a_3^2 + a_3 a_1 + a_3 a_1 \lambda_2 - a_3 \lambda_2 a_2 - a_3 \lambda_2 + \lambda_2 a_2 = 0. \quad (10)$$

Eliminating λ_2 out of these equations by resultant method yields:

$$a_3(a_3 - 1)(a_3 b_1 - a_3 b_2 - b_1 a_2 + a_1 b_2). \quad (11)$$

Therefore, we distinguish the following cases:

1. For $a_3 = 0$, Equation (10) imply $\lambda_2 a_2 = 0$ and $\lambda_2 b_2 = 0$, respectively. $a_2 = b_2 = 0$ imply the conditions of Lemma 1.
2. For $a_3 = 1$, Equation (10) imply $\lambda_2 = -1$. Then, the second and third leg are identical under consideration of symmetric leg-replacement.
3. For $a_3 b_1 - a_3 b_2 - b_1 a_2 + a_1 b_2 = 0$, we have to distinguish two cases:
 - (a) $b_1 = b_2$: now, the condition simplifies to $b_1(a_1 - a_2) = 0$. As $b_1 = 0$ implies $B_3 = 0$ a contradiction, we set $a_1 = a_2$. Then, Equation (10) imply $\lambda_2 = -a_3$, which results in the conditions of Lemma 1.
 - (b) $b_1 \neq b_2$: Under this assumption, we can solve this equation for a_3 . A further two cases have to be distinguished:

- i. $b_2a_1 - b_1a_2 - b_2 = 0$: If one solves this equation for a_i , then Equation (10) implies $b_j = 0$ for distinct $i, j \in \{1, 2\}$. In both cases, we end up with the conditions of Lemma 1.
- ii. $b_2a_1 - b_1a_2 - b_2 \neq 0$: Under this assumption, we can solve the condition implied by Equation (10) for λ_2 , which yields:

$$\lambda_2 = \frac{(b_1a_2 - b_2a_1)b_1}{(b_1 - b_2)(b_2a_1 - b_1a_2 - b_2)}. \tag{12}$$

It can easily be checked that the obtained solution corresponds to the hexapod’s platform and base illustrated in Figure 7, which are also known as Duporcq’s complete quadrilaterals [38]. In the remainder of the paper, this interesting solution, which is discussed/studied in more detail in the next section, is called *Duporcq hexapod*. Based on this notation, we can formulate the following theorem.

Theorem 5. *Besides the trivial cases mentioned in Lemma 1, the quadrics Λ_1, Λ_2 and Λ_3 belong to a pencil if and only if they correspond to sphere conditions of three legs of a Duporcq hexapod (which are not identical under symmetric leg-replacements).*



Figure 7. Illustration of Duporcq’s complete quadrilaterals: The base (left) is congruent with the platform (right).

4. Duporcq Hexapod

Due to the results obtained in Section 3 and Theorems 2 and 3, we can conclude that only the Duporcq hexapod of Theorem 5 possesses plane-symmetric self-motions, which are neither planar nor spherical motions. Therefore, we discuss this hexapod in more detail in this section.

In [38], Duporcq describes the following remarkable motion: *Let M_1, \dots, M_6 and m_1, \dots, m_6 be the vertices of two complete quadrilaterals, which are congruent. Moreover, the vertices are labeled in a way that m_i is the opposite vertex of M_i for $i \in \{1, \dots, 6\}$ (cf. Figure 7). Then, there exists a 2-parametric line-symmetric motion where each m_i is running on spheres centered in M_i .*

It is well known [39] (Section 1) that this is an architecturally singular hexapod and that one can remove any leg without changing the direct kinematics of the mechanism. The resulting pentapod is called *Duporcq pentapod* and its line-symmetric self-motions were also studied in [39]. For the coordinatisation of the platform points and base points used in Section 3.4, the 2-parametric line-symmetric self-motion fulfills $e_0 = f_0 = 0$ (cf. [39] (Section 4)).

Remark 5. *Note that the theoretic results of Section 4 are visualized on the basis of the following example:*

$$a_1 = \frac{3}{2}, \quad a_2 = b_1 = 3, \quad b_2 = \frac{9}{4}, \quad d_1^2 = \frac{17}{2}, \quad d_2^2 = \frac{33}{2}. \tag{13}$$

This input data implies

$$a_3 = \frac{15}{2}, \quad A_3 = \frac{8}{7}, \quad B_3 = \frac{6}{7}, \quad d_3^2 = \frac{13231}{196}, \tag{14}$$

with respect to the coordinatisation of the platform points and base points used in Section 3.4.

4.1. Plane-Symmetric Self-Motions of the Duporcq Hexapod

First of all, it should be pointed out that the plane-symmetric self-motions of the Duporcq manipulator were not known until now. They can be computed as follows: We express f_3 from the condition $\Lambda_2 - \Lambda_1$ (which is linear in f_3). Plugging the resulting expression into Λ_1 implies a homogenous quartic equation Y in e_0, e_1, e_2 , which already represents the plane-symmetric self-motion (cf. Figure 8, left).

In the following, we are interested in the transition poses between this one-dimensional plane-symmetric self-motion and the above-mentioned two-dimensional line-symmetric one. Therefore, we only have to intersect the quartic curve Y with $e_0 = 0$, which yields four of these so-called *branching singularities* [40]. These four transition poses are totally flat configurations of the Duporcq hexapod (cf. Table 1, Figure 8, red left and Figure 9).

Remark 6. Note that a further prominent example of a hexapod, which possesses flat poses during its self-motion, is Bricard’s flexible octahedron of type 3 (cf. [41]).

Moreover, it should be mentioned that the Duporcq hexapod is a kinematotropic mechanism (according to the notation of Wohlhart [42]). To the best of the author’s knowledge, only one further hexapod with this property is known so far, which is the so-called Wren platform (see [42] (Section 3) and [21] (Section 2.2)).

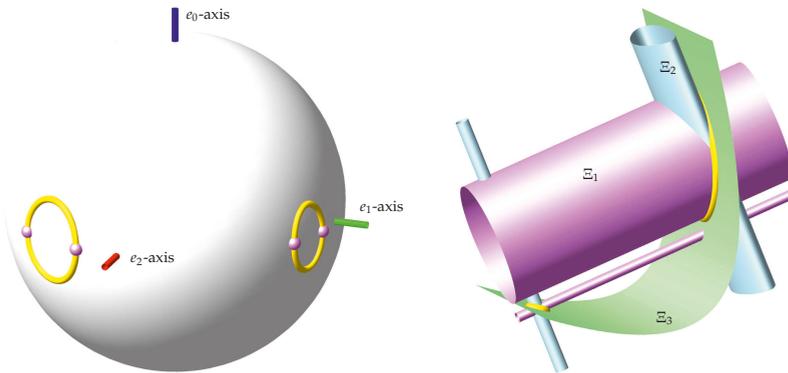


Figure 8. (left): the quartic Y is displayed under consideration of the normalization condition $N = 1$. For the example at hand, it consists of two components (as antipodal points yield the same displacement). Intersection points of the displayed spherical curve with the equator plane yield the branching singularities between plane-symmetric and line-symmetric self-motions. They are numbered from left to right by 1 to 4. (right): visualization of the surfaces Ξ_i under the assumption that $u_0 = 0$ corresponds to the ideal plane. The surface Ξ_i is a cylinder in direction of the u_i -axis (for $i = 1, 2, 3$).

Table 1. The Study parameters of the four flat transition poses illustrated in Figure 9. As they result as roots of a polynomial of degree 4, they can be computed explicitly, but, in order to avoid too long expressions, they are displayed numerically.

Flat Pose	e_1	e_2	f_3
1	-0.63171148011492395006	0.77520358996267041460	0.24434973773984142590
2	-0.26236530678800600560	0.96496862425367773706	0.36840718493416854565
3	0.89932040897259076870	0.43729029489044469464	-1.6168042368274940498
4	0.98317707611585865513	0.18265496708349071532	-2.3876030965525136289

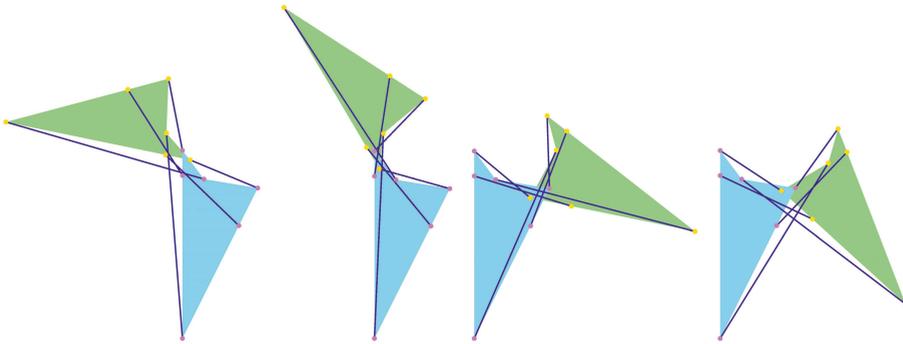


Figure 9. The four flat transition poses numbered from left to right by 1 to 4.

For the example at hand, the fixed axode can be described in the dual representation (If $ax + by + cz + d = 0$ is the equation of the plane of symmetry, then its dual representation is given by the homogenous quadruplet $(u_0 : u_1 : u_2 : u_3) = (d : a : b : c)$ according to [11] (Section 6.2)) as the intersection of the following three surfaces displayed in Figure 8, right:

$$\begin{aligned}
 \Xi_1 : & 178596u_3^2u_2^2 + 45924u_0^4 + 69696u_3^4 + 464508u_2^3u_0 + 293436u_2u_3^2u_0 \\
 & + 573049u_2^2u_0^2 + 124921u_3^2u_0^2 + 108900u_2^4 + 276336u_2u_0^3 = 0, \\
 \Xi_2 : & 108900u_1^4 + 592944u_1^3u_0 + 39204u_3^2u_1^2 + 688345u_1^2u_0^2 + 361152u_3^2u_1u_0 \\
 & - 300932u_1u_0^3 + 18724u_0^4 + 831744u_3^2u_0^2 = 0, \\
 \Xi_3 : & 264u_1^2 + 709u_1u_0 + 198u_2u_1 + 912u_2u_0 + 326u_0^2 = 0.
 \end{aligned} \tag{15}$$

Based on these surfaces, it can be checked (e.g., by computing the Hilbert-polynomial) that the fixed axode corresponds to an algebraic curve of degree 4 in the dual representation. This curve can easily be parametrized as follows (two branches):

$$u_0 = 1, \quad u_2 = -\frac{264u_1^2 + 709u_1 + 326}{6(33u_1 + 152)}, \quad u_3 = \pm \frac{\sqrt{w}}{6(33u_1 + 152)}, \tag{16}$$

with $w = 300932u_1 - 18724 - 688345u_1^2 - 592944u_1^3 - 108900u_1^4$. Moreover it can be seen (cf. Figure 8, right) that the curve has two components. The left one is obtained for $u_1 \in [0.07650139; 0.27046582]$ and the right one for $u_1 \in [-3.17251656; -2.61929914]$. Note that the borders of the two intervals are the roots of w , which can be computed explicitly, but, in order to avoid too long expressions, we displayed them numerically.

Based on the parametrization given in Equation (16), one can easily calculate (cf. [11] (Equation (6.8))) the curve of regression of the fixed axode, which is also displayed in Figure 10.

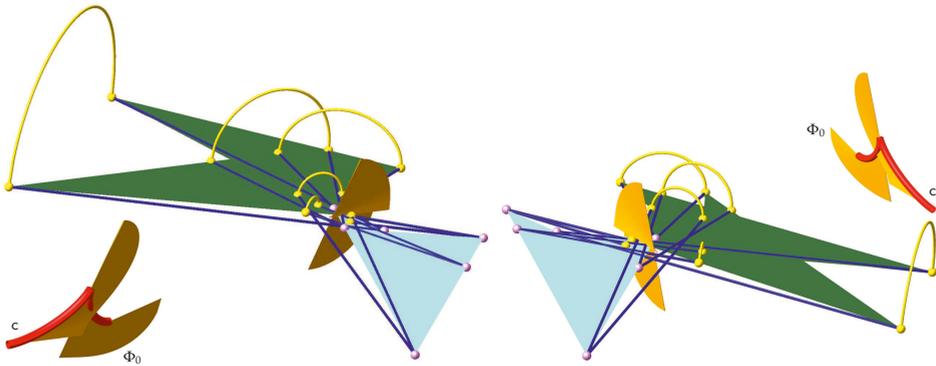


Figure 10. Trajectories of the platform points during the plane-symmetric self-motion between the flat poses 1 and 2 (left) and the flat poses 3 and 4 (right). Moreover, the fixed axodes Φ_0 are displayed, which look like cones upon the first viewing. However, a blow up (in the lower left corner and upper right corner, respectively) of the region of the supposed vertex shows the line of regression c of Φ_0 . For the illustrated self-motion, the tangents of c in the two end points span the carrier plane (xy -plane) of the flat poses. If one considers the complete self-motion, then c has four cusps (obtained by reflecting the illustrated curve at the xy -plane).

4.2. Point-Symmetric Self-Motions of the Duporcq Hexapod

Finally, we want to correct a statement given in [39] (Remark 4), where it is stated that

1. the Duporcq manipulator also has pure translational one-dimensional self-motions,
2. each two-dimensional line-symmetric self-motion of a Duporcq manipulator contains a pure translational one-dimensional sub-self-motion.

The first statement is true in contrast to the second one. In fact, the pure translational self-motion (which can be considered as point-symmetric self-motion) has two branching singularities, where they can switch into a 2-parametric line-symmetric self-motion. This can easily be seen as follows:

For the coordinatisation of the platform points and base points used in Section 3.4, the 1-parametric point-symmetric motion fulfills $e_0 = e_1 = e_2 = f_3 = 0$. It can be computed by expressing f_1 from $\Lambda_2 - \Lambda_1$ (which is linear in f_1). Plugging the resulting expression into Λ_1 implies a homogenous quadratic equation in e_3, f_0, f_2 , which already represents the point-symmetric self-motion. By the additional condition $f_0 = 0$, we obtain the two mentioned branching singularities, which are again totally flat configurations of the Duporcq hexapod (cf. Table 2 and Figure 11).

Table 2. The Study parameters of the two flat transition poses illustrated in Figure 11. As they result as roots of a polynomial of degree 2, they can be computed explicitly, but in order to avoid too long expressions they are again displayed numerically.

Flat Pose	e_3	f_1	f_2
1	1	0.1406805116103807682	-0.2234541534831142304
2	1	2.9246864608666834522	-1.0586559382600050357

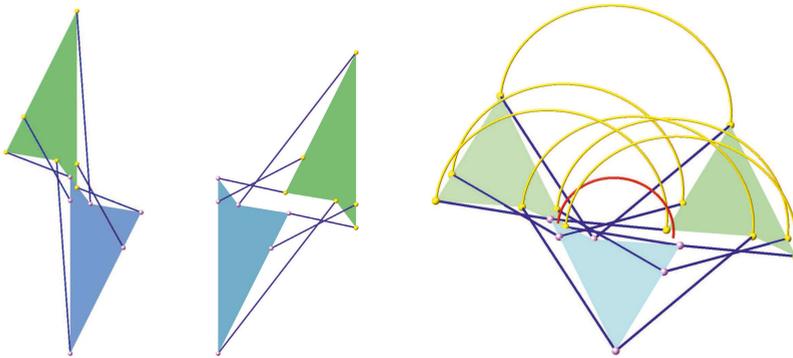


Figure 11. The first and second flat pose (**left** and **center**, respectively) and the translational self-motion between them (**right**). This circular translation can also be seen as a point-symmetric motion, where the corresponding curve (half-circle) is illustrated in red.

Finally it should be noted that there is no branching singularity between plane-symmetric self-motions and point-symmetric self-motions as $e_0 = e_1 = e_2 = e_3 = 0$ has to hold, which contradicts the normalizing condition $N = 1$. Summed up one can say, that the Duporcq hexapod is a twofold kinematotropic mechanism, as there are branching singularities between the two-dimensional line-symmetric self-motion and the one-dimensional

- point-symmetric self-motion,
- plane-symmetric self-motion.

Due to its kinematotropic behavior and its total flat branching singularities the Duporcq manipulator is possibly of interest for the design of deployable structures.

5. Conclusions

This paper gives a complete classification of hexapods with plane-symmetric self-motions. It turns out that besides the planar/spherical symmetric rollings with circular paths and two trivial cases (butterfly self-motion and two-dimensional spherical self-motion), only one further solution exists, which is the so-called Duporcq hexapod. This is the only manipulator possessing plane-symmetric self-motions, which are neither planar nor spherical motions (and therefore also no line-symmetric motions). Moreover, the Duporcq hexapod is may be of interest for the design of deployable structures due to its kinematotropic behavior and total flat branching singularities.

Funding: The author is supported by Grant No. P 24927-N25 of the Austrian Science Fund FWF within the project “Stewart Gough platforms with self-motions”.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Quetelet, L.A.J. Memoire sur une nouvelle maniere de considerer les caustiques, produites soit par reflexion soit par refraction. *Brux. Nouv. Mem.* **1826**, *3*, 89–140. (In French)
2. Tölke, J. Ebene euklidische und sphärische symmetrische Rollungen. *Mech. Mach. Theory* **1978**, *13*, 187–198. (In German) [[CrossRef](#)]
3. Boreis, R. Über die symmetrische Rollung. *Österr. Ing. Arch.* **1953**, *7*, 243–246. (In German)
4. Bottema, O. Characteristic properties of the symmetric plane motion. *Proc. K. Ned. Akad. Wet. B* **1972**, *75*, 145–151.
5. Darboux, G. *Lecons sur la Theorie Generale des Surfaces*; Gauthier-Villars: Paris, France, 1887. (In French)

6. Krames, J. Über Fußpunktkurven von Regelflächen und eine besondere Klasse von Raumbewegungen (Über symmetrische Schrotungen I). *Monatsh. Math. Phys.* **1937**, *45*, 394–406. (In German) [[CrossRef](#)]
7. Selig, J.M.; Husty, M. Half-turns and line symmetric motions. *Mech. Mach. Theory* **2011**, *46*, 156–167. [[CrossRef](#)]
8. Bottema, O.; Roth, B. *Theoretical Kinematics*; North-Holland: Amsterdam, The Netherlands, 1979.
9. Gallet, M.; Nawratil, G.; Schicho, J.; Selig, J.M. Mobile Icosapods. *Adv. Appl. Math.* **2017**, *88*, 1–25. [[CrossRef](#)]
10. Hernandez-Gutierrez, I. Screw Surfaces in the Analysis and Synthesis of Mechanisms. Ph.D. Thesis, Stanford University, Stanford, CA, USA, 1989.
11. Pottmann, H.; Wallner, J. *Computational Line Geometry*; Springer: Berlin/Heidelberg, Germany, 2001.
12. Kunze, S.; Stachel, H. Über ein sechsgliedriges räumliches Getriebe. *Elem. Math.* **1974**, *29*, 25–32. (In German)
13. Baker, J.E. The single screw reciprocal to the general plane-symmetric six-screw linkage. *J. Geom. Graph.* **1997**, *1*, 5–12.
14. Husty, M.E. Borel's and R. Bricard's Papers on Displacements with Spherical Paths and their Relevance to Self-Motions of Parallel Manipulators. In *Proceedings of the International Symposium on History of Machines and Mechanisms—HMM 2000*; Ceccarelli, M., Ed.; Kluwer: Dordrecht, The Netherlands, 2000; pp. 163–171.
15. Nawratil, G. Correcting Duporcq's theorem. *Mech. Mach. Theory* **2014**, *73*, 282–295. [[CrossRef](#)] [[PubMed](#)]
16. Karger, A. Singularities and self-motions of equiform platforms. *Mech. Mach. Theory* **2001**, *36*, 801–815. [[CrossRef](#)]
17. Karger, A. Singularities and self-motions of a special type of platforms. In *Advances in Robot Kinematics: Theory and Applications*; Lenarcic, J., Thomas, F., Eds.; Kluwer: Dordrecht, The Netherlands, 2002; pp. 155–164.
18. Karger, A. Parallel manipulators with simple geometrical structure. In *Proceedings of the EUCOMES 08: The Second European Conference on Mechanism Science*; Ceccarelli, M., Ed.; Springer: Dordrecht, The Netherlands, 2008; pp. 463–470.
19. Nawratil, G. Non-existence of planar projective Stewart Gough platforms with elliptic self-motions. In *Computational Kinematics: Proceedings of the 6th International Workshop on Computational Kinematics (CK2013)*; Thomas, F., Perez Garcia, A., Eds.; Springer: Dordrecht, The Netherlands, 2013; pp. 49–57.
20. Nawratil, G. On equiform Stewart Gough platforms with self-motions. *J. Geom. Graph.* **2013**, *17*, 163–175.
21. Nawratil, G. Congruent Stewart Gough platforms with non-translational self-motions. In *Proceedings of the 16th International Conference on Geometry and Graphics*; Schröcker, H.-P., Husty, M., Eds.; Innsbruck University Press: Innsbruck, Austria, 2014; pp. 204–215.
22. Nawratil, G. On the Self-Mobility of Point-Symmetric Hexapods. *Symmetry* **2014**, *6*, 954–974. [[CrossRef](#)]
23. Dietmaier, P. Forward kinematics and mobility criteria of one type of symmetric Stewart-Gough platforms. In *Recent Advances in Robot Kinematics*; Lenarcic, J., Parenti-Castelli, V., Eds.; Kluwer: Dordrecht, The Netherlands, 1996; pp. 379–388.
24. Karger, A.; Husty, M. Classification of all self-motions of the original Stewart-Gough platform. *Comput. Aided Des.* **1998**, *30*, 205–215. [[CrossRef](#)]
25. Nawratil, G. Self-motions of parallel manipulators associated with flexible octahedra. In *Proceedings of the Austrian Robotics Workshop*; Hofbauer, M., Husty, M., Eds.; Umit Lecture Notes: Hall in Tyrol, Austria, 2011; pp. 232–248.
26. Nawratil, G. Planar Stewart Gough platforms with a type II DM self-motion. *J. Geom.* **2011**, *102*, 149–169. [[CrossRef](#)]
27. Husty, M.L.; Karger, A. Self motions of Stewart-Gough platforms, an overview. In *Proceedings of the Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators*, Quebec City, QC, Canada, 2–3 October 2002; Gosselin, C.M., Ebert-Uphoff, I., Eds.; pp. 131–141.
28. Nawratil, G. Introducing the theory of bonds for Stewart Gough platforms with self-motions. *ASME J. Mech. Rob.* **2014**, *6*. [[CrossRef](#)]
29. Gallet, M.; Nawratil, G.; Schicho, J. Liaison Linkages. *J. Symb. Comput.* **2017**, *79*, 65–98. [[CrossRef](#)]
30. Borel, E. Mémoire sur les déplacements à trajectoires sphériques. In *Mém. Présent. Var. Sci. Acad. Sci. Natl. Inst. Fr. TOME XXXIII*; Imprimerie Nationale: Paris, France, 1908; pp. 1–128. (In French)
31. Bricard, R. Mémoire sur les déplacements à trajectoires sphériques. *J. École Polytech.* **1906**, *11*, 1–96. (In French)
32. Karger, A. New Self-Motions of Parallel Manipulators. In *Advances in Robot Kinematics: Analysis and Design*; Lenarcic, J., Wenger, P., Eds.; Springer: Dordrecht, The Netherlands, 2008; pp. 275–282.
33. Karger, A. Self-motions of Stewart-Gough platforms. *Comput. Aided Geom. Des.* **2008**, *25*, 775–783. [[CrossRef](#)]

34. Figliolini, G.; Angeles, J. The Spherical Equivalent of Bresse's Circles: The Case of Crossed Double-Crank Linkages. *ASME J. Mech. Rob.* **2014**, *9*. [[CrossRef](#)]
35. Glaeser, G.; Stachel, H.; Odehnal, B. *The Universe of Conics*; Springer Spektrum: Berlin/Heidelberg, Germany, 2016.
36. Husty, M.L. An algorithm for solving the direct kinematics of general Stewart-Gough platforms. *Mech. Mach. Theory* **1996**, *31*, 365–380. [[CrossRef](#)]
37. Borras, J.; Thomas, F.; Torras, C. On Delta Transforms. *IEEE Trans. Robot. Autom.* **2009**, *25*, 1225–1236. [[CrossRef](#)]
38. Duporcq, E. Sur un remarquable déplacement à deux paramètres. *Bull. Soc. Math. Fr.* **1901**, *29*, 1–4. (In French)
39. Nawratil, G.; Schicho, J. Duporcq Pentapods. *ASME J. Mech. Rob.* **2017**, *9*. [[CrossRef](#)]
40. Gogu, G. Branching singularities in kinematotropic parallel mechanisms. In *Computational Kinematics: Proceedings of the 5th International Workshop on Computational Kinematics*; Kecskemethy, A., Müller, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 341–348.
41. Stachel, H. Flexible Polyhedral Surfaces With Two Flat Poses. *Symmetry* **2015**, *7*, 774–787. [[CrossRef](#)]
42. Wohlhart, K. Kinematotropic linkages. In *Recent Advances in Robot Kinematics*; Lenarcic, J., Parenti-Castelli, V., Eds.; Kluwer: Dordrecht, The Netherlands, 1996; pp. 359–368.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Kinematics Analysis of a Class of Spherical PKMs by Projective Angles

Giovanni Legnani ^{1,2,*} and Irene Fassi ²

¹ Department of Mechanical and Industrial Engineering, University of Brescia, 25123 Brescia, Italy

² Institute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing, National Research Council, 20133 Milan, Italy; irene.fassi@cnr.it

* Correspondence: giovanni.legnani@unibs.it; Tel.: +39-030-3715-425

Received: 4 July 2018; Accepted: 17 September 2018; Published: 20 September 2018

Abstract: This paper presents the kinematics analysis of a class of spherical PKMs Parallel Kinematics Machines exploiting a novel approach. The analysis takes advantage of the properties of the projective angles, which are a set of angular conventions of which their properties have only recently been presented. Direct, inverse kinematics and singular configurations are discussed. The analysis, which results in the solution of easy equations, is developed at position, velocity and acceleration level.

Keywords: projective angles; spherical PKM; direct kinematics; inverse kinematics

1. Introduction

Spherical PKMs Parallel Kinematics Machines are a class of parallel manipulators whose mobile platform may rotate around one fixed point. They typically have 3 DOF Degrees of Freedom. They have been studied for a long time [1,2].

It is possible to identify two big classes of spherical PKMs. In the first one (class A), the spherical motion is guaranteed by the convergence in one point of the rotation axes of all of the joints. In the PKM of the second class (class B), the mobile platform is connected to the ground by a spherical or a universal joint that is placed in the center of rotation and the motion is transmitted to it by external legs that may have revolute joints of which the axes do not pass for the center of the rotations.

Among the spherical PKMs belonging to the first class, one of the most popular is the agile eye, which was firstly proposed by Gosselin and was then also adopted by others [2–7]. Its structure is presented in Figure 1. It has three identical legs, each with three revolute joints which have axes that all converge in the same point around which the end-effector rotates. This PKM exhibits an excellent rotational ability and it can assume a compact structure (Figure 2). However, as discussed in the following sections, this PKM is over constrained and, thus, requires high manufacturing and assembly precision to ensure a correct motion without overloading the structure with undesired internal stresses.

The properties of the projective angles can be used to describe the 3D rotation motion of any rigid body and, in this sense, are general. It is a particular angular convention with pros and cons, as with any other convention (Euler angles, quaternions . . .).

The application to PKM is particularly convenient to the class of spherical manipulators based on rotational joints forming cardan sequences, because in this case, the angles of the angular convention coincide with the joint angles of the PKM.

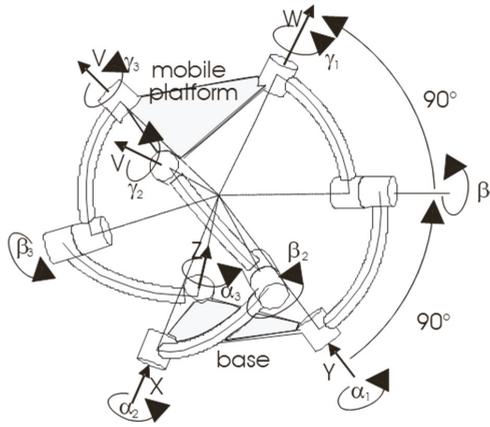


Figure 1. The kinematic structure of the agile eye.

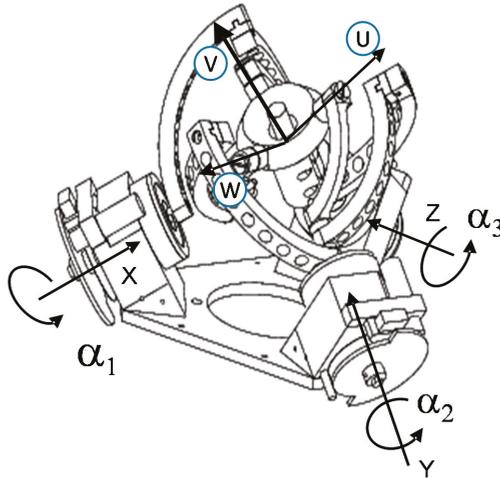


Figure 2. The agile eye (Adapted from [7]).

One example of the second class is the 3-DOF 3SPS/S that is presented in [8–11], which has three legs with spherical-prismatic-spherical-joints and one additional spherical joint connecting the mobile platform to the fixed base. Another example is described in [12,13], and it is shown in Figure 3. This PKM (better described in the following paragraphs of this paper) can be considered a non-overconstrained version of the agile-eye. The legs are serial kinematic chains for which standard methodologies for their analysis are available [14,15].

Several approaches have been developed for the synthesis of the different spherical PKMs. For example, Kong and Gosselin [16,17] proposed the adoption of the screw theory. Fang and Tsai [18] developed a family of spherical PKMs with legs of identical structure, while Karouia and Hervé [19] developed spherical PKMs with legs with different structures. A list with the classification of different spherical manipulators has been suggested by Hess-Coelho [20], which includes a methodology to evaluate their performances. Many other parallel orientation mechanisms have been described in numerous papers (e.g., [3,21–31]). Generally, the first joint of each leg is actuated, however some manipulators adopt transmissions based on parallelograms [31,32].

Analogous concepts were used to design non-redundant PKM with decoupled rotational and translational motion [33] and 2D orientating mechanisms [34,35].

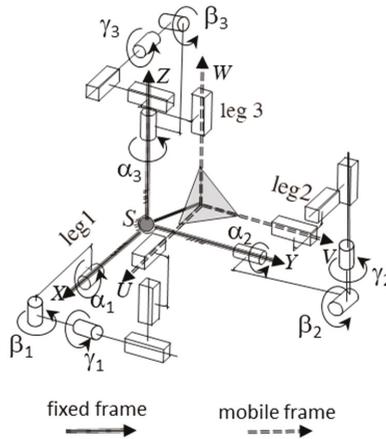


Figure 3. A non-overconstrained version of the agile-eye.

The concept of projective angles has been introduced by [36] and represents a useful approach for the kinematic analysis of PKMs. In [12], this concept has been applied to the angular position analysis of a non-overconstrained variation of the agile eye.

The present paper extends the abovementioned approach to the full kinematic analysis of some spherical PKMs, resulting in a set of easily solvable equations, at position, velocity and acceleration level. Specifically, the proposed approach will be used to solve the direct and inverse kinematics of spherical manipulators, belonging to the two mentioned classes (class A, B).

2. Materials and Methods

2.1. The Agile Eye

The 3RRR agile eye (Figures 1 and 2) is a particular version of a spherical PKM in which the axes of the rotational joints of each link form an angle of 90° and they all converge in one point, which is the centre of rotation.

The mobility of the agile-eye can be studied by the well know Grubler-Kutzbach formula. Having seven bodies and thus, a total of 42 DoF (degrees of freedom) and nine revolute joints for a total of 45 constrains, this means that there are six redundant constrains since the PKM clearly has 3 DoF.

The motion is possible because all of the joint axes converge in one point. The redundant constraints can be removed in several ways by changing the nature of the joints. Recently, a non-overconstrained version of the agile-eye (Figure 3) was presented in [12].

The mobile platform is connected to the fixed base by three legs having a series of three revolute and three prismatic joints. The first revolute joint of each leg is actuated; in addition, a spherical joint is also present and connects the fixed base and the mobile platform. The PKM is of type 3RRRPPP/S. The joint actuator coordinates of this PKM are the rotations of the actuators that are connected to the first joint of each leg:

$$Q = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \tag{1}$$

where the subscript $i = 1, 2, 3$ indicates the leg. The rotations of the non-actuated joints are represented by the angles β_i and γ_i . The choice of the name for the angles α_i and β_i , which is identical to those that

we will use for the definitions of the projective angles (see the next section), is justified by the analogies that will be discussed further in this paper.

This PKM can be considered a variant of the agile-eye because the sequences of the revolute joints are the same. The prismatic joints remove the over constraints by leaving three additional translation degrees of freedom which are removed by the spherical joint.

The direct and the inverse kinematics studies the relation between the attitude of the mobile platform and the rotation of the joint actuators, and since the sequence of the joints is the same in the two platforms, their equation may be written in a unified way. Recently, a solution of the position analysis of the PKM of Figure 3 which was based on the concept of the projective angles was presented [12]. Moreover, the possibility to also use projective angles for velocity and acceleration analysis has been proposed in [13]. By joining these results, it is possible to propose a methodology for a full kinematic analysis of the presented class of spherical PKM.

2.2. The Projective Angles

The angular position (attitude) of one body may be represented by the rotation matrix expressing the angular position of a frame that is attached on the body with respect to a fixed frame.

We indicate by X, Y and Z the fixed axes, and by U, V and W the mobile axes (Figures 4–6). The rotation matrix is then:

$$\mathbf{R} = [\mathbf{u}, \mathbf{v}, \mathbf{w}] = \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & & w_z \end{bmatrix} \tag{2}$$

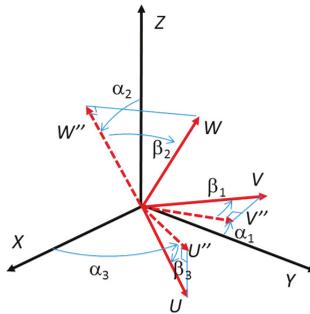


Figure 4. Definition of the projective angles α_i and of the auxiliary angles β_i .

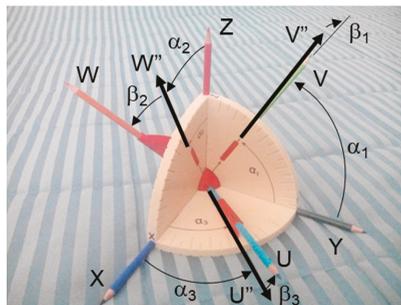


Figure 5. A 3D model to show the projective angles α_i and of the auxiliary angles β_i .

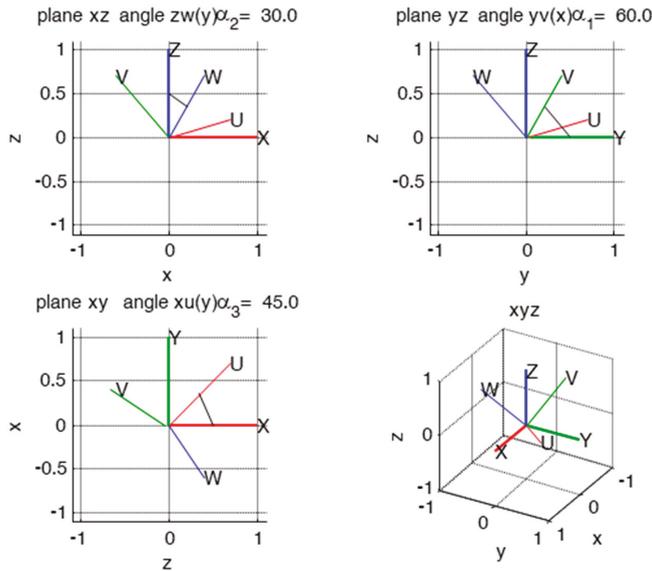


Figure 6. Visualization of the projective angles on the xy, yz and xz plane; in this example, the projective angles assume the value: 60°, 30°, 45°.

The projective angles α_1 , α_2 and α_3 are defined as follows (Figures 5 and 6):

1. project the unit vector v of V on plane YZ obtaining vector v'' , α_1 is the angle between v'' and Y ;
2. project the unit vector w of W on plane XZ obtaining vector w'' , α_2 is the angle between w'' and Z ;
3. project the unit vector u of U on plane XY obtaining vector u'' , α_3 is the angle between u'' and X .

The definition of the following auxiliary angles β_1 , β_2 and β_3 is also necessary for the discussion (Figure 4).

1. β_1 is the angle between V and the plane YZ
2. β_2 is the angle between W and the plane XZ
3. β_3 is the angle between U and the plane XY

and so, the angular position of the mobile frame is:

$$\mathbf{R} = \begin{bmatrix} \mathbf{U} & \mathbf{V} & \mathbf{W} \end{bmatrix} = \begin{bmatrix} \cos \alpha_3 \cos \beta_3 & -\sin \beta_1 & \sin \alpha_2 \cos \beta_2 \\ \sin \alpha_3 \cos \beta_3 & \cos \alpha_1 \cos \beta_1 & -\sin \beta_2 \\ -\sin \beta_3 & \sin \alpha_1 \cos \beta_1 & \cos \alpha_2 \cos \beta_2 \end{bmatrix} \quad (3)$$

It is important to note that, by construction, $\cos(\beta_i) > 0$.

The projective angles of the mobile frame may be extracted from the rotation matrix \mathbf{R} as:

$$\mathbf{A} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \begin{bmatrix} \text{atan2}(v_z, v_y) \\ \text{atan2}(w_x, w_z) \\ \text{atan2}(u_y, u_x) \end{bmatrix} \quad (4)$$

where $\text{atan2}(a,b)$ is the four quadrant extension of the arctangent function $\text{atan}(a/b)$. The possibility of using atan2 rather than other inverse trigonometric function (e.g., atan) eliminates the ambiguity of the identification of the correct quadrant and the risk of division by 0.

Therefore, knowing the angular position of the mobile frame (i.e., the rotation matrix \mathbf{R}), the projective angles may be easily evaluated. The inverse operation (i.e., determining \mathbf{R} when the projective angles are known) will be discussed further in this paper (Section 2.6).

2.3. Inverse Kinematics of the Spherical PKM

To study the mentioned spherical PKM, a fixed frame and a mobile frame were established, as shown in Figures 1–3. To simplify the analysis, the coordinate axes are chosen coincidentally to the rotation axis of some revolute joints. The XYZ axis of the fixed base frame are parallel with the axes of the actuated joints, while the UVW axes of the mobile frame are parallel with the axis of the joints γ_i of the mobile frame. The inverse kinematics of the considered spherical PKM can be solved by realizing that the joints of each leg constitute a Cardanic sequence. With reference to Figures 3 and 4, we get:

leg 1: rotations $\alpha_1\beta_1\gamma_1$ around the following axis sequence XZY,

leg 2: rotations $\alpha_2\beta_2\gamma_2$ around YXZ,

leg 3: rotations $\alpha_3\beta_3\gamma_3$ around ZYX.

The rotation matrix \mathbf{R} expressing the attitude of the mobile platform can easily be evaluated by considering each different leg. For leg 1, we got:

$$\mathbf{R} = \mathbf{R}_1 = \begin{bmatrix} \cos \beta_1 \cos \gamma_1 & -\sin \beta_1 & \cos \beta_1 \sin \gamma_1 \\ \cos \alpha_1 \sin \beta_1 \cos \gamma_1 + \sin \alpha_1 \sin \gamma_1 & \cos \alpha_1 \cos \beta_1 & \cos \alpha_1 \sin \beta_1 \sin \gamma_1 - \sin \alpha_1 \cos \gamma_1 \\ \sin \alpha_1 \sin \beta_1 \cos \gamma_1 - \cos \alpha_1 \sin \gamma_1 & \sin \alpha_1 \cos \beta_1 & \sin \alpha_1 \sin \beta_1 \sin \gamma_1 + \cos \alpha_1 \cos \gamma_1 \end{bmatrix} \quad (5)$$

and similar equations may be written for legs 2 and 3, which lead to the matrices \mathbf{R}_2 and \mathbf{R}_3 . By indicating the rotation matrix as:

$$\mathbf{R} = \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (6)$$

the joint rotations can easily be determined, obtaining two solutions for each leg.

For leg 1 we got:

$$\begin{cases} \alpha_{11} = \text{atan2}(r_{32}, r_{22}) \\ \beta_{11} = \text{asin}(-r_{12}) \\ \gamma_{11} = \text{atan2}(r_{13}, r_{11}) \end{cases}, \begin{cases} \alpha_{12} = \text{atan2}(-r_{32}, -r_{22}) \\ \beta_{12} = \pi - \text{asin}(-r_{12}) \\ \gamma_{12} = \text{atan2}(-r_{13}, -r_{11}) \end{cases} \quad (7)$$

and similar equations may be written for legs 2 and 3. The relations between the two solutions for the i -th leg is:

$$\begin{cases} \alpha_{i2} = \alpha_{i1} + \pi \\ \beta_{i2} = \pi - \beta_{i1} \\ \gamma_{i2} = \gamma_{i1} + \pi \end{cases} \quad (8)$$

and the inverse kinematics is therefore solved by selecting the equations that were associated with the first joint of each leg:

$$\mathbf{Q} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \begin{bmatrix} \text{atan2}(v_z, v_y) + k_1\pi \\ \text{atan2}(w_x, w_z) + k_2\pi \\ \text{atan2}(u_y, u_x) + k_3\pi \end{bmatrix} \quad k_i = 0, 1 \quad (9)$$

A total of 8 possible solutions were found. Kinematic singularities are present for the configurations for which the atan2 function is not defined, i.e., atan2(0,0) which happens when the second joint angle of one leg is ± 90 degrees and so $\cos(\beta_i) = 0$.

2.4. Direct Kinematics

To solve the direct kinematics (find the matrix \mathbf{R} when the joint coordinates α_i are known), we can write the rotation matrix of the mobile platform by choosing one column for each of the matrix of each leg (Equation (5) for leg 1 and similar equations for the other legs). More precisely, we can bring the second column from R_1 , the third column from R_2 and the first one from R_3 .

$$\mathbf{R} = \begin{bmatrix} \mathbf{U} & \mathbf{V} & \mathbf{W} \end{bmatrix} = \begin{bmatrix} \cos \alpha_3 \cos \beta_3 & -\sin \beta_1 & \sin \alpha_2 \cos \beta_2 \\ \sin \alpha_3 \cos \beta_3 & \cos \alpha_1 \cos \beta_1 & -\sin \beta_2 \\ -\sin \beta_3 & \sin \alpha_1 \cos \beta_1 & \cos \alpha_2 \cos \beta_2 \end{bmatrix} \quad (10)$$

assuming $\cos(\beta_i) \neq 0$ and dividing each column by the corresponding $\cos(\beta_i)$, we got:

$$\mathbf{R}' = \begin{bmatrix} \mathbf{U}' & \mathbf{V}' & \mathbf{W}' \end{bmatrix} = \begin{bmatrix} \cos \alpha_3 & -a' & \sin \alpha_2 \\ \sin \alpha_3 & \cos \alpha_1 & -b' \\ -c' & \sin \alpha_1 & \cos \alpha_2 \end{bmatrix} \quad (11)$$

with $a' = \text{tg}\beta_1$, $b' = \text{tg}\beta_2$ and $c' = \text{tg}\beta_3$.

Since we are considering the direct kinematics, the angles α_i are given, while a' , b' and c' can be computed by the conditions of orthogonality of the three columns:

$$\mathbf{V}' \cdot \mathbf{W}' = 0, \mathbf{W}' \cdot \mathbf{U}' = 0, \mathbf{U}' \cdot \mathbf{V}' = 0 \quad (12)$$

that can be expressed as:

$$\begin{cases} a' \sin \alpha_2 + b' \cos \alpha_1 - \cos \alpha_2 \sin \alpha_1 = 0 \\ b' \sin \alpha_3 + c' \cos \alpha_2 - \cos \alpha_3 \sin \alpha_2 = 0 \\ c' \sin \alpha_1 + a' \cos \alpha_3 - \cos \alpha_1 \sin \alpha_3 = 0 \end{cases} \quad (13)$$

that in matrix form is:

$$\begin{bmatrix} \sin \alpha_2 & \cos \alpha_1 & 0 \\ 0 & \sin \alpha_3 & \cos \alpha_2 \\ \cos \alpha_3 & 0 & \sin \alpha_1 \end{bmatrix} \begin{bmatrix} a' \\ b' \\ c' \end{bmatrix} = \begin{bmatrix} \cos \alpha_2 \sin \alpha_1 \\ \cos \alpha_3 \sin \alpha_2 \\ \cos \alpha_1 \sin \alpha_3 \end{bmatrix} \quad (14)$$

which is a linear system with respect to a' , b' and c' , of which the solution is:

$$\begin{cases} a' = \frac{\cos \alpha_2 \sin \alpha_3 - \cos \alpha_1 \cos \alpha_3 \sin \alpha_2 \sin \alpha_1}{\Delta} \\ b' = \frac{\cos \alpha_3 \sin \alpha_1 - \sin \alpha_2 \cos \alpha_1 \sin \alpha_3 \cos \alpha_2}{\Delta} \\ c' = \frac{\cos \alpha_1 \sin \alpha_2 - \cos \alpha_3 \cos \alpha_2 \sin \alpha_1 \sin \alpha_3}{\Delta} \end{cases} \quad (15)$$

where $\Delta = \cos \alpha_3 \cos \alpha_2 \cos \alpha_1 + \sin \alpha_3 \sin \alpha_2 \sin \alpha_1$, which is the determinant of the linear system that must not be zero for the invertibility of the matrix. Moreover, the determinant must be positive ($\Delta > 0$) because the mobile frame is right (for $\Delta < 0$ we get a left frame, while $\Delta = 0$ results in a singular configuration).

From Equation (15), the value of the angles β_i is immediately found:

$$\beta_1 = \text{atan}(a') + k_1\pi, \beta_2 = \text{atan}(b') + k_2\pi, \beta_3 = \text{atan}(c') + k_3\pi \quad (16)$$

with $k_i = 0, 1$ and so the direct kinematics has 8 different solutions and matrix \mathbf{R} (Equation (10)) is determined.

By only considering the right frames, the singular configurations happened for the following angular position of the mobile platform:

$$\mathbf{R} = \mathbf{R}_a = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (17)$$

$$\mathbf{R} = \mathbf{R}_b = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix} \quad (18)$$

$$\mathbf{R} = \mathbf{R}_c = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -1 \\ -1 & 0 & 0 \end{bmatrix} \quad (19)$$

$$\mathbf{R} = \mathbf{R}_d = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{bmatrix} \quad (20)$$

which happens for $\cos(\beta_i) = 0$ and $\sin(\beta_i) = \pm 1$. For these configurations, the actuated joint angles α_i may assume any value.

2.5. Projective Angles and Spherical PKM

An analysis of the definition of the projective angles and of the kinematics of the considered spherical PKM highlights several analogies.

It is worth noting that Equations (4) and (9) are very similar and that they coincide for $k_i = 0$. It is therefore possible to conclude that if for the PKM we choose the solution with $\cos(\beta_i) > 0$, the joint coordinates of the PKM coincide with the projective angles of the mobile platform.

Similarly, the procedure that was adopted to solve the direct kinematics of the PKM can be adopted to compute the angular position of one frame which corresponds to an assigned value of the projective angles α_i . In this case, however, we must always assume $\cos(\beta_i) > 0$, and so the solution is unique.

Finally, if we describe the angular position of the mobile platform using the projective angles A , their relation to the joint coordinates Q is:

$$A = Q + \begin{bmatrix} k_1\pi \\ k_2\pi \\ k_3\pi \end{bmatrix}, \quad k_i = 0, 1 \quad (21)$$

and so

$$J = \frac{\partial A}{\partial Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (22)$$

According to the above equations, and adopting the projective angle convention, the PKM can be considered ‘decoupled’, in the sense that each actuator influences just one of the end-effector coordinates [37]. Therefore, for $k_i = 0$, the joint coordinates of the PKM coincide with the projective angles of the mobile frame and also the angle β_i that was used in the description of the projective angles and in the spherical PKM (2nd joint of each leg) have the same values. If the 2nd solution is chosen for the PKM, its angles α_i and β_i differ from those of the projective angles by 180° (see Equation (21) for analogy). Of course, due to the non-integrability of the angular velocity vector Ω it is:

$$\dot{A} = \dot{Q} \text{ but } \Omega \neq \dot{Q} \quad (23)$$

because it is impossible to define a set of angular coordinates representing a 3D orientation of one body which has a time derivative that coincides with its angular velocity Ω [37–39].

A broader analysis of velocity and acceleration is found in the next sections. The analysis will be performed with reference to the abovementioned PKMs; by remembering Equation (21), the results are immediately extended to the projective angles.

We may conclude that the projective angles are a sort of “intrinsic” notation to study the considered spherical PKMs.

2.6. From Projective Angle to Rotation Matrix

Considering the analogies between the kinematic of the spherical PKM and the projective angles, the rotation matrix R that is associated with a set of projective angles A is found by Equations (10) and (16) assuming $k_1 = k_2 = k_3 = 0$ since, by definition of the projective angles, the cosine of the angles β_i are positive.

2.7. Velocity Analysis

Since the legs of the PKM form a Cardanic sequence, the velocity analysis is easily developed. The solution proposed in the following can be applied both to the spherical PKM under study and to the projective angles.

By considering the different legs, we have a first rotation α around one Cartesian axis, a second β rotation around an axis rotated by α and, finally, a third rotation γ around one axis which depends on α and β . For instance, for leg 1, the three rotation axes are defined by the following unit vectors $a_{11} = [1\ 0\ 0]^t$, $a_{12} = [0\ -\sin\alpha_1\ \cos\alpha_1]^t$ and $a_{13} = [-\sin\beta_1\ \cos\alpha_1\ \cos\beta_1\ \sin\alpha_1\ \cos\beta_1]^t$; therefore, for the three legs, and representing the angular velocity of the frame by $\Omega = [\omega_x\ \omega_y\ \omega_z]^T$, we obtain the following results

$$\Omega = a_{11}\dot{\alpha}_1 + a_{12}\dot{\beta}_1 + a_{13}\dot{\gamma}_1 \tag{24}$$

And so, for leg 1, it is:

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin\beta_1 \\ 0 & -\sin\alpha_1 & \cos\alpha_1\cos\beta_1 \\ 0 & \cos\alpha_1 & \sin\alpha_1\cos\beta_1 \end{bmatrix} \begin{bmatrix} \dot{\alpha}_1 \\ \dot{\beta}_1 \\ \dot{\gamma}_1 \end{bmatrix} \tag{25}$$

and similar equations may be written for legs 2 and 3. The above presented equations may be inverted as:

$$\begin{bmatrix} \dot{\alpha}_1 \\ \dot{\beta}_1 \\ \dot{\gamma}_1 \end{bmatrix} = \begin{bmatrix} 1 & \cos\alpha_1\frac{\sin\beta_1}{\cos\beta_1} & \sin\alpha_1\frac{\sin\beta_1}{\cos\beta_1} \\ 0 & -\sin\alpha_1 & \cos\alpha_1 \\ 0 & \frac{\cos\alpha_1}{\cos\beta_1} & \frac{\sin\alpha_1}{\cos\beta_1} \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \tag{26}$$

By considering the first row of the matrix of Equation (26) and the analogous equations for leg 2 and 3, the velocity of the actuated joints is then obtained by the angular velocity of the mobile platform:

$$\dot{Q} = \begin{bmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \\ \dot{\alpha}_3 \end{bmatrix} = \begin{bmatrix} 1 & \cos\alpha_1\frac{\sin\beta_1}{\cos\beta_1} & \sin\alpha_1\frac{\sin\beta_1}{\cos\beta_1} \\ \sin\alpha_2\frac{\sin\beta_2}{\cos\beta_2} & 1 & \cos\alpha_2\frac{\sin\beta_2}{\cos\beta_2} \\ \cos\alpha_3\frac{\sin\beta_3}{\cos\beta_3} & \sin\alpha_3\frac{\sin\beta_3}{\cos\beta_3} & 1 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = C\Omega \tag{27}$$

while the angular velocity of the end-effector is obtained by the angular velocity of the actuators by inverting the equation:

$$\Omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} -\frac{\cos \alpha_2 \sin \alpha_3}{\Delta \tan \beta_1} & \frac{\sin \alpha_1 \sin \alpha_3}{\Delta \tan \beta_2} & \frac{\cos \alpha_1 \cos \alpha_2}{\Delta \tan \beta_3} \\ \frac{\cos \alpha_2 \cos \alpha_3}{\Delta \tan \beta_1} & -\frac{\cos \alpha_3 \sin \alpha_1}{\Delta \tan \beta_2} & \frac{\sin \alpha_1 \sin \alpha_2}{\Delta \tan \beta_3} \\ \frac{\sin \alpha_2 \sin \alpha_3}{\Delta \tan \beta_1} & \frac{\cos \alpha_1 \cos \alpha_3}{\Delta \tan \beta_2} & -\frac{\cos \alpha_1 \sin \alpha_2}{\Delta \tan \beta_3} \end{bmatrix} \begin{bmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \\ \dot{\alpha}_3 \end{bmatrix} = C^{-1} \dot{Q} \quad (28)$$

where $\Delta = \cos \alpha_3 \cos \alpha_2 \cos \alpha_1 + \sin \alpha_3 \sin \alpha_2 \sin \alpha_1$.

Equations (27) and (28) are defined for all of the non-singular configurations (for direct or inverse kinematics), which implies:

$$\Delta \neq 0, \cos \beta_i \neq 0 \text{ and } \tan \beta_i \neq 0 \quad (29)$$

2.8. Acceleration Analysis

The acceleration analysis may be developed by starting from the velocity relation Equation (27) which synthetically reads:

$$\dot{Q} = C\Omega \quad (30)$$

by a derivation with respect to the time, we found that the joint actuators acceleration was necessary to give an angular acceleration $\dot{\Omega}$ to the end-effector:

$$\ddot{Q} = \dot{C}\Omega + C\dot{\Omega} \quad (31)$$

and inversely, the angular acceleration of the mobile platform is:

$$\dot{\Omega} = C^{-1}(\ddot{Q} - \dot{C}\Omega) \quad (32)$$

With

$$\dot{C} = \sum_{i=1}^3 \frac{\partial C}{\partial \alpha_i} \dot{\alpha}_i + \sum_{i=1}^3 \frac{\partial C}{\partial \beta_i} \dot{\beta}_i = C_a + C_b \quad (33)$$

where C_a and C_b are:

$$C_a = \begin{bmatrix} 0 & -\dot{\alpha}_1 \sin \alpha_1 \tan \beta_1 & \dot{\alpha}_1 \cos \alpha_1 \tan \beta_1 \\ \dot{\alpha}_2 \cos \alpha_2 \tan \beta_2 & 0 & -\dot{\alpha}_2 \sin \alpha_2 \tan \beta_2 \\ -\dot{\alpha}_3 \sin \alpha_3 \tan \beta_3 & \dot{\alpha}_3 \cos \alpha_3 \tan \beta_3 & 0 \end{bmatrix} \quad (34)$$

The time derivatives of the angles β_i to be inserted in C_b are easily obtained from Equation (26) and those for legs 2 and 3 as:

$$\begin{bmatrix} \dot{\beta}_1 \\ \dot{\beta}_2 \\ \dot{\beta}_3 \end{bmatrix} = \begin{bmatrix} 0 & -\sin \alpha_1 & \cos \alpha_1 \\ \cos \alpha_2 & 0 & -\sin \alpha_2 \\ -\sin \alpha_3 & \cos \alpha_3 & 0 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (35)$$

The proposed solution can be applied both to the spherical PKM under study and to the projective angles.

3. Results

This paper has presented a methodology for the full kinematics analysis of a class of spherical PKMs. This methodology takes advantage of the properties of the projective angles for which the analysis is extended to velocity and acceleration. All of the solutions are found and the singular cases are discussed.

4. Discussion

The properties of the projective angles can be used to describe the 3D rotation motion of any rigid body and, in this sense, this notation is general. It is a particular angular convention with pros and cons, as with any other convention (Euler angles, quaternions . . .). The application to PKM is particularly convenient to the class of spherical manipulators based on rotational joints forming cardan sequences because, in this case, the angles of the angular convention coincide with the joint angles of the PKM. In this context, the paper highlights the numerous analogies between the direct and inverse kinematics of the PKM and the relations between the rotation matrix describing the orientation of the end-effector and the projective angles of the frame that were associated with it. We may conclude that the projective angles are a sort of “intrinsic” notation to study the considered spherical PKMs.

The paper solves this relation for angular position, velocity and acceleration.

The proposed methodology is an alternative solution to those that were proposed by classical papers.

The inverse kinematics analysis of the agile eye, as reported in [1], results, for any actuated joint, in a quadratic equation of the tangent of the angle of the corresponding joint rotation. With the proposed methodology, if we adopt the projection angles as angular convention, we do not need to perform calculations because the actuated joint rotation coincides with the projective angles of the mobile frame, therefore there is no need to perform calculations.

The direct analysis of the agile eye is reported in [2] and results in a polynomial of degree 8 which leads to 8 real solutions. When adopting the methodology that was proposed in this paper, it is necessary to solve a 3×3 linear system instead and optionally to add $\pm 180^\circ$ to the angles to generate the different solutions.

The problem that was addressed in the paper has multiple solutions, both for direct and inverse kinematics. The chosen solution can be selected according to common robotic practice: each solution corresponds to a different assembly configuration. Different configurations can be reached by crossing a singular configuration. Since this operation may create control problems in the execution of trajectories generally, at each time, the “most close” configuration with respect to the current configuration is chosen. So, for direct kinematics, if \mathbf{R} is the actual configuration at time T and $\mathbf{R}_1', \mathbf{R}_2', \dots$ are the different solutions for $t = T + dT$, the i -th solution chosen is the one that minimizes $\|\mathbf{R} - \mathbf{R}_i'\|$. A similar approach can be applied to the inverse kinematics minimizing the rotation of the motors to reach the next pose. This also ensures the absence of discontinuity in the joint motion.

If the angular position of the end-effector is represented by projective angles, the Jacobian matrix is the identity matrix and therefore, in the domain of the projective angles, the PKM can be defined as decoupled.

Author Contributions: Conceptualization, G.L. and I.F.; Writing—Original Draft Preparation, G.L. and I.F., Writing-Review & Editing G.L., Project Administration, I.F.

Funding: Project Evolut and UniBS ex 60%.

Acknowledgments: The 3D model of Figure 5 was designed and fabricated by Giulio Spagnuolo (ITIA-CNR). His cooperation is greatly appreciated.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Gosselin, C.; Angeles, J. The Optimum Kinematic Design of a Spherical Three-Degrees-of-Freedom Parallel Manipulator. *ASME J. Mech. Trans. Autom. Des.* **1989**, *111*, 202–207. [[CrossRef](#)]
2. Gosselin, C.M.; Sefrioui, J.; Richard, M.J. On the direct kinematics of spherical three-degree-of-freedom parallel manipulators of general architecture. *ASME J. Mech. Des.* **1994**, *116*, 594–598. [[CrossRef](#)]

3. Gosselin, C.M.; Lavoie, E. On the kinematic design of spherical three-degree-of-freedom parallel manipulators. *Int. J. Robot. Res.* **1993**, *12*, 394–402. [[CrossRef](#)]
4. Kong, X.; Gosselin, C.M. A formula that produces a unique solution to the forward displacement analysis of a quadratic spherical parallel manipulator: The Agile Eye. *ASME J. Mech. Robot.* **2010**, *2*, 044501. [[CrossRef](#)]
5. Malosio, M.; Pio Negri, S.; Pedrocchi, N.; Vicentini, F.; Caimmi, M.; Molinari Tosatti, L. A Spherical Parallel Three Degrees-of-Freedom Robot for Ankle-Foot Neuro-Rehabilitation. In Proceedings of the 34th Annual International Conference of the IEEE EMBS, San Diego, CA, USA, 28 August–1 September 2012.
6. Malosio, M.; Caimmi, M.; Ometto, M.; Molinari Tosatti, L. Ergonomics and kinematic compatibility of PKankle, a fully-parallel spherical robot for ankle-foot rehabilitation. In Proceedings of the 5th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechanics (BioRob), São Paulo, Brazil, 12–15 August 2014.
7. Gosselin, C.M.; St. Pierre, E.; Gagné, M. On the development of the Agile Eye. *IEEE Robot. Autom. Mag.* **1996**, *3*, 29–37. [[CrossRef](#)]
8. Innocenti, C.; Parenti-Castelli, V. Echelon Form Solution of Direct Kinematics for the General Fully-Parallel Spherical Wrist. *Mech. Mach. Theory* **1993**, *28*, 553–561. [[CrossRef](#)]
9. Wohlhart, K. Displacement Analysis of the General Spherical Stewart Platform. *Mech. Mach. Theory* **1994**, *29*, 581–589. [[CrossRef](#)]
10. Huang, Z.; Yao, Y.L. A New Closed-Form Kinematics of the Generalized 3-DOF Spherical Parallel Manipulator. *Robotica* **1999**, *17*, 475–485. [[CrossRef](#)]
11. Alici, G.; Shirinzadeh, B. Topology Optimisation and Singularity Analysis of a 3-SPS Parallel Manipulator with a Passive Constraining Spherical Joint. *Mech. Mach. Theory* **2004**, *39*, 215–235. [[CrossRef](#)]
12. Kuo, C.-H.; Dai, J.S.; Legnani, G. A non-overconstrained variant of the Agile Eye with a special decoupled kinematics. *Robotica* **2014**, *32*, 889–905. [[CrossRef](#)]
13. Legnani, G.; Fassi, I. Representation of 3D Motion by Projective Angles. In Proceedings of the ASME IDETC2015, Boston, MA, USA, 2–5 August 2015.
14. Legnani, G.; Casolo, F.; Righettini, P.; Zappa, B. A homogeneous matrix approach to 3D kinematics and dynamics—I. Theory. *Mech. Mach. Theory* **1996**, *31*, 573–587. [[CrossRef](#)]
15. Legnani, G.; Casolo, F.; Righettini, P.; Zappa, B. A homogeneous matrix approach to 3D kinematics and dynamics—II. Applications to chain of rigid bodies and serial manipulators. *Mech. Mach. Theory* **1996**, *31*, 589–605. [[CrossRef](#)]
16. Kong, X.; Gosselin, C.M. Type Synthesis of Three-Degree-of-Freedom Spherical Parallel Manipulators. *Int. J. Robot. Res.* **2004**, *23*, 237–245. [[CrossRef](#)]
17. Kong, X.; Gosselin, C.M. Type Synthesis of 3-DOF Spherical Parallel Manipulators Based on Screw Theory. *ASME J. Mech. Des.* **2004**, *126*, 101–126. [[CrossRef](#)]
18. Fang, Y.; Tsai, L.-W. Structure Synthesis of a Class of 3-DOF Rotational Parallel Manipulators. *IEEE Trans. Robot. Autom.* **2004**, *20*, 117–121. [[CrossRef](#)]
19. Karouia, M.; Hervé, J.M. Asymmetrical 3-Dof Spherical Parallel Mechanisms. *Eur. J. Mech. A Solids* **2005**, *24*, 47–57. [[CrossRef](#)]
20. Hess-Coelho, T.A. Topological Synthesis of a Parallel Wrist Mechanism. *ASME J. Mech. Des.* **2006**, *128*, 230–235. [[CrossRef](#)]
21. Karouia, M.; Hervé, J.M. A Three-DOF Tripod for Generating Spherical Rotation. In *Advances in Robot Kinematics*; Lenarčič, J., Stanišič, M.M., Eds.; Springer: Basel, Switzerland, 2000; pp. 395–402.
22. Di Gregorio, R. Kinematics of a New Spherical Parallel Manipulator with Three Equal Legs: The 3-URC Wrist. *J. Robot. Syst.* **2001**, *18*, 213–219. [[CrossRef](#)]
23. Di Gregorio, R. A New Parallel Wrist Using Only Revolute Pairs: The 3-RUU Wrist. *Robotica* **2001**, *19*, 305–309. [[CrossRef](#)]
24. Di Gregorio, R. A New Family of Spherical Parallel Manipulators. *Robotica* **2002**, *20*, 353–358. [[CrossRef](#)]
25. Di Gregorio, R. Kinematics of the 3-UPU Wrist. *Mech. Mach. Theory* **2003**, *38*, 253–263. [[CrossRef](#)]
26. Di Gregorio, R. The 3-RRS Wrist: A New, Simple and Non-Overconstrained Spherical Parallel Manipulator. *ASME J. Mech. Des.* **2004**, *126*, 850–855. [[CrossRef](#)]
27. Di Gregorio, R. Kinematics of the 3-RSR wrist. *IEEE Trans. Robot.* **2004**, *20*, 750–753. [[CrossRef](#)]
28. Karouia, M.; Hervé, J.M. Non-Overconstrained 3-Dof Spherical Parallel Manipulators of Type: 3-RCC, 3-CCR, 3-CRC. *Robotica* **2006**, *24*, 85–94. [[CrossRef](#)]

29. Gogu, G. Fully-Isotropic Three-Degree-of-Freedom Parallel Wrists. In Proceedings of the IEEE International Conference on Robotics and Automation, Rome, Italy, 10–14 April 2007; pp. 895–900.
30. Enferadi, J.; Tootoonchi, A.A. A Novel Spherical Parallel Manipulator: Forward Position Problem, Singularity Analysis, and Isotropy Design. *Robotica* **2009**, *27*, 663–676. [[CrossRef](#)]
31. Baumann, R.; Maeder, W.; Glauser, D.; Clavel, R. The PantoScope: A Spherical Remote-Center-of-Motion Parallel Manipulator for Force Reflection. In Proceedings of the IEEE International Conference on Robotics and Automation, Albuquerque, NM, USA, 20–25 April 1997; pp. 718–723.
32. Vischer, P.; Clavel, R. Argos: A Novel 3-DoF Parallel Wrist Mechanism. *Int. J. Robot. Res.* **2000**, *19*, 5–11. [[CrossRef](#)]
33. Kuo, C.-H.; Dai, J.S. Kinematics of a Fully-Decoupled Remote Center-of-Motion Parallel Manipulator for Minimally Invasive Surgery. *ASME J. Med. Devices* **2012**, *6*, 021008. [[CrossRef](#)]
34. Carricato, M.; Parenti-Castelli, V. A Novel Fully Decoupled Two-Degrees-of-Freedom Parallel Wrist. *Int. J. Robot. Res.* **2004**, *23*, 661–667. [[CrossRef](#)]
35. Gallardo, J.; Rodríguez, R.; Caudillo, M.; Rico, J.M. A Family of Spherical Parallel Manipulators with Two Legs. *Mech. Mach. Theory* **2008**, *43*, 201–216. [[CrossRef](#)]
36. Kuo, C.-H. Projective-Angle-Based Rotation Matrix and Its Applications. In Proceedings of the Second IFToMM Asian Conference on Mechanism and Machine Science, Tokyo, Japan, 7–10 November 2012.
37. Legnani, G.; Fassi, I.; Giberti, H.; Cinquemani, S.; Tosi, D. A new isotropic and decoupled 6-DoF parallel manipulator. *Mech. Mach. Theory* **2012**, *58*, 64–81. [[CrossRef](#)]
38. Angeles, J. *Rational Kinematics*; Springer: London, UK, 1988.
39. Nappo, F. On the absence of a general link between the angular velocity and the characterization of a rigid displacement. *Rendiconti di Matematica* **1970**, *3*. Series VI.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Non-Linear Lumped-Parameter Modeling of Planar Multi-Link Manipulators with Highly Flexible Arms

Ivan Giorgio ^{1,2,*} and Dionisio Del Vescovo ^{1,2,†}

¹ Department of Mechanical and Aerospace Engineering, SAPIENZA Università di Roma, via Eudossiana 18, 00184 Rome, Italy; dionisio.delvescovo@uniroma1.it

² Research center on Mathematics and Mechanics of Complex Systems, Università degli studi dell'Aquila, 67100 L'Aquila, Italy

* Correspondence: ivan.giorgio@uniroma1.it

† These authors contributed equally to this work.

Received: 28 June 2018; Accepted: 21 September 2018; Published: 25 September 2018

Abstract: The problem of the trajectory-tracking and vibration control of highly flexible planar multi-links robot arms is investigated. We discretize the links according to the Hencky bar-chain model, which is an application of the lumped parameters techniques. In this approach, each link is considered as a kinematic chain of rigid bodies, and suitable springs are added in order to model bending resistance. The control strategy employed is based on an optimal input pre-shaping and a feedback of the joint angles to treat the effects of undesired disturbances. Some numerical examples are given to show the potentialities of the proposed control, and a comparison with a standard collocated Proportional-Derivative (PD) control strategy is performed. In particular, we study the cases of a linear and a parabolic trajectory with a polynomial time law chosen to minimize the onset of possible vibrations.

Keywords: nonlinear flexible beams; discrete modeling; underactuated robots; optimal preshaping input

1. Introduction

Although the literature on flexible robotic manipulators is very varied and covers different aspects of the dynamic analysis and control of these mechanical systems [1–6], in industrial practice, the robotic arms are still treated as rigid multi-body systems. Only in aerospace, for evident weight issues, this strong assumption has long since been removed. In this context, however, moderately flexible links are considered to simplify the analysis and synthesis of rest-to-rest motion, trajectory-tracking and vibration control, resulting in a degradation of performance.

In this paper, we want to address the problem of considering planar multi-link highly flexible robotic arms whose transversal deflections are very large, and therefore, no linearization procedure is adoptable (refer to [7,8] for some relevant real applications). For these systems, it is possible to model the links with the continuous non-linear beam model of Euler–Bernoulli, i.e., the elastica theory [9–15]. However, as this continuous one-dimensional model is characterized by infinite degrees of freedom, it is rather difficult to analyze the manipulator's motion and to design the related control. For this reason, usually a spatial discretization of the continuous model is resorted to. Typically, three methods of discretization are used: the assumed modes technique; the Finite Element Method (FEM); and the lumped parameters method. The assumed modes approach is an intrinsically linear method, and therefore, it is applicable only to a one-link system with small flexibility or multi-link systems in which the task time is much greater than the characteristic period of oscillation of the first modes. The latter assumption is strongly requested because a progressive linearization must be carried out in the neighborhood of the generic current configuration, which indeed varies with the motion [16,17]. The finite element method can be seen as a special case of the assumed

modes technique, but it can be generalized to the nonlinear case and can therefore cover also the cases taken into account for the present study. As an example, we mention Du et al. [18], who address the problem of a non-linear 3D flexible manipulator with FE analysis. The discretization by lumped parameters [19,20] is one of the earliest methods and is characterized by an extreme simplicity of modeling and computational advantages. Moreover, being intrinsically nonlinear, it does not introduce any approximation due to some kind of linearization [21,22]. Although the finite element method converges faster than that of lumped parameters and requires a lower number of degrees of freedom to obtain the same accuracy, to deal with nonlinear cases, it is necessary to make use of a specific, rather complex formulation, which could be familiar to experts of computational mechanics, but less accessible to a larger audience (see, e.g., [23]). It should be noted, indeed, that the commercial FEM codes are rather lacking in dealing with the problem of large deflections of beams, to the best of the authors' knowledge. There are many papers that analyze the accuracy of the lumped parameters approach from different points of view, and in all of them, it is possible to find that the error in the approximation with such a method is inversely proportional at least to the first power of the number of elements used [24,25]. In the case of a cantilever beam subject to flexural oscillations, the inverse square law for the error can be verified without difficulty [26]. This means that with a rather simple modeling method, it is possible to reach the desired degree of accuracy simply by appropriately selecting the number of elements; a thumb rule is to use 13 elements per wave-length to be analyzed [26]. In the past, often the lumped-parameter approach was underestimated because assigning spring lumped constants was considered not straightforward [3]. Unfortunately, this is the result of a certain compartmentalization of expertise that sometimes occurs [27]. Indeed, it is widely known to those dealing with beam homogenization how to properly assign these discretized constants of stiffness [28–33].

In this paper, we will use the lumped parameters approach because of its simplicity and versatility. In this formulation, an elastic rod is discretized into a set of rigid segments, which are free to rotate relative to their adjacent neighbors. Springs located in these joints give the system the ability to resist bending. The case study can be classified as an under-actuated system. Indeed, the considered multi-link arm is subject to a lower number of actuators than degrees of freedom. The particular formulation adopted, among the various advantages, could make use of widely established results in classical robotics for this type of problem since in fact the model adopted is a kinematic chain of rigid bodies (see, e.g., [34–36] and the references therein).

System flexibility leads to vibration and, in turn, to an imprecise positioning due mainly to a non-minimum phase character of the system. As an illustrative numerical example, we consider a control strategy for the problem of the trajectory-tracking in the framework of the input shaping control [37–39] with a feedback for the stabilization of the response. In particular, to obtain the command torques, instead of using proper filtering as is usually done, we formulate an optimal control problem with the aim of minimizing the positioning error of the tip manipulator (see, e.g., [40]). This variational approach has been adopted for trajectory planning both for flexible [41,42] and for rigid arms [6]. Herein, instead of obtaining a trajectory with the desired mechanical characteristics, the best possible input command is found to follow a given trajectory.

The paper is organized as follows: Section 2 is devoted to describing the adopted discrete method, which is applied to a planar two-link flexible arm. Section 3 reports on numerical simulations for some trajectory-tracking cases and a comparison with a standard control approach. The paper ends with the conclusions and some future perspectives.

2. Dynamic Modeling of the Flexible Robot Manipulator

To address the complex problem of trajectory control of the tip for flexible robot manipulators, we propose to study an elemental prototype case whose behavior is rich enough to easily extend the obtained results to more generalized situations, namely more links or manipulators subject to a 3D motion (see, e.g., [43] for the analogous 3D formulation). Specifically, a planar horizontal robot

manipulator constituted of two highly flexible links is considered. Each link of the manipulator is characterized by a length ℓ_i (with $i = \{1, 2\}$), a uniform distribution of stiffness and mass density and is driven by an actuator, with mass m_{hi} , inertia J_{hi} and supplying a torque τ_i . The manipulator eventually may carry a tip payload of mass m_p and inertia J_p .

To model this system, we consider a lumped-parameter discretization. Since the manipulator is made up of two links, which can be modeled in the range of large deflections by the elastica theory, we adopt the well-known Hencky technique [28,29,43–45] to discretize the system and, then, using a more comfortable Lagrangian mechanical system, to study the motion of the robot and to design the trajectory tracing. Therefore, the considered discrete system consists of two articulated chains of n_i rigid rods of length η connected to each other by means of zero-torque hinges, also known as ‘pseudo-joints’. At each joint of the same link, a rotational spring is placed in order to model the resistance to being bent of the arm (see Figure 1). In other words, the torques provided by these springs represent the spatial discretization of the internal actions of the links, namely the bending moment. The Lagrangian coordinates, which describe the configurations of the manipulator are $\Phi_j(t)$ (with $j = \{1, 2, \dots, n_1 + n_2\}$). In particular, they represent the orientation of the rigid rods with respect to the x -axis. Moreover, the following definitions are useful to specify the angles of the actuation:

$$\begin{cases} \theta_1 = \Phi_1 \\ \theta_2 = \Phi_{n_1+1} - \Phi_{n_1} \end{cases} \quad (1)$$

for the two links, and the relative angles:

$$\begin{cases} \varphi_{h-1}^{(1)} = \Phi_h - \Phi_{h-1} & \text{with } h = 2, \dots, n_1 \\ \varphi_{k-1-n_1}^{(2)} = \Phi_k - \Phi_{k-1} & \text{with } k = n_1 + 2, \dots, n_1 + n_2 \end{cases} \quad (2)$$

which are relevant to define the deformation energy of the links whose label is reported between parentheses as a superscript. Indeed, the relative angles can be used to describe a discrete point-wise curvature. The two links prior to deformation are straight. Their mass is discretized with lumped masses m_i at the boundaries of each rigid segment by dividing the mass of each segment at its ends. The position of each point mass can be written as:

$$\begin{cases} x_j(t) = \sum_{k=1}^j \eta \cos(\Phi_k) \\ y_j(t) = \sum_{k=1}^j \eta \sin(\Phi_k) \end{cases} \quad (3)$$

and by a differentiation with respect to time, the velocities of the point masses are given by:

$$\begin{cases} \dot{x}_j(t) = -\sum_{k=1}^j \eta \dot{\Phi}_k \sin(\Phi_k) \\ \dot{y}_j(t) = \sum_{k=1}^j \eta \dot{\Phi}_k \cos(\Phi_k) \end{cases} \quad (4)$$

The equations of the motion can be derived from the Lagrangian:

$$\mathcal{L} = \mathfrak{K} - \Psi \quad (5)$$

where \mathfrak{K} and Ψ are the kinetic and potential energies of the system, respectively. Particularly, the kinetic energy can be obtained by the sum of three contributions, i.e., a term due to the links:

$$\mathfrak{K}_\ell = \sum_{j=1}^{n_1+n_2} \frac{1}{2} m_j \left\{ \left[\sum_{k=1}^j \eta \dot{\Phi}_k \sin(\Phi_k) \right]^2 + \left[\sum_{k=1}^j \eta \dot{\Phi}_k \cos(\Phi_k) \right]^2 \right\} \quad (6)$$

a term due to the two actuators:

$$\mathfrak{R}_h = \frac{1}{2} J_{h1} \dot{\Phi}_1^2 + \frac{1}{2} m_{h2} (\dot{x}_{n_1}^2 + \dot{y}_{n_1}^2) + \frac{1}{2} J_{h2} \dot{\Phi}_{n_1+1}^2 \quad (7)$$

and the last term for the payload:

$$\mathfrak{R}_p = \frac{1}{2} m_p (\dot{x}_{n_1+n_2}^2 + \dot{y}_{n_1+n_2}^2) + \frac{1}{2} J_p \dot{\Phi}_{n_1+n_2}^2 \quad (8)$$

The elastic potential energy Ψ is assumed to be:

$$\Psi = \sum_{i=1}^2 \sum_{j=1}^{n_i-1} \kappa_{bj}^{(i)} [1 - \cos(\varphi_j^{(i)})] \quad (9)$$

where a lumped bending stiffness $\kappa_{bj}^{(i)} = YJ_{\ell_i}/\eta$ associated with the rotational springs is introduced using the elastic modulus of the beam's material, Y , and the second moment of the area of the beam's cross-section, J_{ℓ_i} [46,47]. Note that the potential energy in Equation (9) is positive definite, and in the continuum limit, i.e., for η tending to zero, the expression (9) becomes an energy density, which is quadratic in the curvature of the beam axis [46], in accord with the elastica theory. As a first approximation, we also consider a viscous dissipation (see, e.g., [48,49]), introducing the Rayleigh dissipation function:

$$\mathfrak{R} = \sum_{i=1}^2 \sum_{j=1}^{n_i-1} \frac{1}{2} c_{bj}^{(i)} (\dot{\varphi}_j^{(i)})^2 \quad (10)$$

$c_{bj}^{(i)}$ being a lumped viscous coefficient. This type of dissipation can be associated with the rate of the bending deformation; thus with similar reasoning used for the elastic bending mode deformation, it is possible to evaluate the lumped dissipation with the expression: $c_{bj}^{(i)} = C_v J_{\ell_i}/\eta$, where the material parameter C_v is the viscous coefficient of the beam material, which can be experimentally identified (see, e.g., [50]).

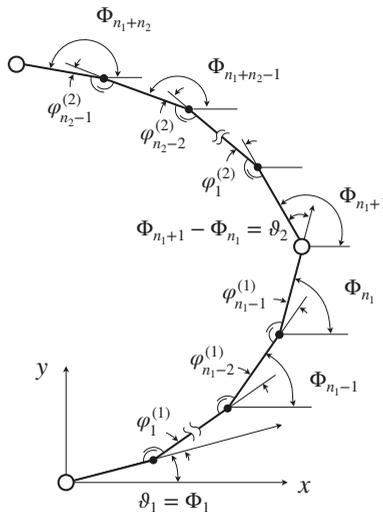


Figure 1. Discrete system for the planar two-link flexible robot.

The Euler–Lagrange equations of motion, thus, are:

$$\frac{\partial}{\partial t} \left(\frac{\partial \bar{\mathcal{K}}}{\partial \dot{\Phi}_i} \right) - \frac{\partial \bar{\mathcal{K}}}{\partial \Phi_i} + \frac{\partial \Psi}{\partial \Phi_i} + \frac{\partial \mathfrak{A}}{\partial \dot{\Phi}_i} = Q_i \quad \text{for } i = 1 \dots n_1 + n_2 \quad (11)$$

The virtual work of the torques applied by actuators at the basis of each link is:

$$\delta W = \tau_1 \delta \theta_1 + \tau_2 \delta \theta_2 = \tau_1 \delta \Phi_1 + \tau_2 (\delta \Phi_{n_1+1} - \delta \Phi_{n_1}) \quad (12)$$

hence, the only generalized forces different from zero are:

$$Q_1 = \tau_1, \quad Q_{n_1+1} = \tau_2, \quad Q_{n_1} = -\tau_2 \quad (13)$$

One of the performance specifications in a control system should be a satisfactory regulation against disturbances. Here, to test the effectiveness of the proposed control, we take into consideration two kinds of disturbances, due to the actual realization of the actuators, that add up to the torques introduced in Equation (13); specifically, friction torques, which arise at the actuated joints, and cogging torques, typical for electrical motors. The former disturbance is described by a Lund–Grenoble model [51], which is able to capture most of the major nonlinear effects involved in the considered case such as pre-sliding displacement, stick-slip motion, the Stribeck effect, and so forth; the latter disturbance, due to the interaction between the permanent magnets of the rotor and the slots of the stator, is modeled by means of a constitutive relationship, experimentally identified, i.e., a periodic function of the relative position between the stator and the rotor of the motor [52]. An evolution rule for the friction torques τ_{fi} is assumed as follows:

$$\frac{d\tau_{fi}}{dt} = k_i \dot{\theta}_i \left(1 - \frac{\tau_{fi}}{\tau_L(\dot{\theta}_i)} \text{sign}(\dot{\theta}_i) \right) \quad (14)$$

where $\tau_L(\dot{\theta}_i) = \tau_C + (\tau_S - \tau_C) \exp \left[- (\dot{\theta}_i / v_s)^2 \right]$ is the limit torque introduced to take into account the Stribeck effect. In detail, $\tau_S = 0.2 \text{ Nm}$ is the static friction; $\tau_C = 0.1 \text{ Nm}$ is the Coulomb friction torque; and $v_s = 0.1 \text{ s}^{-1}$ represents the Stribeck velocity. The cogging torques are evaluated as a function of the angles θ_i as follows:

$$\tau_{cgi} = T_{cg} \sum_{k=1}^6 B_k \sin(n_p k \theta_i) \quad (15)$$

where $T_{cg} = 0.1 \text{ Nm}$ is the amplitude of the torque, $n_p = 2$ is the number of poles of the motors and the coefficients of the trigonometric polynomial B_k are assumed to be $\{-0.7937, -0.3586, -0.0341, 0.0039, -0.0016, -0.0064\}$ in the performed simulations.

3. Numerical Examples for Some Trajectory-Tracking Cases

In this section, a control scheme for trajectory-tracking and vibration control of flexible arms is used to show the potentialities of the proposed formulation. This approach is based on an optimal design of the command torques applied to the actuated joints that aim at following the desired trajectory and reducing vibrations. The control strategy, thus, includes a feedforward control based on such a command input and a feedback control that stabilizes the 2R flexible robot along the desired trajectory by a joint-based collocated Proportional-Derivative (PD) controller [39]. Precisely, the optimal control technique is used to produce input profiles for the torques acting on the flexible system as described below. Since the command shaping technique does not require additional sensors or actuators, this technique is particular attractive in order to have a hardware apparatus for the control characterized by minimal equipment.

As a first step, we plan a desired trajectory $x_d(t)$ connecting the ends of each link from arbitrary initial points to desired final points in a given time interval $\mathcal{I} = [0, t_f]$, i.e., we set

$x_d(t) = (x_{n_1,d}, y_{n_1,d}, x_{n_1+n_2,d}, y_{n_1+n_2,d})$. Then, denoting the actual trajectory of the ends of each link, evaluated on the solution of Equation (11) in \mathcal{I} without disturbances, with $\tilde{x}(t) = (\tilde{x}_{n_1}, \tilde{y}_{n_1}, \tilde{x}_{n_1+n_2}, \tilde{y}_{n_1+n_2})$, the optimal control problem for the design of the input torques can be formulated as follows:

Find the torques $\tau(t) = (\tau_1, \tau_2)$ as real-valued smooth functions defined on \mathcal{I} , which minimize the continuous-time cost functional:

$$J(\tilde{\Phi}, \tau, t) = \frac{1}{2} \int_0^{t_f} (\tilde{x} - x_d)^\top R (\tilde{x} - x_d) dt \tag{16}$$

subject to the dynamic constraints that $\tilde{x}(t)$ is computed on the solution of Equation (11) with given initial conditions. R is a constant diagonal positive definite weight matrix.

In detail, we directly minimize the functional J varying the torques $\tau_1(t)$ and $\tau_2(t)$ instead of solving the Euler–Lagrange equations, which may laboriously be obtainable by means of calculus of variations. Therefore, representing the torques τ_i in a discrete way as follows:

$$\tilde{\tau}_i(t) = \tau_{0i}(t) + w(t) \sum_{h=0}^{n_d} a_h^{(i)} p_h(t) \tag{17}$$

where $\tau_{0i}(t)$ is a reference torque, $p_h(t)$ are interpolation functions defined on \mathcal{I} and $w(t)$ is a proper window function, the problem (16) results in finding the coefficients $a_h^{(i)}$ that minimize the functional J evaluated with the approximated shapes $\tilde{\tau}_i(t)$. The particular form chosen for Equation (17) is based on the idea of finding an approximate solution for the considered problem, by starting from the exact solution of a related, simpler problem and then adding a correction. The first term $\tau_{0i}(t)$, indeed, is the required torque evaluated for the 2R rigid robot, while the other term represents the correction needed to solve the primary problem. In particular, the function $w(t)$ is conceived of in order to account for a correction of the torques, which tends to zero at the beginning and at the end of the interval \mathcal{I} . In this way, a jump in the torques, responsible for the onset of possible vibrations, can be avoided. A possible choice of this function $w(t)$ is the Welch window, defined as:

$$w(t) = \begin{cases} 4 \frac{t}{t_f} \left(1 - \frac{t}{t_f}\right) & \text{for } t \in \mathcal{I} \\ 0 & \text{outside } \mathcal{I} \end{cases} \tag{18}$$

Indeed, this window is designed, as many others, to moderate the sudden changes of a rectangular window and, thus, to improve dynamic range. Regarding the p_h functions, thinking of a Taylor expansion properly truncated to express the torque corrections, we consider $p_h(t) = t^h$. Of course, the corrections can be expressed in alternative ways, for example a truncated Fourier series can also be a valid representation. In both cases, a convergence analysis is needed to determine how many terms should be taken into account to minimize the truncation error.

Once the optimal torques $\tau_{id}(t)$ are obtained by solving the problem (16), the complete control strategy can be expressed in the following way:

$$\tau_i = \tau_{di} + K_{Pi}(\vartheta_{di} - \vartheta_i) + K_{Di}(\dot{\vartheta}_{di} - \dot{\vartheta}_i) \tag{19}$$

where $(\vartheta_i, \dot{\vartheta}_i)$ are related to the actual trajectory for the joint angles, directly measured by the motor encoders, while $(\vartheta_{di}, \dot{\vartheta}_{di})$ correspond to the desired trajectory, which is computed off-line by a numerical simulation of the manipulator performed using the optimal torques $\tau_{di}(t)$ as input. To choose the PD gains, a standard technique can be employed based on setting the natural frequencies, which govern the speed of response, as well as taking into account the saturation of each actuator (for a detailed description see, e.g., [53]).

To illustrate the potentialities of the proposed control strategy, we examine some representative examples in which the manipulator tip is constrained to move along two different paths, namely a straight line segment and a piece of parabolic curve.

In the first case, the desired coordinates for the end effector are assumed to be:

$$\begin{cases} x_{n_1+n_2,d}(t) = \ell_1 + \ell_2 - s(t) \cos\left(\arctan\left(\frac{\ell_1}{\ell_1+\ell_2}\right)\right) \\ y_{n_1+n_2,d}(t) = s(t) \sin\left(\arctan\left(\frac{\ell_1}{\ell_1+\ell_2}\right)\right) \end{cases} \quad (20)$$

in which the time function $s(t)$ is a Peisekah polydyne, which is expressed as follows:

$$s(t) = A_d \left[126 \left(\frac{t}{t_f}\right)^5 - 420 \left(\frac{t}{t_f}\right)^6 + 540 \left(\frac{t}{t_f}\right)^7 - 315 \left(\frac{t}{t_f}\right)^8 + 70 \left(\frac{t}{t_f}\right)^9 \right] \quad (21)$$

where $A_d = \sqrt{\ell_1^2 + (\ell_1 + \ell_2)^2}$ and the task time is set to $t_f = 1$ s. The well-known time law (21) is chosen because the values of its derivatives with respect to time up to fourth order at the initial and final times are all zero. This feature is particularly desired to avoid exciting vibrations. Here, the initial configuration provides the robotic arm arranged along the x axis completely unfolded, while the final arrangement of the arm is characterized by having the end effector in the position of coordinates $(0, \ell_1)$.

In the second case, the desired coordinates for the tip manipulator follow the parabola:

$$\begin{cases} x_{n_1+n_2,d}(t) = (\ell_1 + \ell_2) - s(t) (\ell_1 + \ell_2) \\ y_{n_1+n_2,d}(t) = \ell_1 s(t)^2 \end{cases} \quad (22)$$

where the function $s(t)$ is given by Equation (21) with $A_d = 1$ and $t_f = 1.25$ s. The initial and final configurations are set up as in the previous case.

In all the cases, we extend the desired time interval for a while in order to stabilize, in the optimization stage, the solution at the final configuration. The desired coordinates of the intermediate joint, $(x_{n_1,d}, y_{n_1,d})$, have been calculated considering the robotic arm as a 2R rigid robot for both cases addressed.

Equations (11) are numerically solved by means of the computing system Simulink considering a 2R flexible arm with links of length $\ell_1 = \ell_2 = 0.5$ m and having a rectangular cross-section of size 50×2 mm with a second moment of area $J_{\ell_1} = J_{\ell_2} = 3.33 \times 10^{-11}$ m⁴. The links are discretized using 99 rigid segments. The Young modulus of the links is $Y = 200$ GPa, and the mass of each of them is 0.3925 kg. The viscous coefficients are assumed to be $c_b^{(1)} = c_b^{(2)} = 0.15$ Nms. The actuators are characterized by masses $m_{h1} = m_{h2} = 1$ kg and moments of inertia $J_{h1} = J_{h2} = 0.1$ kg m², while the payload has $m_p = 0.1$ kg and $J_p = 0.005$ kg m². For the implementation of the optimal problem, we assume the non-vanishing elements of \mathbf{R} to be $R_{11} = R_{22} = 10$ and $R_{33} = R_{44} = 100$ to give more importance to the tip error.

Figures 2a and 3a show the torques τ_{0i} evaluated for the 2R rigid robot, while Figures 2b and 3b display the second term of Equation (17) obtained as a result of the optimization problem.

The used coefficients of the feedback loops are $K_{p1} = 25$ Nm, $K_{p2} = 22$ Nm, $K_{D1} = 0.42$ Nms and $K_{D2} = 0.22$ Nms.

In Figures 4–7 are compiled the results obtained for the two examined cases, respectively for the linear and the parabolic case.

In particular, Figures 4 and 6 show the reference path (dashed black line) for the tip of the arm in which the start and end points are highlighted with a circle and a star, respectively. These figures also exhibit the actual trajectories of the two ends of the links, as well as some intermediate deformed configurations for the links. The deformations of the links are clearly in the range of large deflections, and therefore, any linearization procedure is not allowed in the investigated cases.

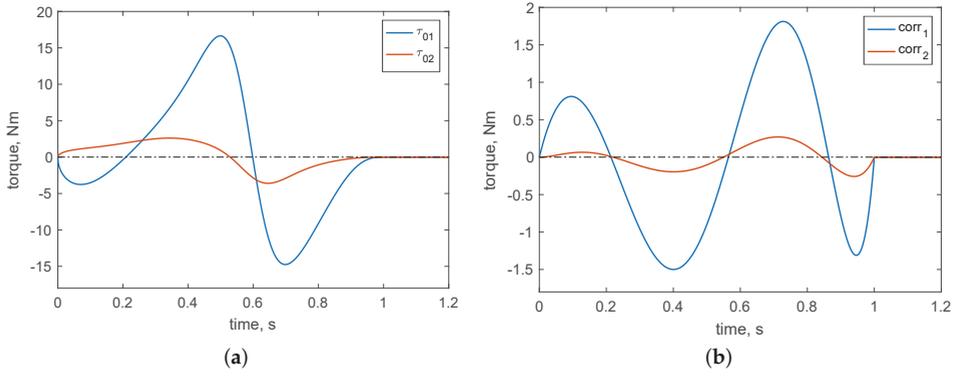


Figure 2. Linear trajectory case: (a) reference rigid torques; (b) correction torques.

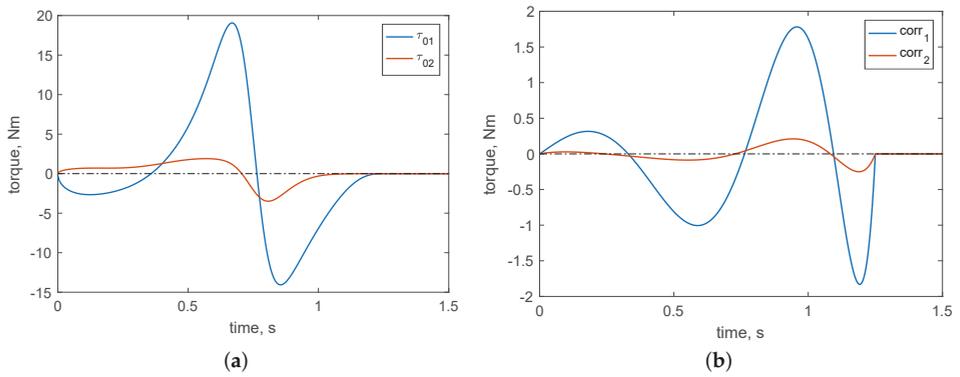


Figure 3. Parabolic trajectory case: (a) reference rigid torques; (b) correction torques.

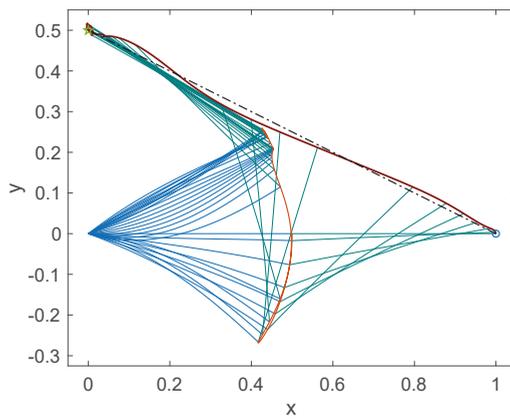


Figure 4. Stroboscopic motion for linear trajectory case.

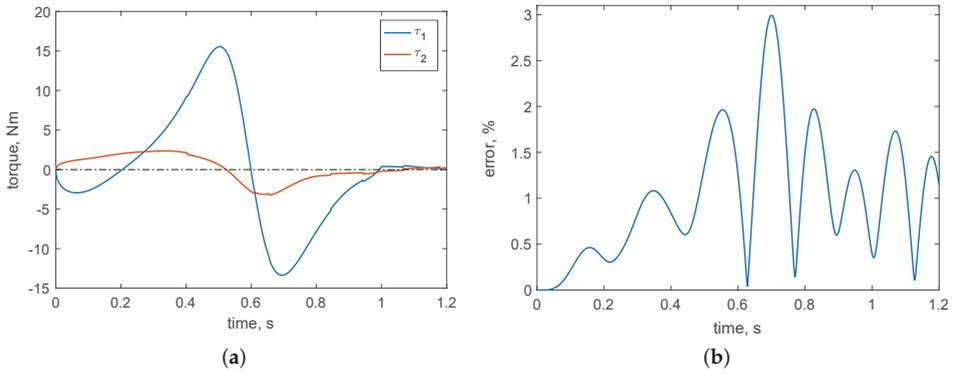


Figure 5. Linear trajectory case: (a) actual torques; (b) relative tracking error for the tip.

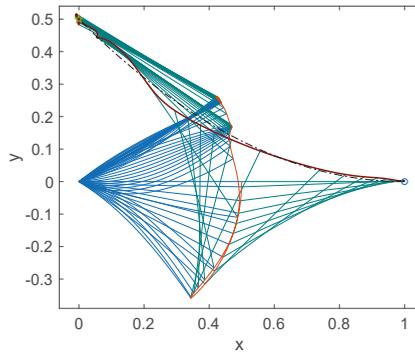


Figure 6. Stroboscopic motion for the parabolic trajectory case.

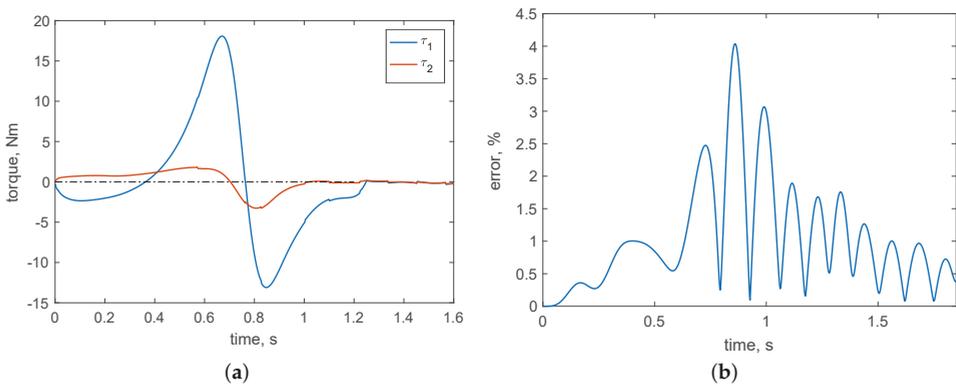


Figure 7. Parabolic trajectory case: (a) actual torques; (b) relative tracking error for the tip.

Figures 5a and 7a display the actual joint torques resulting from the control strategy of a feedforward with an optimal input command and a feedback of the signals, angles and

angular velocities, acquired from the joints. These plots evince the difference between the two joint torques, as well as the power required by the motors and therefore their size. Of course, once the time law (21) has been set, to limit the maximum torque that can be provided, it is possible to change the task time. Thus, we set the task times for the considered cases in order to limit the torques at reliable values.

Figures 5b and 7b exhibit the tip error in following the desired trajectory for the three studied cases. We can see from the graphs that the errors normalized to the full length of the robotic arm were less than 4.5% despite the great deformability of the links and the shortness of the task time. Tables 1 and 2 summarize the coefficients, obtained minimizing the functional J , to represent the torques in terms of the interpolation polynomial functions considered for the analyzed cases.

Table 1. Optimal torque coefficients for the linear trajectory case.

	$a_0^{(i)}$	$a_1^{(i)}$	$a_2^{(i)}$	$a_3^{(i)}$	$a_4^{(i)}$	$a_5^{(i)}$	$a_6^{(i)}$
Link 1	113.65	-590.40	102.40	210.97	3612.9	-3800.3	-0.00178
Link 2	-0.1364	105.65	-780.47	1570.4	-848.80	-108.65	-0.01214

Table 2. Optimal torque coefficients for the parabolic trajectory case.

	$a_0^{(i)}$	$a_1^{(i)}$	$a_2^{(i)}$	$a_3^{(i)}$	$a_4^{(i)}$	$a_5^{(i)}$	$a_6^{(i)}$	$a_7^{(i)}$	$a_8^{(i)}$
Link 1	0.8090	1.114	-11.239	-1.681	1.297	19.485	14.592	-6.116	-15.732
Link 2	0.2139	-1.434	2.659	0.0668	-9.700	10.966	3.453	-5.661	-0.2856

The order of the interpolating polynomial was fixed by increasing it subsequently until the error obtained by the optimization process stabilized. To perform the minimization, we employ a MATLAB code that makes use of the function *fminsearch*.

To estimate the efficiency of the proposed approach, we compare the used control strategy with a standard PD control with a feedback of the angular joints ($\vartheta_i, \dot{\vartheta}_i$) using a desired trajectory evaluated for the linear and parabolic rigid cases, respectively, Equations (20) and (22) and the time law (21). Figure 8 shows the relative tracking error of the manipulator tip in both examined cases with the PD feedback gains: $K_{P1} = 25$ Nm, $K_{P2} = 22$ Nm, $K_{D1} = 4.2$ Nms and $K_{D2} = 2.2$ Nms. The tracking performances obtained exhibit a maximum relative error of about 30%, much greater than the optimal pre-shaping input approach.

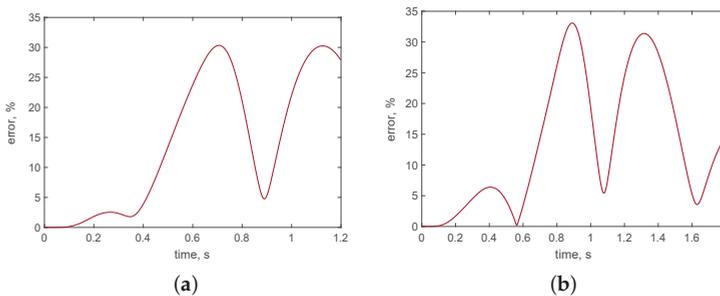


Figure 8. Relative tracking error for the tip with collocated PD control: linear trajectory case (a); parabolic trajectory case (b).

4. Conclusions

In this article, we propose a lumped parameters modeling approach for flexible robotic manipulators in the nonlinear regime. In this context, the method of the assumption of the modes,

typically used, cannot be utilized because the hypotheses on which it is based are inadequate for the given problem. We propose to use this formulation for its great simplicity of modeling and for its intrinsic nonlinear nature, instead of the finite element method, which requires a greater modeling effort even if it has a faster convergence speed. Furthermore, an isogeometric formulation, still not fully affirmed as an alternative to the standard FEM, can also improve the results because at equal accuracy, it requires a lesser number of degrees of freedom and seems to be promising for further advances (see, e.g., [54–56] for recent developments).

The number of degrees of freedom that should be treated in the nonlinear case of large link deflections does not allow us to opt for a well-established control strategy such as online computed-torque; therefore, a different control approach must be developed. We present a control approach based on a pre-shaping input that, instead of using appropriately designed signal filters, produces a feedforward command signal for the motors using an optimal problem in which a functional, properly defined, is minimized on the basis of the positioning error of the end effector of the manipulator. In order to stabilize the response close to the desired one, a feedback signal is used together to make the system less sensitive to external disturbances. The employed control strategy is therefore more suitable to treat a greater number of degrees of freedom and can be implemented with minimal hardware equipment. The system considered is under-actuated, and therefore, it is not possible to obtain a perfect positioning of the manipulator tip. However, the simulated numerical cases show that under very strict operating conditions, it is possible to obtain a trajectory tracking with an error of less than about 4.5%. A comparison with a standard feedback PD control strategy shows that, with the optimal pre-shaping input, it is possible to achieve a better tip positioning.

The optimal problem in this paper has been solved numerically to explore the possibilities offered by the proposed method. The preliminary results achieved are quite encouraging, and therefore, as a future research direction, it would be interesting to address the optimal problem rigorously with the calculus of the variations and, in this way, characterize the minimum error obtainable accurately in the various operating conditions.

Author Contributions: I.G. and D.D.V. conceived and designed the experiments; I.G. and D.D.V. performed the experiments; I.G. and D.D.V. analyzed the data; I.G. and D.D.V. contributed analysis tools; I.G. and D.D.V. wrote the paper.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cannon, R.H., Jr.; Schmitz, E. Initial experiments on the end-point control of a flexible one-link robot. *Int. J. Robot. Res.* **1984**, *3*, 62–75.
2. Tosunoglu, S.; Lin, S.H.; Tesar, D. Accessibility and controllability of flexible robotic manipulators. *J. Dyn. Syst. Meas. Control* **1992**, *114*, 50–58.
3. Sayahkarajy, M.; Mohamed, Z.; Mohd Faudzi, A.A. Review of modeling and control of flexible-link manipulators. *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.* **2016**, *230*, 861–873.
4. Önsay, T.; Akay, A. Vibration reduction of a flexible arm by time-optimal open-loop control. *J. Sound Vib.* **1991**, *147*, 283–300.
5. Meckl, P.; Seering, W. Experimental evaluation of shaped inputs to reduce vibration for a cartesian robot. *J. Dyn. Syst. Meas. Control* **1990**, *112*, 159–165.
6. Gregory, J.; Olivares, A.; Staffetti, E. Energy-optimal trajectory planning for robot manipulators with holonomic constraints. *Syst. Control Lett.* **2012**, *61*, 279–291.
7. Book, W.J. Controlled motion in an elastic world. *J. Dyn. Syst. Meas. Control* **1993**, *115*, 252–261.
8. Katzschmann, R.K.; Marchese, A.D.; Rus, D. Autonomous object manipulation using a soft planar grasping manipulator. *Soft Robot.* **2015**, *2*, 155–164.
9. Antman, S.S. Kirchhoff's problem for nonlinearly elastic rods. *Q. Appl. Math.* **1974**, *32*, 221–240.
10. Steigmann, D.J.; Faulkner, M.G. Variational theory for spatial rods. *J. Elast.* **1993**, *33*, 1–26.

11. Greco, L.; Cuomo, M. Consistent tangent operator for an exact Kirchhoff rod model. *Contin. Mech. Thermodyn.* **2015**, *27*, 861–877.
12. Altenbach, H.; Birsan, M.; Eremeyev, V.A. Cosserat-type rods. In *Generalized Continua from the Theory to Engineering Applications*; Springer: Vienna, Austria, 2013; pp. 179–248.
13. Eugster, S.R.; Hesch, C.; Betsch, P.; Glocker, C. Director-based beam finite elements relying on the geometrically exact beam theory formulated in skew coordinates. *Int. J. Numer. Methods Eng.* **2014**, *97*, 111–129.
14. Della Corte, A.; dell’Isola, F.; Esposito, R.; Pulvirenti, M. Equilibria of a clamped Euler beam (Elastica) with distributed load: Large deformations. *Math. Models Methods Appl. Sci.* **2017**, *27*, 1391–1421.
15. Spagnuolo, M.; Andreaus, U. A targeted review on large deformations of planar elastic beams: Extensibility, distributed loads, buckling and post-buckling. *Math. Mech. Solids* **2018**. [[CrossRef](#)]
16. Giorgio, I.; Della Corte, A.; Del Vescovo, D. Modelling flexible multi-link robots for vibration control: Numerical simulations and real-time experiments. *Math. Mech. Solids* **2017**. [[CrossRef](#)]
17. Low, K.H. Solution schemes for the system equations of flexible robots. *J. Field Robot.* **1989**, *6*, 383–405.
18. Du, H.; Lim, M.; Liew, K. A nonlinear finite element model for dynamics of flexible manipulators. *Mech. Mach. Theory* **1996**, *31*, 1109–1119.
19. Wang, Y.; Huston, R.L. A lumped parameter method in the nonlinear analysis of flexible multibody systems. *Comput. Struct.* **1994**, *50*, 421–432.
20. Šalinić, S. An improved variant of Hencky bar-chain model for buckling and bending vibration of beams with end masses and springs. *Mech. Syst. Signal Process.* **2017**, *90*, 30–43.
21. Yoshikawa, T.; Hosoda, K. Modeling of flexible manipulators using virtual rigid links and passive joints. *Int. J. Robot. Res.* **1996**, *15*, 290–299.
22. Konno, A.; Uchiyama, M. Vibration suppression control of spatial flexible manipulators. *Control Eng. Pract.* **1995**, *3*, 1315–1321.
23. Garcea, G.; Trunfio, G.A.; Casciaro, R. Mixed formulation and locking in path-following nonlinear analysis. *Comput. Methods Appl. Mech. Eng.* **1998**, *165*, 247–272.
24. Livesley, R. The equivalence of continuous and discrete mass distributions in certain vibration problems. *Q. J. Mech. Appl. Math.* **1955**, *8*, 353–360.
25. Leckie, F.A.; Lindberg, G.M. The effect of lumped parameters on beam frequencies. *Aeronaut. Q.* **1963**, *14*, 224–240.
26. Duncan, W.J. A critical examination of the representation of massive and elastic bodies by systems of rigid masses elastically connected. *Q. J. Mech. Appl. Math.* **1952**, *5*, 97–108.
27. Dell’Isola, F.; Bucci, S.; Battista, A. Against the fragmentation of knowledge: The power of multidisciplinary research for the design of metamaterials. In *Advanced Methods of Continuum Mechanics for Materials and Structures*; Springer: Singapore, 2016; pp. 523–545.
28. Jawed, M.K.; Novelia, A.; O’Reilly, O.M. *A Primer on the Kinematics of Discrete Elastic Rods*; Springer: Cham, Switzerland, 2018.
29. Zhang, H.; Wang, C.M.; Challamel, N. Buckling and vibration of Hencky bar-chain with internal elastic springs. *Int. J. Mech. Sci.* **2016**, *119*, 383–395.
30. Alibert, J.J.; Della Corte, A.; Seppecher, P. Convergence of Hencky-type discrete beam model to Euler inextensible elastica in large deformation: Rigorous proof. In *Mathematical Modelling in Solid Mechanics*; Springer: Singapore, 2017; pp. 1–12.
31. Alibert, J.J.; Della Corte, A.; Giorgio, I.; Battista, A. Extensional Elastica in large deformation as Γ -limit of a discrete 1D mechanical system. *Z. Angew. Math. Phys.* **2017**, *68*. [[CrossRef](#)]
32. Battista, A.; Della Corte, A.; dell’Isola, F.; Seppecher, P. Large deformations of 1D microstructured systems modeled as generalized Timoshenko beams. *Z. Angew. Math. Phys.* **2018**, *69*. [[CrossRef](#)]
33. Khakalo, S.; Balobanov, V.; Niiranen, J. Modelling size-dependent bending, buckling and vibrations of 2D triangular lattices by strain gradient elasticity models: Applications to sandwich beams and auxetics. *Int. J. Eng. Sci.* **2018**, *127*, 33–52.
34. De Luca, A.; Mattone, R.; Oriolo, G. Control of underactuated mechanical systems: Application to the planar 2R robot. In Proceedings of the 35th IEEE Conference on Decision and Control, Kobe, Japan, 13 December 1996; IEEE: Piscataway, NJ, USA, 1996; Volume 2, pp. 1455–1460.
35. De Luca, A.; Oriolo, G. Trajectory planning and control for planar robots with passive last joint. *Int. J. Robot. Res.* **2002**, *21*, 575–590.

36. De Luca, A.; Mattone, R.; Oriolo, G. Stabilization of an underactuated planar 2R manipulator. *Int. J. Robust Nonlinear Control IFAC-Affil. J.* **2000**, *10*, 181–198.
37. Mohamed, Z.; Martins, J.; Tokhi, M.; Da Costa, J.S.; Botto, M. Vibration control of a very flexible manipulator system. *Control Eng. Pract.* **2005**, *13*, 267–277.
38. Singhose, W.; Singer, N.; Seering, W. Comparison of command shaping methods for reducing residual vibration. In Proceedings of the European Control Conference, Rome, Italy, 5–8 September 1995; pp. 1126–1131.
39. De Luca, A.; Caiano, V.; Del Vecovo, D. Experiments on Rest-to-rest Motion of a Flexible Arm. In *Experimental Robotics VIII*; Springer: Berlin, Germany, 2003; pp. 338–349.
40. Pepe, G.; Carcaterra, A.; Giorgio, I.; Del Vecovo, D. Variational Feedback Control for a nonlinear beam under an earthquake excitation. *Math. Mech. Solids* **2016**, *21*, 1234–1246.
41. Boscaroli, P.; Gasparetto, A. Model-based trajectory planning for flexible-link mechanisms with bounded jerk. *Robot. Comput.-Integr. Manuf.* **2013**, *29*, 90–99.
42. Boscaroli, P.; Gasparetto, A. Optimal trajectory planning for nonlinear systems: Robust and constrained solution. *Robotica* **2016**, *34*, 1243–1259.
43. Turco, E. Discrete is it enough? The revival of Piola–Hencky keynotes to analyze three-dimensional Elasticity. *Contin. Mech. Thermodyn.* **2018**. [[CrossRef](#)]
44. Rubinstein, D. Dynamics of a flexible beam and a system of rigid rods, with fully inverse (one-sided) boundary conditions. *Comput. Methods Appl. Mech. Eng.* **1999**, *175*, 87–97.
45. Wang, C.M.; Zhang, H.; Gao, R.P.; Duan, W.H.; Challamel, N. Hencky bar-chain model for buckling and vibration of beams with elastic end restraints. *Int. J. Struct. Stab. Dyn.* **2015**, *15*. [[CrossRef](#)]
46. Dell’Isola, F.; Giorgio, I.; Pawlikowski, M.; Rizzi, N.L. Large deformations of planar extensible beams and pantographic lattices: Heuristic homogenization, experimental and numerical examples of equilibrium. *Proc. R. Soc. A* **2016**, *472*. [[CrossRef](#)]
47. Turco, E.; dell’Isola, F.; Cazzani, A.; Rizzi, N.L. Hencky-type discrete model for pantographic structures: Numerical comparison with second gradient continuum models. *Z. Angew. Math. Phys.* **2016**, *67*. [[CrossRef](#)]
48. Altenbach, H.; Eremeyev, V.A. On the constitutive equations of viscoelastic micropolar plates and shells of differential type. *Math. Mech. Complex Syst.* **2015**, *3*, 273–283.
49. Cuomo, M. Forms of the dissipation function for a class of viscoplastic models. *Math. Mech. Complex Syst.* **2017**, *5*, 217–237.
50. Dietrich, L.; Lekszycki, T.; Turski, K. Problems of identification of mechanical characteristics of viscoelastic composites. *Acta Mech.* **1998**, *126*, 153–167.
51. De Wit, C.C.; Olsson, H.; Astrom, K.J.; Lischinsky, P. A new model for control of systems with friction. *IEEE Trans. Autom. Control* **1995**, *40*, 419–425.
52. Buechner, S.; Schreiber, V.; Amthor, A.; Ament, C.; Eichhorn, M. Nonlinear modeling and identification of a dc-motor with friction and cogging. In Proceedings of the Industrial Electronics Society, IECON 2013–39th Annual Conference of the IEEE, Vienna, Austria, 10–13 November 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 3621–3627.
53. Dawson, D.M.; Abdallah, C.T.; Lewis, F.L. *Robot Manipulator Control: Theory and Practice*; CRC Press: Boca Raton, FL, USA, 2003.
54. Greco, L.; Cuomo, M.; Contraffatto, L.; Gazzo, S. An efficient blended mixed B-spline formulation for removing membrane locking in plane curved Kirchhoff rods. *Comput. Methods Appl. Mech. Eng.* **2017**, *324*, 476–511.
55. Cazzani, A.; Malagù, M.; Turco, E. Isogeometric analysis of plane-curved beams. *Math. Mech. Solids* **2016**, *21*, 562–577.
56. Niiranen, J.; Balobanov, V.; Kiendl, J.; Hosseini, S. Variational formulations, model comparisons and numerical methods for Euler–Bernoulli micro-and nano-beam models. *Math. Mech. Solids* **2017**. [[CrossRef](#)]



Article

Adaptive Balancing of Robots and Mechatronic Systems

Liviu Ciupitu

Mechanical Engineering and Mechatronics Department, Politehnica University of Bucharest, 060042 Bucharest, Romania; liviu.ciupitu@gmail.com; Tel.: +40-072-274-5319

Received: 16 September 2018; Accepted: 2 November 2018; Published: 7 November 2018

Abstract: Present paper is dealing with the adaptive static balancing of robot or other mechatronic arms that are rocking in vertical plane and whose static loads are variable, by using counterweights and springs. Some simple passive and approximate solutions are proposed, and an example is shown. The results show that a very simple passive solution which is using for gravity compensation a simple translational counterweight (that could be for example the actuating motor itself) articulated by one single bar leads to very good results in case of approximate balancing when the payload has a known variation.

Keywords: adaptive; balancing; counterweight; mechatronic system; robot

1. Introduction

Static balancing of a mechanical system is one of the first demanding steps in the design process of any mechanical system which is moving with relatively small accelerations and which is overcoming relatively large forces, in order to match first of all the need of energy consumption and it is also an important aspect of the overall performance of it [1].

Static balancing can be regarded as the total or partial cancellation of the mechanical effects (force or moment) of static loads to the actuating system of mechanical system, in all configurations, respectively in a finite number of configurations, from functioning domain, under quasi-static conditions [1,2]. The effect of this action is the maintaining of the mechanical system in a rest state at any configuration or at a finite number of configurations respectively, from working space and its actuators are not required to overcome the static loads. The movement inside working space can be done with a power-less actuating system which consumes energy only for overcoming the friction forces and balancing errors. Anyway the friction forces are opposing to the movement, contributing in this way to the maintaining of the mechanical system in a rest state.

The main static load is given by gravitational field of Earth, which is reflecting into the weight forces of all bodies that compose the mechanical system. In the case that weight forces are the only static loads of static balancing operation then the mechanical system is called gravity compensate. Also the effect of these loads to the actuating system is present only in the case that the mechanical system is not working in horizontal plane with respect to gravity field.

Consequently, the potential energy of mechanical system remains constant or approximately constant and the centre of gravity of mechanical system remain fixed with respect to a referential frame or is moving along a horizontal direction or into a horizontal plane with respect to Earth.

Another important observation and hypothesis is that due to the small displacements of the centres of gravity of elements, with respect to the distance from the centre of the Earth to each body mass centres, then the weight forces are constant.

In this case the actuators of mechatronic system are not required to sustain the weight of its moving elements.

But, in the case of a manipulation robot for example, as is also the case of cranes too, the manipulation weight could be variable in steps. As is presented in article [3] for the case of an industrial robot [4] which is designed to manipulate payloads of 16 kg maximum mass, balanced by springs for a middle weight mass of 8 kg, the forces induced in actuating system are amplified about 4 times when the weight is increasing or decreasing from the mean value. In fact, in terms of resistance moments (torques) at shafts of rotating actuators, as is shown in Figure 1a for the most frequent case of an articulated arm, this variation occurs (and has a cosine variation) even the payload has constant weight G_p . In case the load has variable weight (as is the case of oil pump-jack systems for example [5] then a more complex variation is possible (Figure 1b—solid curve line 1). A special situation is the one when the variation is known, and it is repeating during one cycle. In this case the adaptive solution could be a passive one (i.e., not actuated). Otherwise the balancing system should adapt in real time by using a local and supplementary actuation system and by aid of a controlling system and the required sensors and transducers [2].

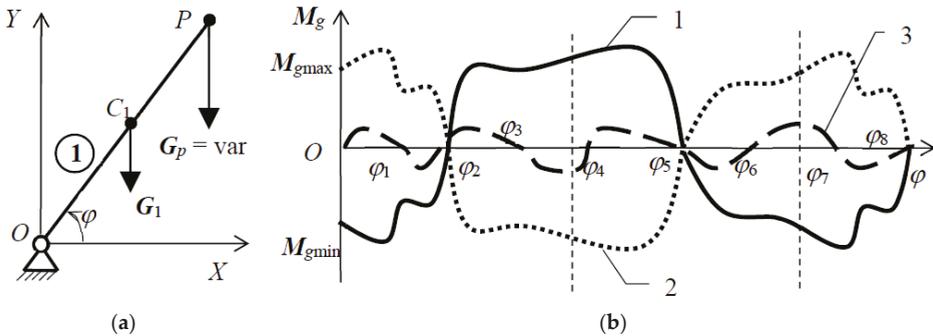


Figure 1. (a) Rocker arm (b) Gravitational moment variation of the variable weight forces.

Many other mechanical systems, which are automatized more and more in these days, becoming in this way mechatronic systems, have to overcome variable payloads or resistant forces during the functioning. Beside the manipulation robots used in palletizing for example [5–9], articulated cranes [10–12] and pump-jack oil pumps [13–15], a large category of ergonomic manipulators [16–18] are facing the variable payload and have to adapt to this condition.

By balancing, another moment which is opposing the load moment (Figure 1b—dotted curve line 2) should be induced in order to compensate or eliminate the effect of load. If the difference between the load moment and the balancing moment is zero, then the system is perfect (exact) balanced in all positions from its work field [19]. If there are only some positions where the difference is zero (Figure 1b—discontinuous curve line 3) then an approximate balancing is obtained [20].

In order to compensate the effects of static loads that depend to displacements, then forces which depend also to displacements should be used. The main candidates are the weight forces represented by counterweights and the elastic forces of springs or gases. Industrial robots are using both of these solutions (Figure 2) for example ABB industrial robot of IRB 6499 RF model [6].

Even in the case of static balancing by using counterweights the overall mass of the mechanical system is increased and from dynamics point of view the situation could become worse than in the case the mechanical system is even unbalanced, this solution is still useful and widely used in engineering because of the simplicity and for mechanical systems which are manipulating large loads, and which are operating at low or moderate dynamics.

As for the adaptation to the variation of the payload in case of robots and automatized mechanical systems many ideas are proposed in scientific papers or patents passed years [2,21–24] but no realization in practice for these domains. If there are some adjustments in order to adapt to the variation of payload they are made off-line [6,9,21] while mechatronic system is in a rest state. On the

other hand if the variation is not very large then an approximate simple solution is solving the problem in practice. Anyway the difficulty of static balancing comes from the variation of the payload and in the case of spatial mechanisms of robots or other mechatronic systems became more difficult due to the complexity of the movements.

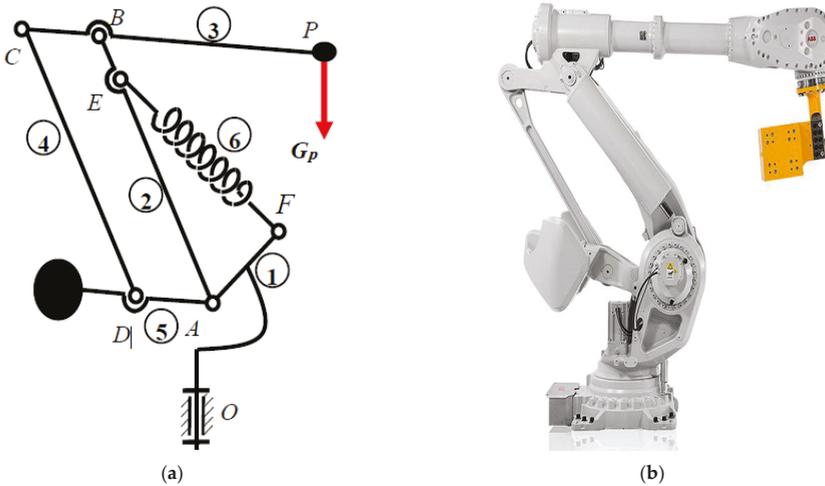


Figure 2. Industrial robot static balanced by counterweight and spring: (a) kinematic schema; (b) picture of industrial robot.

2. Adaptive Balancing by Using Counterweights

The method of adding the counterweights involves the increasing of moving masses, overall size, inertia and the stresses of the mechanism links [20]. Some of the mechanical systems [1] accept this method because of operating at low or moderate dynamics, from safety reasons or in cases where the right spring is difficult and costly to be obtained [2], or the spring balancing solution is too complicated to be fitted to [25]. Anyway, an internal mass redistribution so that parts of mechanical systems (actuators, electric motors, other mechanical transmission, either electric or electronic parts from controlling cabinet which could be relocated on the robot body) to act as counterweights like in the case of industrial robots [9], or as is the case of camera stabilizers [26], is first step when the static balancing problem starts [2].

Variation of gravitational moment given by the weight force of the rocking arm ① (Figure 1a) G_1 and by the variable payload G_p has the expression:

$$M_g(t) = -G_1 OC_1 \cos \varphi(t) - G_p(t) OP \cos \varphi(t) = f_1(t) \cos \varphi(t) \tag{1}$$

where:

$$f_1(t) = c_1 + c_2 G_p(t) \tag{2}$$

with:

$$c_1 = -G_1 OC_1 = \text{const. and } c_2 = -OP = \text{const.} \tag{3}$$

Then the balancing moment should be:

$$M_b = M_b(t) = f_2(t) \tag{4}$$

so that:

$$f_2(t) \cong -f_1(t) \cos \varphi(t) \tag{5}$$

Let suppose the case of the rocking arm ① which is gravity compensated for its weight G_1 and for the weight of the constant part from the variation of payload G_{pc} (Figure 3) by a counterweight mounted fixed on the rocking arm ① at a proper distance on the opposite side then centre of mass C_1 according to origin point O (not represented in the following). In this case the constant c_1 from Relation (3) became:

$$c_1 = -G_1 OC_1 - G_{pc} OP = \text{const.} \tag{6}$$

A variation of static load in linear form (as is represented in Figure 3—dashed line) was studied in Reference [25] and will be taken as comparison in Example section. By taking into consideration some frictions in the mechanical system of payload let suppose the variation of payload is known and cyclic with a symmetric variation of second degree evolution during one period of time T (Figure 3):

$$G_p(t) = G_{pc} + 4 \frac{G_{pv,\max}}{T} t - 4 \frac{G_{pv,\max}}{T^2} t^2, \text{ where } t \in [0, T] \tag{7}$$

In order to gravity compensate the variable component G_{pv} by using also a supplementary counterweight then 2 possibilities could be taken into consideration: a variable weight of the additional counterweight or a movable counterweight with a fixed weight.

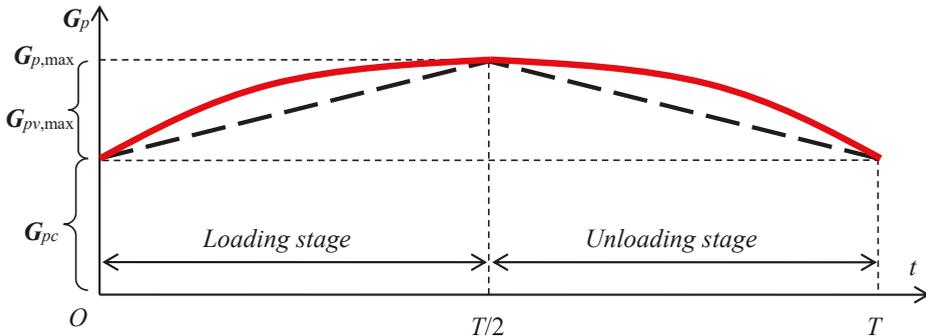


Figure 3. Parabolic variation of a cyclic payload.

To make a variable weight for the counterweight is not impossible but is complicated and in order to compensate a continuous variation then liquid weights are needed, which are complicating much more the system and the dynamics became also very important. From practical point of view the changing of the location of the additional counterweight on the balanced element (as is the studied rocking arm ① in Figure 1a) is a feasible solution when the speeds and accelerations are not very high.

There are also 2 possible ways of moving the additional counterweight ② relatively to the balanced element: by translating onto it (Figure 4a without bar ③) or by rotating around a point which is becoming a joint on it by using an additional bar (Figure 4b without bar ③).

Despite of the pretentious prismatic joint the solution with translating counterweight became very popular [4,15] due to the better dynamics of the multi-body system and due to the simplicity of the transmission of the supplementary actuator.

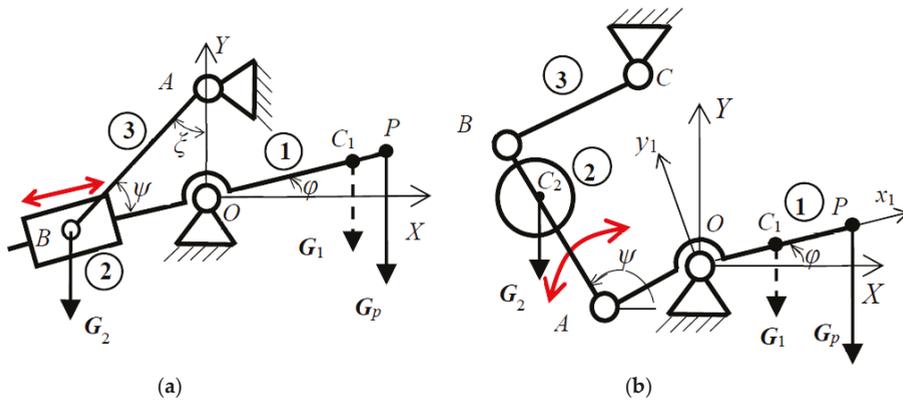


Figure 4. Movable counterweight in order to compensate variable payload: (a) translational counterweight; (b) rocking counterweight.

In case of a known cyclic variation of payload, as it is represented in Figure 3, then a passive adaptive solution is possible to be used that is, a solution which does not require supplementary energy added from the outside of the mechanical system. The simplest solution is presented in Reference [15] by linking the counterweight ② to the mechanism base through a simple bar denoted by ③ and connected by two joints as is shown in Figure 4a.

In Figure 4a is presented the symmetric solution which is leading to a reduced number of exact balancing positions (maximum three). In this case the gravitational moment which has to be compensated is:

$$M_g(t) = -G_{pv}(t) OP \cos \varphi(t) = c f_3(t), \tag{8}$$

where:

$$G_{pv}(t) = G_p(t) - G_{pc} \tag{9}$$

$$c = -4 OP \frac{G_{pv,max}}{T} \tag{10}$$

and:

$$f_3(t) = t \left(1 - \frac{t}{T}\right) \cos \varphi(t) \text{ where } t \in [0, T], \tag{11}$$

The balancing moment M_b of counterweight ② has the expression:

$$M_b(t) = G_2 OB(t) \cos \varphi(t), \tag{12}$$

where in the weight G_2 could be count as added the part of the weight of the connecting bar ③ concentrated in point B because is fixed one (Figure 3a).

The position of the counterweight on the balanced arm ① has the expression:

$$OB(t) = \sqrt{AB^2 - OA^2 \cos^2 \varphi(t)} - OA \sin \varphi(t) \tag{13}$$

or:

$$OB^2(t) = OA^2 + AB^2 - 2 OA AB \cos \varphi(t) \tag{14}$$

where:

$$\zeta = \frac{\pi}{2} - \varphi(t) - \psi(t) \text{ and } \sin \psi(t) = \frac{OA}{AB} \cos \varphi(t) \tag{15}$$

Unbalancing moment M_u is given by relation:

$$M_u = M_b + M_g, \tag{16}$$

and by comparing relations (12) and (13) with (8)–(11) is obvious that unbalancing cannot be zero which is anyway shown in Example section. But the unbalancing is better than in the case of linear variation of payload at the same conditions.

As for the solution from Figure 4b, with the rocking counterweight, the balancing is also approximate. The position of bar BC with respect to reference system XOY has a more complicated form (resulted by the solving of positional kinematics of RRR dyad composed by elements AB and BC) because it depends to:

- the position of points A and B;
- the length of bars BC and AB.

Analytic solving (and numerical one too [27]) leads to two mathematical solutions from kinematics but only one is correct from balancing point of view, the one when $\pi/2 < |\psi| < \pi$.

General and non-symmetric solution from Figure 4b requires an optimization solving too with following design variables:

- coordinates of points $A(x_{1A}, y_{1A})$ and $C(X_C, Y_C)$;
- lengths of connecting bars AB and BC;
- position AC_2 of the counterweight on the bar ② and the mass of the counterweight m_2 .

3. Adaptive Balancing by Using Springs

There are many papers and patents [1,2] which studied during the time the problem of static balancing by using springs. Most of them consider the problem when the static load is constant and more of that do not take into consideration the spring mass. In any case the simplest and general solution of articulating a spring between the balanced arm and the ground (Figure 5) is leading to an approximate solution [20].

In this way a better idea is to join the spring to the balanced arm so that its weight concentrated in the joint to act as a counterweight too (Figure 5). But as is wrote in many papers and even from the beginning started by Carwardine [28], the solution from Figure 5 requires zero-free-length springs [25] or zero-free-length elastic systems [29].

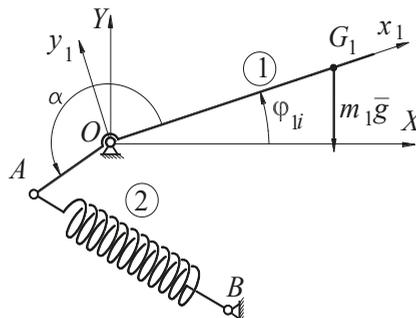


Figure 5. Static balancing by spring.

One of the solution is to remove one of the spring joints and to intercalate some linkages with zero degrees of freedom [20,25]. In case of variable load this solution requires to intercalate linkages with active joints in order to obtain the required adaptation. In Reference [30] is proposed a solution with active prismatic joints. Also in paper [21] is proposed a more complicated solution with prismatic

joints. A simple solution with only one adjusting of a joint of spring on vertical direction is proposed in Reference [22]. The adjusting of the other joint of spring (the one joined to the balanced movable arm) is proposed in Reference [23]. Authors of paper [24] propose an interesting idea to adjust the position of the both joints by using the simultaneous adjusting in order to not consume supplementary energy from the outside.

Prismatic joints are always more complicated from maintenance point of view and not only. So revolute joints are more proper and in Figure 6 are represented solutions to relocate spring joints by using active controlled joints.

Joint C and D are only controlling active joints. Once the adaptation to the variable load G_p is done then joint A and joint B respectively, are fixed to the arm and to the ground respectively.

Mixed solution with prismatic and revolute joints as active control joints are presented in Figure 7.

Let take as example the simple one degree of freedom relocation of fixed joint B by a prismatic joint presented in Figure 8.

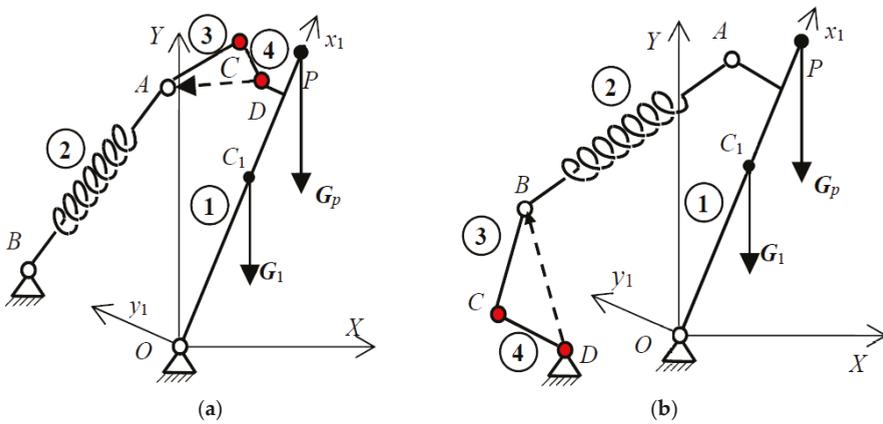


Figure 6. Adjusting mechanisms with revolute pairs: (a) Controlled relocation of joint A of spring; (b) Controlled relocation of fixed joint B of spring.

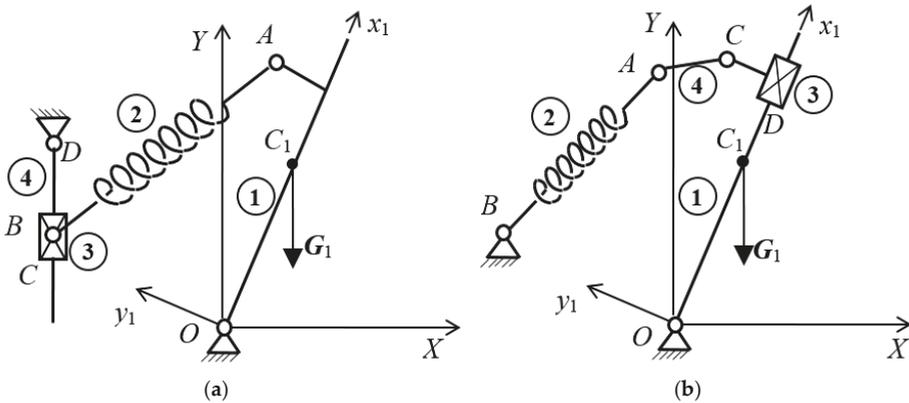


Figure 7. Adjusting mechanisms with revolute and prismatic pairs: (a) Controlled relocation of fixed joint B of spring; (b) Controlled relocation of joint A of spring.

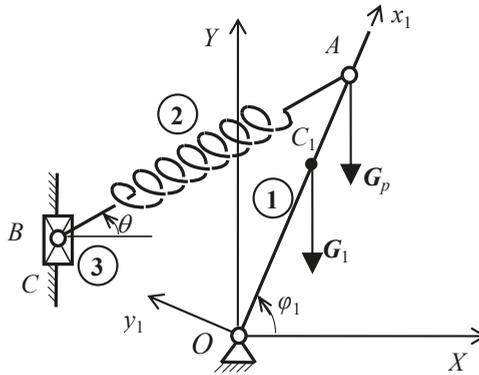


Figure 8. Controlled relocation of fixed joint B of spring by only one active prismatic joint.

Without any reduction of the generality of the study let consider joint A on Ox_1 axis and also the point of action of payload in same point A. In this case the equilibrium equation of rocking arm ① is given by equation:

$$F_s OA \sin(\theta - \varphi_1) - M_{g1} = 0 \tag{17}$$

where:

$$M_{g1} = (m_1 OC_1 + m_p OA) g \cos \varphi_1 \tag{18}$$

Force of spring is:

$$F_s = F_{s0} + k (l_s - l_{s0}), \tag{19}$$

and:

$$\theta = \operatorname{atan} \frac{Y_A - Y_B}{X_A - X_B}, \left(\begin{matrix} X_A \\ Y_A \end{matrix} \right) = \begin{pmatrix} \cos \varphi_1 & -\sin \varphi_1 \\ \sin \varphi_1 & \cos \varphi_1 \end{pmatrix} \begin{pmatrix} x_{1A} \\ 0 \end{pmatrix}, \tag{20}$$

$$OA = \sqrt{X_A^2 + Y_A^2}, l_s = AB = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2}, \tag{21}$$

Are known or considered known at the beginning of the synthesis problem:

- force of spring F_{s0} corresponding to the length of spring l_{s0} ,
- coordinates: x_{1A} , X_B and Y_B .

When a modification of payload occurs then:

$$G_p' = G_p + \Delta G_p \text{ or } m_p' = m_p + \Delta m_p \tag{22}$$

According with this modification the Y-coordinate of point B should be changed by controlling the system:

$$Y_B' = Y_B + \Delta Y_B \tag{23}$$

Accordingly Relations (19)–(21) will became:

$$F_s' = F_{s0} + k (l_s' - l_{s0}) = F_s + k \Delta l_s, \tag{24}$$

$$l_s' = l_s + \Delta l_s = AB' = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B')^2}, \tag{25}$$

$$\theta' = \operatorname{atan} \frac{Y_A - Y_B'}{X_A - X_B}, \tag{26}$$

and new balancing equation:

$$F_s' OA \sin(\theta' - \varphi_1) - M_{g1} - \Delta M_g = 0 \tag{27}$$

where:

$$\Delta M_g = \Delta m_p g OA \cos \varphi_1 = \Delta G_p OA \cos \varphi_1 \tag{28}$$

Due to nonlinearity of Equation (27), comes from Relations (24), (25) and (26), it is impossible to get an explicit relation like:

$$Y_B' = Y_B'(m_p(t)) \tag{29}$$

or

$$\Delta Y_B = \Delta Y_B (\Delta m_p(t)) \tag{30}$$

which is necessary to the control. Only by using a numerical method could solve this problem.

4. Example

In the case of solution from Figure 4a let suppose that the variable part of payload has the maximum value $G_{pv,max} = 4$ N (Figure 3) and is acting at distance $OP = 2$ m while the work space of balanced arm ① is symmetric with respect to the horizontal axis: $\varphi \in [-\pi/2, \pi/2]$. Suppose that the counterweight ② has the weight $G_2 = 3$ N and the connecting road ③ has the length $AB = 2$ m and is articulated on vertical direction at distance $OA = 1$ m.

By taking into consideration a parabolic variation of payload as is represented in Figure 3 (red curve) the maximum unbalancing moment is when the position of balanced arm ① is near the horizontal ($\varphi = 0.095$ [rad]) and has the magnitude of 0.828 Nm (represented by function $h_2(x)$ plotted in red color in graph from Figure 9).

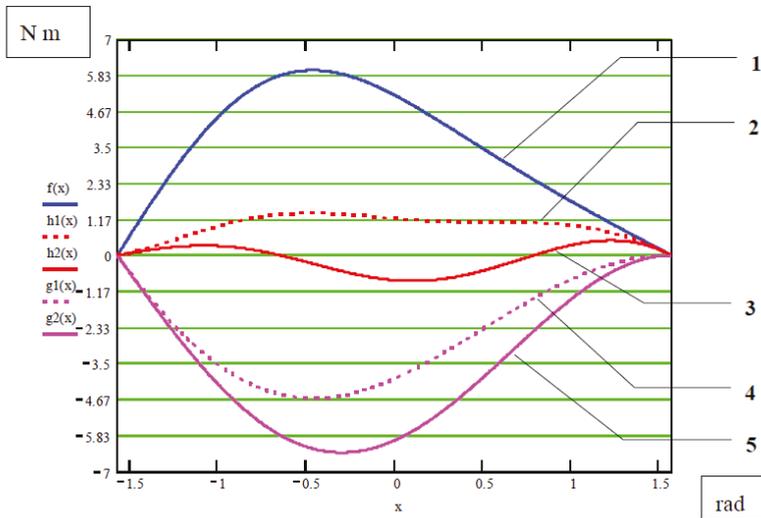


Figure 9. Gravitational moment, unbalanced moments and counterweight balancing moments.

The plotted red dashed curve—represented by function $h_1(x)$ in Figure 9—show the variation of unbalancing moment in case o linear variation of static load [30] which has the maximum value about double than in case of parabolic variation (about 1.4 Nm at position $\varphi = -0.5$ rad).

Better balancing obtained in the case of parabolic variation of payload is expressed also by efficaciousness coefficient [19] with an extended definition in Reference [30]:

$$\varepsilon = 1 - \frac{E_b + E_a}{E_u} \quad (31)$$

where:

E_b —energy consumed by actuating system of the balanced robot;

E_a —supplementary energy consumed by an additional actuating system in order to obtain active balancing of robot;

E_u —energy consumed by actuating system of the unbalanced robot.

In the case of passive balancing the supplementary energy E_a is zero and in present example by comparing the other energies without taking into the consideration the frictions the following efficaciousness coefficient are obtained:

$$\varepsilon_{\text{linear}} = 0.632031$$

$$\varepsilon_{\text{parabolic}} = 0.973446$$

More of that the unbalancing moment in the case of parabolic variation of the payload (red solid curve 3 in Figure 9) has a variation like in Figure 1b, up and down with respect to the horizontal line of perfect balancing, which is a normal and expected variation in case of approximate balancing [20].

5. Conclusions

In case of the counterweight balancing two new solutions are proposed in Figure 4 (both passive) in order to be used in Robotics and Mechatronics fields.

Passive solution from Figure 4a with a single bar joined between translational counterweight and the ground leads to a surprisingly good balancing with respect to the simplicity of the solution demonstrated by a simulation (Figure 9). On the other hand the solution is compensating also the variation of the payload (Figure 3) while the functioning domain is very large from -90° to $+90^\circ$ with respect to horizontal plane. In same conditions the compensation of parabolic variation of payload is better than the compensation in the case of linear variation of payload demonstrated either by simulation results (Figure 9) and by the efficaciousness coefficient defined in Relation (31). Complementary passive solution with a rocking counterweight (see Figure 4b) is used in practice for symmetric solution [10–12] but never seen for the general case as is proposed in Figure 4b.

As for the active solutions with counterweight in papers [30,31] are proposed two complementary solutions (translating and rocking) that are using as counterweight the actuating motor of the balanced arm and even the mechanical transmission could be used as counterweight if the weight of actuating motor is not enough (as are added in the case of camera stabilizers [26] the camera battery and the monitor like additional counterweights).

In case of robots it is also possible to remove electric parts from power and control cabinet (like frequency converters for example) and to mount them as counterweights on the mechanical arm. In this case 2 thick cables (with power and with feed-back information) are replaced by only one power cable.

As for the elastic balancing in the case of variable payload four new active solutions with 2 DOF for joints relocation are presented in Figures 6 and 7. Mathematical model of a related solution, with only one degree of freedom for relocation by translation of the ground joint of the spring (Figure 8), is formulated. The advantages of using springs for static balancing are countered in this case of variable payload by the complicated solutions needed for relocation of the spring's joints.

During the functioning some pressure angles or singularity positions may occur in solutions from Figure 4 for a certain kinematic dimensions set, which is limiting the static balancing operation and

this will be next study together with the solving of an optimum problem as is announced. As for the solutions with springs the limitation is higher due to the real spring itself as is presented in paper [25]. Starting by this study, additional studies could be made by going to different directions and by taking into account many other subsidiary problems which is closing better to the reality.

Funding: This research received no external funding.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Ciupitu, L.; Simionescu, I.; Lee, C.-C. Static Balancing—An Overview. In Proceedings of the First Asian Conference on Mechanism and Machine Science 2010, Taipei, Taiwan, 21–25 October 2010; pp. 250084-1–250084-8.
2. Ciupitu, L. Active Static Balancing of Mechatronic Systems—An Overview. *Appl. Mech. Mater.* **2015**, *811*, 253–259. [CrossRef]
3. Segla, Š.; Ciupitu, L.; Reich, S. Optimization of a Spring Balancing Mechanism for Parallelogram Robot Mechanisms. *Mech. Manip.* **2006**, *5*, 43–48.
4. Vukov. Industrial Robot APR 20. Available online: <http://www.vukov.sk/index.php?lang=en> (accessed on 5 November 2018).
5. McCormick, J. Method and Apparatus for Oil Well Pumping. U.S. Patent 6386322 B1, 14 May 2002.
6. ABB. Industrial Robots IRB 6499 RF. Available online: <https://new.abb.com/products/robotics> (accessed on 5 November 2018).
7. FANUC Inc. Industrial Robots M-900 Series. Available online: <https://www.fanuc.eu/hr/en/robots/robot-filter-page/m-900-series> (accessed on 5 November 2018).
8. Kawasaki Ltd. Industrial Robots MX500. Available online: <https://global.kawasaki.com/> (accessed on 5 November 2018).
9. KUKA GmbH. Industrial Robots IR 160/60 and IR 601/60. Available online: <https://www.kuka.com/en-de> (accessed on 5 November 2018).
10. E-Crane Equilibrium. E-Cranes. Available online: <https://www.e-crane.com/> (accessed on 5 November 2018).
11. METSO Minerals Industries. Metso Balance Crane (MBC). Available online: <https://www.metso.com/> (accessed on 5 November 2018).
12. SeNNebogen. E Series. Available online: <https://www.sennebogen.com/en.html> (accessed on 5 November 2018).
13. Lufkin Automation (Baker Hughes, a GE Company). Sam Well Manager—Rod Pump Controller. Available online: <https://www.westermo.com/industries/oil-and-gas/success-stories/lufkin> (accessed on 5 November 2018).
14. ICM-Electronics. Sucker Rod Pump Control in Oil Wells. Available online: <http://www.icm.rs/> (accessed on 5 November 2018).
15. UPETROM “1 Mai” S.A. Experimental Sucker Rod Pump with Improved Balancing System. Available online: <http://www.researchforindustry.ro/site/viewCompany/282> (accessed on 5 November 2018).
16. AIKOKU Alpha Corp.-Raku-Raku Hand Division. Ergonomic Manipulators Series. Available online: <http://www.aikoku.co.jp/en/> (accessed on 5 November 2018).
17. AUTOMATICA S.A. MB-150 (Ergonomic Manipulator). Available online: <https://automatica.ro/en/> (accessed on 5 November 2018).
18. Perreault, S.; Cardou, P.; Gosselin, C. Approximate static balancing of a planar parallel cable-driven mechanism based on four-bar linkages and springs. *Mech. Mach. Theory* **2014**, *79*, 64–79. [CrossRef]
19. Simionescu, I.; Ciupitu, L. The Static Balancing of Industrial Robot Arms. Part II: Continuous Balancing. *Mech. Mach. Theory J.* **2000**, *35*, 1299–1311. [CrossRef]
20. Simionescu, I.; Ciupitu, L. The Static Balancing of Industrial Robot Arms. Part I: Approximate Balancing. *Mech. Mach. Theory J.* **2000**, *35*, 1287–1298. [CrossRef]

21. Briot, S.; Arakelian, V.H. A New Energy-Free Gravity-Compensation Adaptive System for Balancing of 4-DOF Robot Manipulators with Variable Payloads. In Proceedings of the 14th IFToMM World Congress, Taipei, Taiwan, 25–30 October 2015. [CrossRef]
22. Paz, R.A.; Barajas, J.C.; Ma, O. Autobalancing Control for a Reduced Gravity Simulator. In Proceedings of the 2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Wollongong, Australia, 9–12 July 2013; pp. 405–410, ISBN 978-1-4673-5320-5/13.
23. Masouleh, M.T.; Homayounpour, M. Static Balancing of Parallel Planar 3-DOF Mechanisms, 3-RRR, 3-PRR, 3-RPR and static balancing of variable weights. *Modares Mech. Eng. Sci. Res. Mon. J. Tarbiat Modares Univ.* **2015**, *14*, 321–331.
24. Wisse, B.M.; van Dorsser, W.D.; Barrents, R.; Herder, J.L. Energy-free Adjustment of Gravity Equilibrators Using the Virtual Spring Concept. In Proceedings of the 2007 IEEE 10th International Conference on Rehabilitation Robotics, Noordwijk, The Netherlands, 12–15 June 2007; pp. 742–750.
25. Ciupitu, L. Optimum Design of Balancing Systems with Real Springs. *Appl. Mech. Mater.* **2014**, *555*, 593–598. [CrossRef]
26. Available online: <https://www.instructables.com/id/How-To-Make-an-Impromptu-Camera-Stabilizer/> (accessed on 21 October 2018).
27. Pelecudi, C.; Simionescu, I.; Ene, M.; Moise, V.; Candrea, A. *Probleme de Mecanisme*; Editura Didactică și Pedagogică: Bucuresti, Romania, 1982.
28. Carwardine, G. Improvements in Elastic Force Mechanisms. UK Patent 377.251, 21 July 1932.
29. Ciupitu, L.; Simionescu, I. Zero-free-length Elastic Systems for Static Balancing. In *New Advances in Mechanisms, Mechanical Transmissions and Robotics*; Springer International Publishing: Berlin, Germany, 2017; pp. 59–67, ISBN 978-3-319-45449-8.
30. Ciupitu, L. Adaptive Balancing of Robot Mechanisms System. In Proceedings of the IXth International Conference on Product Design, Robotics, Advanced Mechanical and Mechatronic Systems and Innovation PRASIC 2016, Braşov, Romania, 10–11 November 2016; pp. 77–82.
31. Ciupitu, L.; Vladareanu, L. Adaptive Balancing by Counterweights of Robots and Mechatronic Systems. *Int. J. Model. Optim.* **2018**, *8*, 178–182. [CrossRef]



© 2018 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Design and Implementation of a Dual-Axis Tilting Quadcopter

Ali Bin Junaid ¹, Alejandro Diaz De Cerio Sanchez ², Javier Betancor Bosch ², Nikolaos Vitzilaios ³ and Yahya Zweiri ^{2,4,*}

¹ Department of Mechanical Engineering, KU Leuven, 3000 Leuven, Belgium; ali.binjunaid@kuleuven.be

² Faculty of Science, Engineering and Computing, Kingston University London, London SW15 3DW, UK; aldiazde@gmail.com (A.D.D.C.S.); J.Betancorbosch@kingston.ac.uk (J.B.B.)

³ Department of Mechanical Engineering, University of South Carolina, Columbia, SC 29208, USA; VITZILAIOS@sc.edu

⁴ Khalifa University Center for Autonomous Robotic Systems, Department of Aerospace Engineering, Khalifa University of Science and Technology, P.O. Box 127788 Abu Dhabi, UAE

* Correspondence: Y.Zweiri@Kingston.ac.uk; Tel.: +44-020-8417-4846

Received: 7 August 2018; Accepted: 18 October 2018; Published: 20 October 2018

Abstract: Standard quadcopters are popular largely because of their mechanical simplicity relative to other hovering aircraft, low cost and minimum operator involvement. However, this simplicity imposes fundamental limits on the types of maneuvers possible due to its under-actuation. The dexterity and fault tolerance required for flying in limited spaces like forests and industrial infrastructures dictate the use of a bespoke dual-tilting quadcopter that can launch vertically, performs autonomous flight between adjacent obstacles and is even capable of flying in the event of the failure of one or two motors. This paper proposes an actuation concept to enhance the performance characteristics of the conventional under-actuated quadcopter. The practical formation of this concept is followed by the design, modeling, simulation and prototyping of a dual-axis tilting quadcopter. Outdoor flight tests using tilting rotors, to follow a trajectory containing adjacent obstacles, were conducted in order to compare the flight of conventional quadcopter with the proposed over-actuated vehicle. The results show that the quadcopter with tilting rotors provides more agility and mobility to the vehicle especially in narrow indoor and outdoor infrastructures.

Keywords: aerial robotics; quadcopters; UAVs; dual-tilting; tilting rotors; over-actuation; flight control; rotorcraft

1. Introduction

Over the last few decades, the usage and deployment of UAVs (Unmanned Aerial Vehicles) have been growing, from hobby to military applications. Some of those applications include surveying, maintenance and surveillance tasks, transportation and manipulation, search and rescue [1,2]. Rotorcraft UAVs are of particular interest as they offer advanced capabilities such as Vertical Take Off and Landing (VTOL) and high agility. Quadcopters are the most researched and used platforms in this area.

A quadcopter's lift and thrust is generated by four propellers mounted on high-speed, high-power brushless DC motors. Quadcopters use an electronic control system and electronic sensors to stabilize themselves. With their small size and VTOL capability, quadcopters can fly indoors as well as outdoors. Similar to a conventional helicopters, quadcopters can hover but have significant other advantages such as ease of piloting and mechanical simplicity. Recently, there is a rapid growth in quadcopter development due to the high potential used in numerous commercial applications.

For real world applications, quadcopters require more payload capacity and should be more invulnerable and robust towards external disturbances. However, increased payload capacity demands up scaling the platforms which eventually results in decreased maneuverability and agility [3]. As stated in [3], the inertia of the platform is increased and requires larger control moments to achieve higher agility with the increase of the vehicle size. Secondly, the increased weight results in increased propeller size consequently increasing the inertia. The conventional quadcopter possesses such physical constraints with its control on larger scales.

Conventional quadcopters are under-actuated mechanical systems possessing less control inputs than Degrees Of Freedom (DOFs). Over the last decades, different control techniques have been proposed to deal with the quadcopter under-actuation for an effective and more controlled performance [3–5]. Still, the under-actuation of quadcopter has limitations on its flying ability in free or cluttered space.

In a conventional under-actuated quadcopter, actuators failure results in complete destabilization of the vehicle as its control is completely dependent on the symmetry of the lift. To alleviate this problem, several approaches have been proposed in previous studies. In [6], propellers with variable pitch and shifting of the Center of Gravity (COG) are suggested. These approaches enable to achieve the controllability of the vehicle in roll and pitch axes up to some extent. However, the yaw axis still remains uncontrollable. In [7], a bounded control law was proposed in order to have a safe landing for quadcopter in case of actuators failure.

One approach to overcome this issue is to increase the number of rotors. Typical example includes 4Y octocopters [8]. This approach has advantages such as the mechanical simplicity and reliability. However, this approach results in increased weight and increased inertia hence reducing the agility of the vehicle. Furthermore, increased number of rotors results in larger power consumption which immensely impacts the endurance of the aircraft. Nevertheless, standard hexarotors are not fail-safe multi-rotor platforms and cannot hover with five propellers [9].

Apart from the increasing number of rotors approach, other approaches include variation in the types of actuators keeping in consideration the key factors which are the vehicle's size and weight. Cutler et al. [10] showed an effective approach by using propellers with variable pitch. In this approach, while keeping the weight down, the bandwidth of its actuators was increased. However, actuator failure resulting in instability still remains an issue.

Other solutions to overcome the under-actuated problem include tilt-wing mechanisms [11], UAVs with non-parallel fixed thrust directions [12], or tilt-rotor actuations [13]. Similar approach with the dual-axis tilting of the rotors providing the broad range of control bandwidth for the same number of rotors is proposed in [14]. However, most of the platforms were developed and tested in indoor environments with low payload capacities. Furthermore, indoor navigation systems were used to develop the control of the tilting quadcopter. Similar concept of rotor tilting is used in [15,16], however, rotor tilting is limited to single axis only.

In [17], Control Moment Gyroscopes (CMG) were proposed to increase the control system bandwidth by merging a thrust vectoring approach with additional flywheels in order to be used as CMG, and a vane system for thrust vectoring. However, the extra weight of the flywheels and the thrust vectoring vane system results in increased weight of the aircraft and complicates the design of the vehicle. Gress [18] used Opposed Lateral Tilting (OLT) technique for using the gyroscopic effects for governing the pitch attitude of aircraft, using the propellers as gyroscopes. In [19], OLT proved to achieve higher controllability. The detailed model and control strategy for hovering, with experimental evidence is presented in [20]. To implement such control strategies and actuation mechanisms, it is important to keep in mind key factors for the development of such vehicles, which are weight, mechanical simplicity, cost-effectiveness and ability to manufacture the platform in less time.

Nowadays, advances in the fields of Computer Aided Design (CAD) and Rapid Prototyping (RP) have provided the tools to rapidly generate a prototype from a concept. RP technique allows to automatically construct physical models using additive manufacturing technology [21,22]. Mechanical parts or assembly can be quickly manufactured using 3D CAD design. This technique has

emerged as an innovative tool to reduce the time and cost of manufacturing and fabrication by creating 3D product directly from computer aided design providing the ability to perform design validation and analysis [23,24].

This paper presents design and implementation of a novel actuation strategy which was proposed by co-author in [25] in order to increase the agility and control bandwidth of the conventional quadcopters in outdoor scenarios.

The introduction of dual-axis tilting to the propellers produces an over-actuated quadcopter with ability to have 12 control inputs with 6 control outputs. The tilting of propellers provides the necessary gyroscopic effects to provide fast control and action. The proposed tilting rotor solution for quadcopter uses additional 8 servomotors that allow the rotors to tilt in both axes, an over-actuated system can potentially track an arbitrary trajectory over time. As shown in [26,27], with single axis tilting, full controllability over the quad-rotor position and orientation provides possibility of hovering in a tilted configuration. The research here focuses on the design of such platform and conduct initial experiments to test the actuation modules in outdoor urban environment.

The development of the proposed concept is mainly based on arm design of the quadcopter in which each arm is able to generate three-actuation independently, including the rotors to achieve differential thrusting and dual tilting mechanism to provide broad range of control bandwidth. The computer-aided design (CAD) model was designed and analyzed using finite element analysis (FEA) for the structural rigidity and stability. The experiments were performed using the developed platform which achieved full controllability of the quadcopter thus transforming the system into an over-actuated machine. Flight tests were performed in a trajectory having sharp corners between adjacent obstacles in order to compare the behavior of conventional quadcopter configuration and the proposed actuation strategy. The results show that the quadcopter with tilting rotors provides more agility and mobility to the vehicle especially in narrow indoor and outdoor infrastructures.

The paper layout is as follows: first the design approach for the development of quadcopter with over-actuated mechanism along with its electronics to control the mechanisms is presented. Secondly, the modeling and simulation results are presented. Furthermore, the structural analysis of the rapid prototyped parts is presented ensuring the structural stability of the platform. The flight test results of the developed platform for the conventional and over-actuated configuration are presented. Finally, the results are discussed and some conclusions are drawn.

2. Design Approach

Two servomotors were used to achieve the dual tilting actuation with each rotor. The platform was designed in SolidWorks and manufactured using a ZORTRAX 3D printer [28]. The mechanical design of the quadcopter is shown in Figure 1.

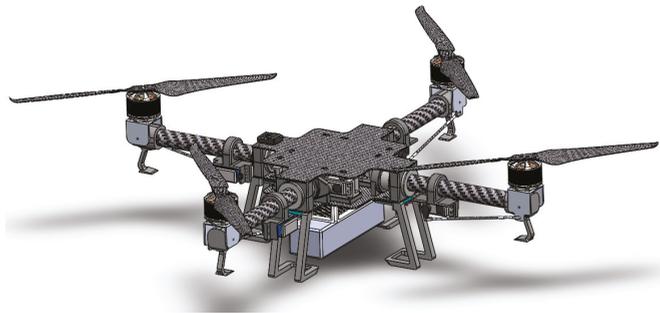


Figure 1. CAD model of proposed quadcopter.

The proposed actuation concept of the quadcopter arms and motors is shown in Figures 2 and 3. The dual tilting mechanism of the arms is mainly based on the servomotor which is coupled with the arm through gears in order to compensate the high torque demand for rotation. This mechanism allows the movement of the whole arm around its axis as shown in Figure 2. Another servomotor is mounted in the arm which connected to the motor mount through push-pull mechanism which rotates parallel to the servo lever as shown in Figure 3. The two angles generated by the servomotors constitute towards the configuration of the rotation axes of the propellers.

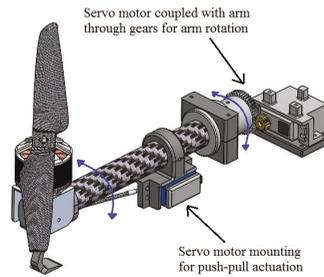


Figure 2. CAD model of proposed dual-tilting arm.

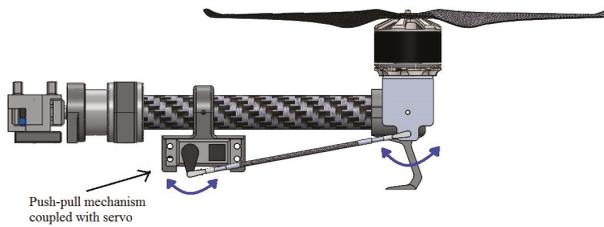


Figure 3. Push-pull mechanism for the tilting of rotor.

The gear mechanism for the coupling of servomotor with the arm of the quadcopter is installed with the gear ratio of 1:3 to fulfill the torque requirement for rotating the arm of the quadcopter. The coupling mechanism is shown in Figure 4. For the actuation of dual-axis tilting system, a total of eight servomotors are used for rotating the arms and motor mountings.

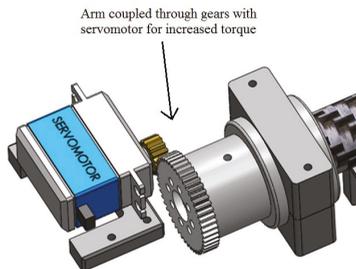


Figure 4. Gear coupling for the tilting of arm.

3. Modeling and Simulation of Over-Actuated Quadcopter

3.1. Modeling

For the development of the mathematical model which defines the dynamics of the dual-axis tilting quadcopter, Newton-Euler formulation has been used to derive the equations. The development process includes the definition of variables and axes reference, formulation using Newton-Euler equations, and including inertia variation [29]. The inertia of the quadcopter has been considered variable due to variation of the rotors position as a result of the tilting. These dynamic equations have been developed using Maple®.

To begin the analysis of the system, three frames have been selected in order to develop the system equations. The first frame is the world frame therefore we refer to it as the frame 1 (World Frame). The next frame is defined to be fixed in the center of gravity of the UAV and is referred as frame 2 (Body Frame). Finally, the third frame is defined to be fixed in each of the rotors and is referred as frame 3 (Rotor Frame), shown in Figure 5. The overall system model diagram is shown in Figure 6. In this paper, kinematics and dynamic modeling of the dual-axis tilting system is developed (Propellers and Tilting block, shown in Figure 6), whereas the conventional quadcopter modeling details are standard text book material therefore omitted here.

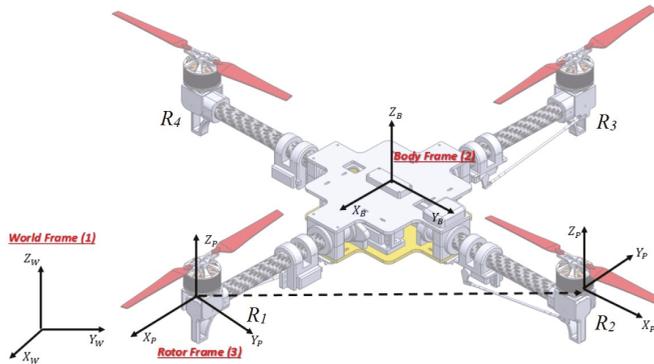


Figure 5. Rotor numbering and reference frames [29].

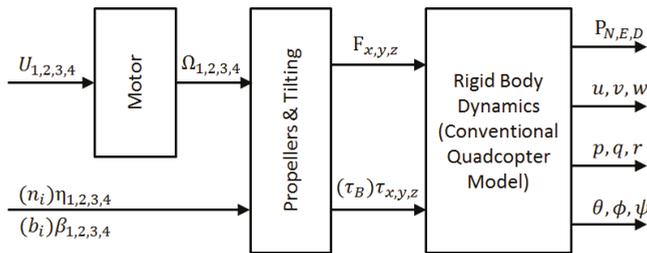


Figure 6. Dual-axis tilting quadcopter modeling diagram.

It is important to set reference frames in each rotor, to get the kinematic and dynamic equations for each rotor in the UAV. To simplify the equations, the four local frames of each rotor have been aligned with respect to the global axis following a rotational relationship with respect to the Z axis by

90° counter-clockwise. The expression which defines the rotation matrix is shown below as example of the rotation for the rotor 2.

$$\begin{bmatrix} i_2 \\ j_2 \\ k_2 \end{bmatrix}_3 = \mathbf{R}_Z\left(\frac{\pi}{2}\right) \cdot \begin{bmatrix} i_1 \\ j_1 \\ k_1 \end{bmatrix}_3$$

Similarly, the rotation matrix is applied for the rotors 3 and 4 with an angular difference of 90° counter-clockwise each, which represents 180° offset for the rotor 3 and 270° offset for the rotor 4. The general rotation matrix for each rotor therefore is defined as:

$$\begin{bmatrix} i_i \\ j_i \\ k_i \end{bmatrix}_3 = \mathbf{R}_Z\left((i-1)\frac{\pi}{2}\right) \cdot \begin{bmatrix} i_1 \\ j_1 \\ k_1 \end{bmatrix}_3 \quad \text{for } i = 1 \dots 4$$

As frame 3 is fixed, every equation is translated from frame 3 to frame 2. As result of translating the local axis of each rotor with respect to frame 2, three matrices for the rotation of each axis are obtained as \mathbf{R}_X , \mathbf{R}_Y and \mathbf{R}_Z .

$$\mathbf{R}_X = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\beta_i(t)) & -\sin(\beta_i(t)) \\ 0 & \sin(\beta_i(t)) & \cos(\beta_i(t)) \end{bmatrix} \quad (1)$$

$$\mathbf{R}_Y = \begin{bmatrix} \cos(\eta_i(t)) & 0 & \sin(\eta_i(t)) \\ 0 & 1 & 0 \\ -\sin(\eta_i(t)) & 0 & \cos(\eta_i(t)) \end{bmatrix} \quad (2)$$

$$\mathbf{R}_Z = \begin{bmatrix} \cos\left(\frac{\pi}{2} \cdot (i-1)\right) & -\sin\left(\frac{\pi}{2} \cdot (i-1)\right) & 0 \\ \sin\left(\frac{\pi}{2} \cdot (i-1)\right) & \cos\left(\frac{\pi}{2} \cdot (i-1)\right) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Finally, the rotation matrix from frame 2 to frame 1 is obtained by three rotation matrices, one in each axis of the frame 1. In this case, the Euler angles are used to identify the orientation of the aircraft. The angles are denoted by θ , φ , ψ corresponding to roll, pitch and yaw and representing the angular displacement of the UAV along the X, Y and Z axes respectively. The rotation matrix from the body to the fixed frame is shown in Equation (3).

$${}^2\mathbf{R}_1 = \begin{bmatrix} c(\theta)c(\psi) & -c(\theta)s(\psi) & s(\theta) \\ s(\varphi)s(\theta)c(\psi) + c(\varphi)s(\psi) & -s(\varphi)s(\theta)s(\psi) + c(\varphi)c(\psi) & -s(\varphi)c(\theta) \\ -c(\varphi)s(\theta)c(\psi) + s(\varphi)s(\psi) & c(\varphi)s(\theta)s(\psi) + s(\varphi)c(\psi) & c(\varphi)c(\theta) \end{bmatrix} \quad (3)$$

where c and s denote to the trigonometric functions *cosine* and *sine* respectively. With these equations the major problem for applying the Newton-Euler equations to our system is sorted. From this point, the calculation of the angular and linear accelerations for the UAV has been done by solving Equation (4). For the calculation of the angular accelerations, the general kinematic equation for the UAV has been used. The Equations (4) and (6) define the angular and linear accelerations of the UAV platform. The motors are rotating the propellers, this represents an angular velocity denoted by Ω .

$$\ddot{\alpha}_B = (I_B + 4I_P)^{-1} \cdot \left[\bar{M}_T - \sum_{i=1}^4 ({}^2\mathbf{R}_3)_i I_P \begin{pmatrix} \ddot{\beta}_i \\ \ddot{\eta}_i \\ \ddot{\Omega}_i \end{pmatrix} + ({}^2\mathbf{R}_3)_i \ddot{\tau}_{ext} - \frac{d}{dt}(I_B)\ddot{\omega}_B - \Theta I_B \ddot{\omega}_B \right] \quad (4)$$

where Θ is the Skew matrix. This is formed by the angular velocity of the platform in three axes [29], it is given as:

$$\Theta = \begin{bmatrix} 0 & -R(t) & Q(t) \\ R(t) & 0 & -P(t) \\ -Q(t) & P(t) & 0 \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \frac{1}{m} \left[({}^1\mathbf{R}_2) \cdot \sum_{i=1}^4 ({}^2\mathbf{R}_{3i} \cdot \vec{T}_{P_i}) \right] \quad (6)$$

Expanding these equations and considering the rotation matrices, the dynamic equations for forces and torques can be obtained as follows:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \begin{bmatrix} (k_{x1} + \lambda_{x1}) & (k_{x2}^* + \lambda_{x2}^*) & (-k_{x3} + \lambda_{x3}) & (-k_{x4}^* - \lambda_{x4}^*) \\ (k_{x1} + \lambda_{y1}) & (k_{x2}^* + \lambda_{y2}^*) & (-k_{x3} + \lambda_{y3}) & (-k_{y4}^* - \lambda_{y4}^*) \\ (k_{z1} + \lambda_{z1}) & (k_{z2}^* + \lambda_{z4}^*) & (-k_{z3} + \lambda_{z4}) & (-k_{z4}^* - \lambda_{z4}^*) \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix}$$

$$\begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} + \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \begin{bmatrix} IP_x \\ IP_y \\ IP_z \end{bmatrix} + \begin{bmatrix} \rho_x \\ \rho_y \\ \rho_z \end{bmatrix}$$

The coefficients of the forces and the torques are given in Appendix A.

It is important to mention that the inertia of the system varies due to the tilting rotors and cannot be assumed as constant (time invariant). Therefore, the inertia matrix (Equation (7)) is modeled in this paper as:

$$I_B = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} + \begin{bmatrix} I_{Px'x'} & 0 & 0 \\ 0 & I_{Py'y'} & 0 \\ 0 & 0 & I_{Pz''z''} \end{bmatrix} + m \cdot \begin{bmatrix} y_C^2 & 0 & 0 \\ 0 & x_C^2 & 0 \\ 0 & 0 & x_C^2 + y_C^2 \end{bmatrix} \quad (7)$$

Expanding Equation (7) yields the inertia values in the main axis which are given in Appendix B.

3.2. Simulation

The simulation of the dynamic equations is implemented using Matlab/Simulink[®] incorporating the motor dynamics, attitude controller for Roll, Pitch and Yaw, and the controller for tilting angles of the rotors (Figure 7). The quadcopter can be simulated to observe the flight behaviour under the influence of different control inputs i.e. attitude commands and tilting rotor angle inputs.

Rectangular path was simulated to observe the performance of UAV using tilting angle of rotors as inputs. In Figure 8, the simulation shows the movement of UAV in a rectangular path with only using the tilting capability of the rotors without changing its attitude. The platform was able to perform sharp cornering maneuvers.

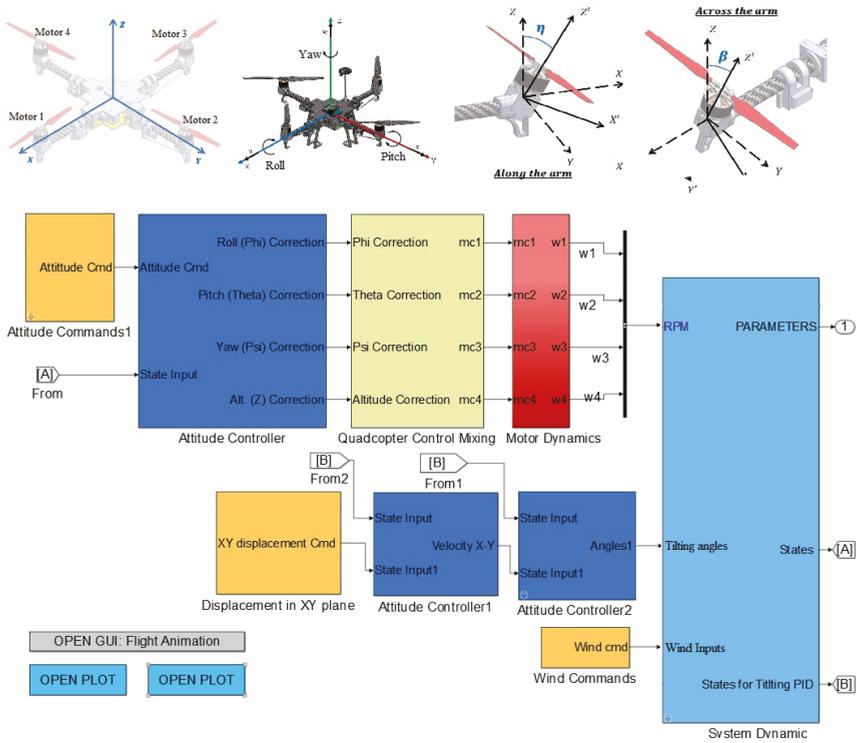


Figure 7. Simulink model.

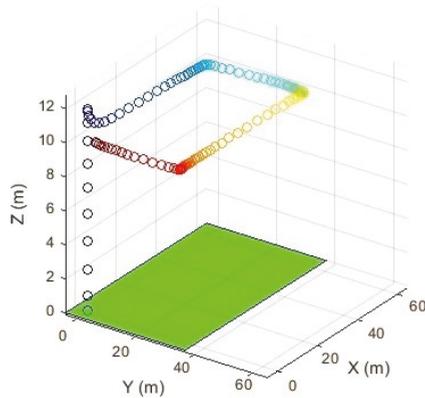


Figure 8. Rectangular path movement of UAV with tilting.

The trajectory was followed only using tilt rotor actuation. The rotor tilting angle along the arm (n) is used to follow a rectangular trajectory. Each rotor is coupled with the opposite rotor with tilting at angles shown in Figure 9. At each corner, the adjacent rotors are tilted to follow the path. Figure 10 shows the attitude of the UAV i.e., Roll and Pitch and it can be observed that the quadcopter performs the maneuver without changing its attitude.

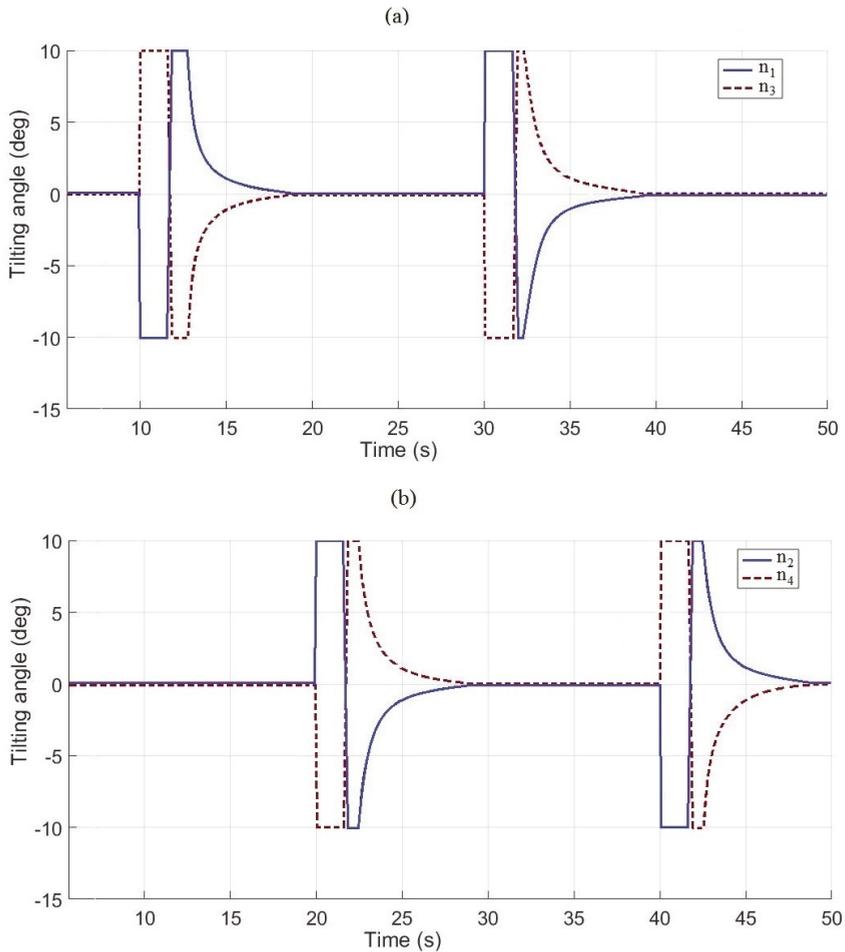


Figure 9. (a) Rotor tilting angles along the arm n_1 and n_3 ; (b) Rotor tilting angles along the arm n_2 and n_4 .

From Figure 9, rotor angles along the arm n_1 and n_3 are tilted at 10 degrees without a change in attitude (as observed in Figure 10) and the quadcopter starts moving in that direction. On the corners, the adjacent rotors (n_2 and n_4) are tilted to follow the trajectory.

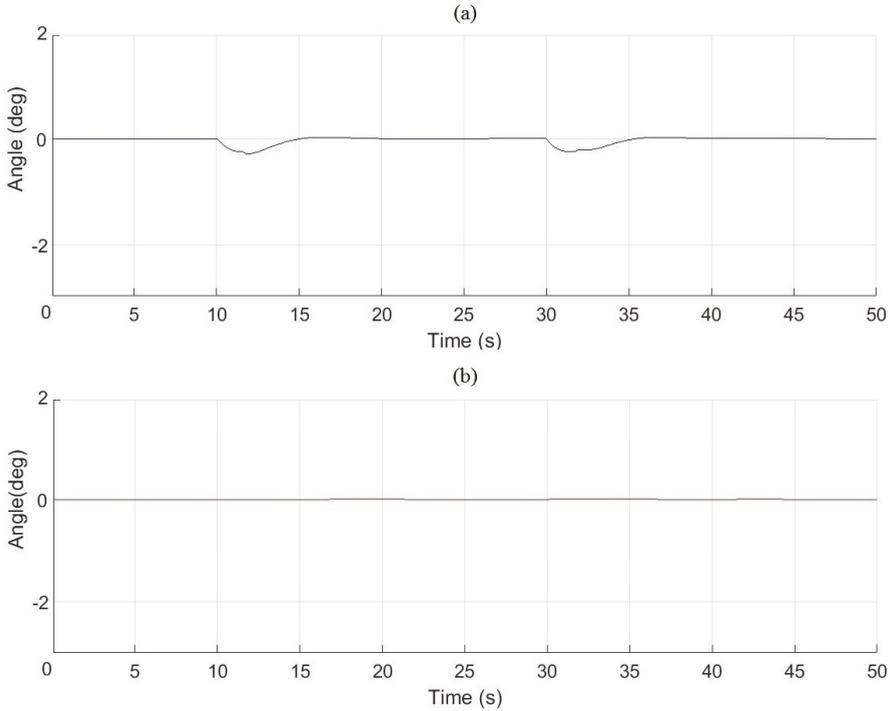


Figure 10. (a) Roll angle of the UAV; (b) Pitch angle of the UAV.

4. Experimental Setup

The four brushless motors selected were the Tiger T-MOTOR MN4014-9 400Kv with a 2-blade 15 × 5 Carbon Fiber propellers as shown in Figure 11.



Figure 11. (Left) Brushless motor; (Right) Propeller used for thrust generation.

For the Flight Control of the UAV, ArduPilotMega (APM) was used. APM is an open-source flight controller, able to control autonomous multicopters, fixed-wing aircraft, traditional helicopters and ground rovers. It is based on the Arduino electronics prototyping platform. Apart from the APM, which is mainly controlling the brushless motors and handling the inner loop control of the vehicle, the Arduino Leonardo microcontroller board is used to control the added servomotors. Figure 12 show the controllers used for the proposed vehicle.

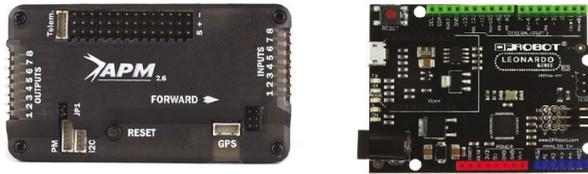


Figure 12. (Left) APM Flight controller; (Right) Arduino Leonardo.

The inner-loop control of the quadcopter is handled by the APM. This on-board inner-loop control device deals with control and stabilization required to hover and perform basic maneuvers. The APM board contains the inertial sensors required for the orientation and heading determination and its software includes the inner-loop control algorithm and the basic Graphical User Interface (GUI) for visualization. Generally, the APM input is provided by the RC remote control which allows the user to manually control and fly the multicopter. The TGY-i10 RC Controller was used as the input for the flight controller. GPS module was attached to the APM in order to navigate the aircraft and track its position for post-processing and analysis. The APM attitude controller was fused with the other controller which was aimed for controlling the inputs for the servo motors responsible for the tilting of the rotors. This controller mainly controls the tilting of the servomotors by combining the inputs for the attitude control of the UAV and the tilting commands for the rotors from the pilot. The commands generated by the human pilot using RC Controller are received by the Arduino Leonardo. The pilot has control of the conventional attitude of quadcopter and tilting servomotors. Based on the control inputs from the pilot, Arduino reads in Pulse Position Modulation (PPM) signals from the RC controller and generates commands (PPM signals) of roll, pitch, yaw and throttle to the APM and commands for servomotors simultaneously. This allows the pilot to have full control of all the actuators of the dual-tilting quadcopter. If the pilot wants to perform maneuvers using only tilting, the Arduino generates commands for the servo motors and proportionally generates attitude commands in order to compensate the tilting actuation.

Currently the tilting angle is limited to 10° for each servomotor. Unlike the conventional quadcopter, this gives the pilot additional control inputs i.e., tilting of the rotors along with conventional control for increased maneuverability. The altitude during all the flight tests was kept in altitude hold mode. The APM flight controller provides the feature of holding the altitude allowing to hover and maneuver at the desired altitude.

The control implementation here is rather basic since a more sophisticated development is out of the scope of this paper but remains in the future work agenda. The conventional inner loop controller of the APM [30] allows the quadcopter to respond to attitude commands of the pilot which are passed through Arduino Leonardo. Arduino Leonardo allows the control of the attitude and the tilting simultaneously. Figure 13 illustrates the experimental setup for control.

After assembling all the manufactured parts and integrating the related modules for the over-actuated mechanisms for the quadcopter, the final product is shown in Figure 14.

Table 1 presents some of the technical specifications of the developed quadcopter.

Table 1. UAV Technical Specifications.

Parameter	Specifications
UAV Dimensions	1048 × 1048 mm
Weight	4 kg
Endurance	20 min
Payload Capacity	2 kg

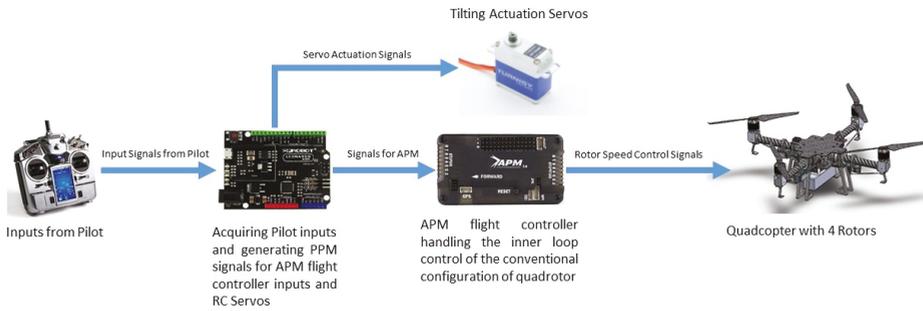


Figure 13. Experimental setup for control of the UAV.



Figure 14. Dual-tilting quadcopter prototype.

5. Results and Discussion

5.1. Flight Test with Conventional Actuation of Quadcopter

After the successful manufacturing and assembly of the vehicle, flight test was conducted for the conventional under-actuated configuration of the quadcopter in order to validate the design parameters and the dynamic model of the vehicle. Relevant flight variables of interest were analyzed in order to observe the behaviour and attitude for the designed vehicle. The results in Figures 15–17 show the control of each axis.

For each axis, the vehicle performs satisfactorily and flies according to the input angles given by the pilot through RC control. GPS is used for the position feedback as ground truth. The proposed system performs well for the conventional configuration and flight controls and validates the design of the platform and its flight stability for the conventional control. Therefore, it can be concluded that the vehicle is stable and able to maintain its attitude. Moreover, the platform can be used for developing and testing of the flight controls for the over-actuated configuration and performs well with proper control techniques for the over-actuated quadcopter design. With the development of over-actuated quadcopter controller, the manufactured vehicle gives plenty of control authority and high maneuverability due to its capability to incorporate the excess number of control inputs.

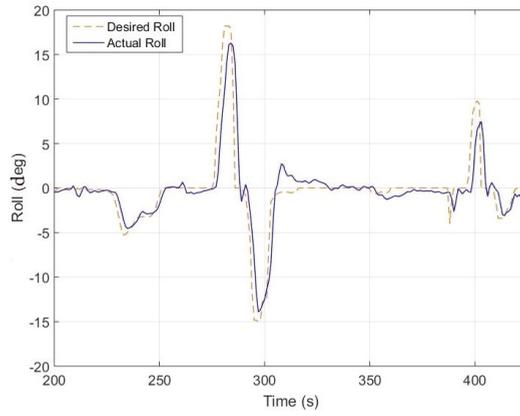


Figure 15. Roll command tracking of the vehicle.

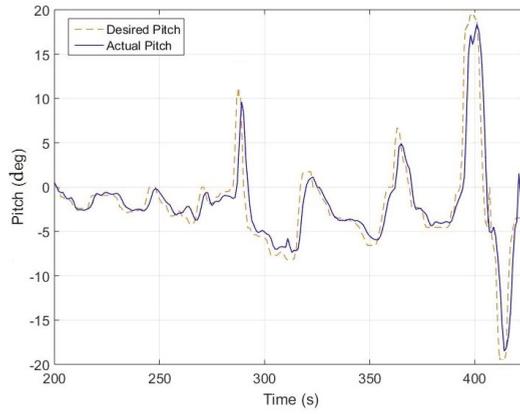


Figure 16. Pitch command tracking of the vehicle.

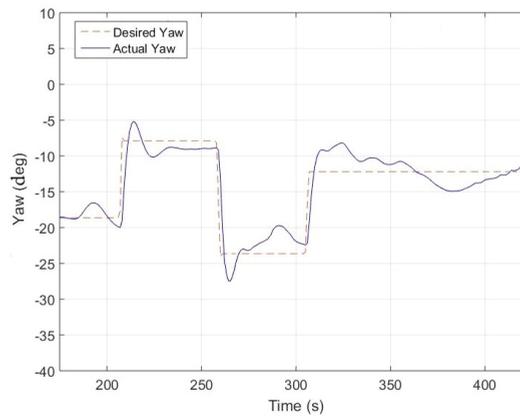


Figure 17. Yaw command tracking of the vehicle.

5.2. Flight Test with Tilting Rotors

The experimental scenario was created in order to validate the performance of the quadcopter using tilting rotors. As mentioned in Section 3, the quadcopter was simulated to imitate a rectangular trajectory which involves sharp cornering for the vehicle. Real corners (imitating trees in a dense forest) were created in an outdoor flying space, the quadcopter was flown to follow the rectangular trajectory along those corners with tilting and without tilting rotors. Figure 18 shows the track followed by the quadcopter using only tilting angles.

The attitude of the quadcopter during the flight can be observed in Figure 19 which shows that the quadcopter is stabilizing and maintaining its horizontal attitude without contributing in the movement in order to follow the trajectory. The rectangular trajectory including the cornering is achieved using only tilting of the rotors along the arms.

The same trajectory following was performed without using tilting of the rotors, with conventional configuration of the quadcopter. The results of trajectory followed by the quadcopter with conventional configuration is shown in Figure 20.

Comparing of the quadcopter attitude in both cases, i.e. with conventional actuation and with tilting rotor actuation for the same trajectory (Figures 19 and 21), it can be observed that the quadcopter with conventional actuation requires the whole frame to be tilted in order to maintain attitude. Whereas dual tilting actuation quadcopter provides the ability to maneuver in a way regardless of its attitude.

Furthermore, it is evident from the comparison of the trajectory followed by the quadcopter with two different actuation strategies that the quadcopter with tilting actuation of the rotors is able to maneuver efficiently through corners which minimizes the effort of the vehicle movement.

From Figures 18 and 20, it can be noticed that the conventional actuation of the quadcopter limits the motion of the vehicle around sharp corners up to certain extent, requiring a larger turning radius in order to do the cornering. The clearing distance through obstacles using tilting is $d = 1.25$ m, while with the conventional configuration, $d = 2.65$ m. It is clear that wider gap is required for the conventional quadcopter to fly through the obstacles whereas with tilting ability, the quadcopter is able to fly more precisely, reducing the clearing distance. The conventional quadcopter would not be able to execute such maneuvers as the under-actuation limits its ability. Tilting provides more controllability and ability to the vehicle as it can move without changing its attitude which helps the developed system to fly through narrow gaps and under trees canopy. The combination of tilting rotors with attitude control provide increased agility and control bandwidth in an urban outdoor scenario where the flying through narrow gaps and obstacles is challenging as compared to under-actuated quadcopter.

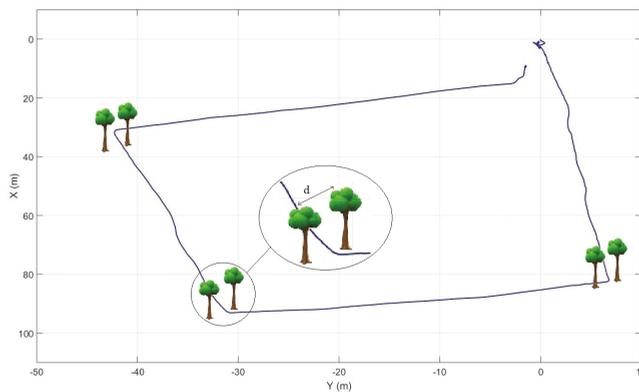


Figure 18. The real-time trajectory followed by UAV using tilting, $d = 1.25$ m.

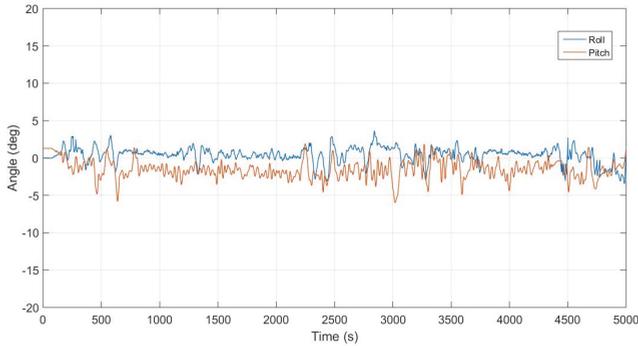


Figure 19. Roll and pitch angle of the UAV while maneuvering only with tilting.

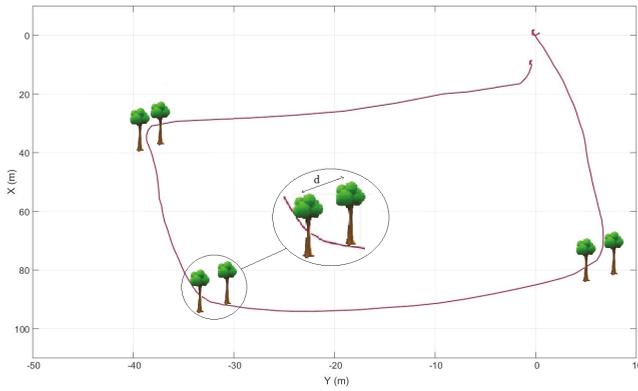


Figure 20. The real-time trajectory followed by UAV without using tilting, $d = 2.65$ m.

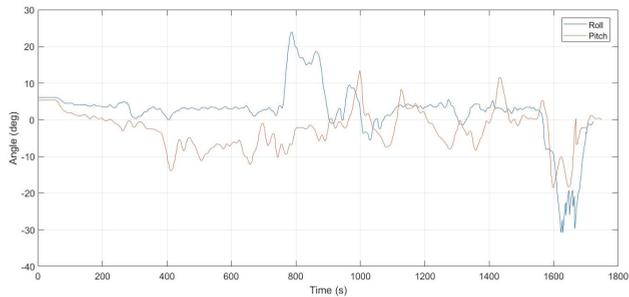


Figure 21. Roll and Pitch angle of the UAV while maneuvering without tilting.

6. Conclusions

This paper proposes a design and implementation of an over-actuated quadcopter with dual-axis tilting rotors. The CAD model was developed following the manufacturing of the system using rapid prototyping in order to minimize the manufacturing time and cost. The modeling and simulation of the over-actuated system allowed observing the behavior of the platform using different control inputs. The flight test results in outdoor conditions show satisfactory performance of the developed platform.

The experiments were performed to observe and compare the capability of over-actuated configuration against conventional configuration which showed that the proposed platform performs better when it comes to flying along corners and between adjacent obstacles, and gives better maneuverability compared to conventional quadcopter.

Integration of dual-axis tilting capability to quadcopters opens different research areas, including developing control and recovery strategies. Quadcopter with dual-tilting actuation will vastly expand its applications in search and rescue missions, detecting missing persons in a wide dense forest area with advantage of being able to fly under trees canopy.

Future work will focus on quantifying the energy consumption and the development of different fail-safe strategies in case of failure of one or two of the rotors. The development of failure strategies will use the full capability of over-actuation presented in the current research.

Author Contributions: All authors have made great contributions to the work. A.B.J., J.B.B. and Y.Z. designed and built the platform. A.D.D.C.S., J.B.B. and Y.Z. developed the model and the simulation. A.B.J., A.D.D.C.S. and N.V. conducted the experiments. A.B.J., N.V. and Y.Z. analyzed the data and revised the manuscript.

Funding: This research received no external funding.

Acknowledgments: The authors would like to thank Miguel Guzman and Zineb Chelh for their help in the preliminary platform design and simulation.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

θ	Pitch angle
φ	Roll angle
ψ	Yaw angle
ω_i	Angular velocity
L	Arm length
dIB_x	Time derivative of inertia along X axis
dIB_y	Time derivative of inertia along Y axis
dIB_z	Time derivative of inertia along Z axis
IB_{xx}	Body's inertia along X axis
IB_{yy}	Body's inertia along Y axis
IB_{zz}	Body's inertia along Z axis
IP_x	Propeller's inertia along X axis
IP_y	Propeller's inertia along Y axis
IP_z	Propeller's inertia along Z axis
i	number of rotor (1, 2, 3, 4)
$P_{N,E,D}$	Position in North, East and Down axis
P	Angular velocity in X axis
Q	Angular velocity in Y axis
R	Angular velocity in Z axis
n_i or η_i	Rotor tilting angle along the arm
b_i or β_i	Rotor tilting angle across the arm
x_W, y_W, z_W	Fixed frame (1)
x_B, y_B, z_B	Body frame (2)
x_P, y_P, z_P	Rotor frame (3)
R_x, R_y, R_z	Rotation matrices in x, y and z axis
α	Angular acceleration
u, v, w	Linear velocity in x, y and z axis
T_i	Thrust generated by the rotor i

Appendix A. Coefficients of the Forces and the Torques

The coefficients of the force dynamic equation in x -axis are as follows:

$$k_{xi} = \sin(\phi) \sin(\theta) [\cos(b_i) [\sin(n_i) \cos(\psi) - \cos(n_i)] - \sin(b_i) \sin(\psi)] \quad (A1)$$

$$k_{xi}^* = \sin(\phi) \sin(\theta) [\cos(b_i) [\sin(n_i) \cos(\psi) + \cos(n_i)] + \sin(b_i) \sin(\psi)] \quad (A2)$$

$$\lambda_{xi} = \cos(\theta) [\sin(b_i) \cos(\psi) + \sin(n_i) \cos(b_i) + \sin(\psi)] \quad (A3)$$

$$\lambda_{xi}^* = \cos(\theta) [\sin(b_i) \cos(\psi) - \sin(n_i) \cos(b_i) + \sin(\psi)] \quad (A4)$$

The coefficients of the force dynamic equation in y -axis are as follows:

$$k_{yi} = \sin(\phi) \sin(\theta) [\cos(b_i) [\sin(n_i) \cos(\psi) + \cos(n_i)] - \sin(b_i) \sin(\psi)] \quad (A5)$$

$$k_{yi}^* = \sin(\phi) \sin(\theta) [\cos(b_i) [\sin(n_i) \cos(\psi) - \cos(n_i)] + \sin(b_i) \sin(\psi)] \quad (A6)$$

$$\lambda_{yi} = \cos(\phi) [\sin(b_i) \cos(\psi) + \sin(n_i) \cos(b_i) \sin(\psi)] \quad (A7)$$

$$\lambda_{yi} = \cos(\phi) [\sin(b_i) \cos(\psi) + \sin(n_i) \cos(b_i) \sin(\psi)] \quad (A8)$$

The coefficients of the force dynamic equation in z -axis are as follows:

$$k_{z1} = \sin(\theta) [\sin(\psi) [\sin(n_1) \cos(b_1) - \sin(b_1) \cos(\phi)] - \sin(n_1) \cos(b_1) \cos(\phi) \cos(\psi)] \quad (A9)$$

$$k_{z2} = \sin(\theta) [\sin(\psi) [\sin(b_2) + \sin(n_2) \cos(b_2) \cos(\phi)] - \sin(b_2) \cos(\phi) \cos(\psi)] \quad (A10)$$

$$k_{z3} = -\sin(\theta) [\sin(\psi) [\sin(n_3) \cos(b_3) + \sin(b_3) \cos(\phi)] - \sin(n_3) \cos(b_3) \cos(\phi) \cos(\psi)] \quad (A11)$$

$$k_{z4} = \sin(\theta) [\sin(\psi) [\sin(b_4) + \sin(n_4) \cos(b_4) \cos(\phi)] - \sin(b_4) \cos(\phi) \cos(\psi)] \quad (A12)$$

$$\lambda_{z1} = \cos(n_1) \cos(b_1) \cos(\theta) \cos(\phi) - \sin(b_1) \sin(\phi) \cos(\psi) \quad (A13)$$

$$\lambda_{z2} = \cos(b_2) [\sin(n_2) \sin(\phi) \cos(\psi) - \cos(n_2) \cos(\phi) \cos(\theta)] \quad (A14)$$

$$\lambda_{z3} = -\cos(n_3) \cos(b_3) \cos(\theta) \cos(\phi) + \sin(b_3) \sin(\phi) \cos(\psi) \quad (A15)$$

$$\lambda_{z4} = -\cos(b_4) [\sin(n_4) \sin(\phi) \cos(\psi) - \cos(n_4) \cos(\phi) \cos(\theta)] \quad (A16)$$

The coefficients of the torque dynamic equation in x -axis are as follows:

$$l_{xx} = -\cos(n_1) IP_x \ddot{b}_1 - \sin(n_2) \sin(b_2) IP_x \ddot{b}_2 + \cos(n_3) IP_x \ddot{b}_3 + \sin(n_4) \sin(b_4) IP_x \ddot{b}_4 \quad (A17)$$

$$l_{xy} = -\sin(n_1) \sin(b_1) IP_y \ddot{n}_1 + \cos(b_2) IP_y \ddot{n}_2 + \sin(n_3) \sin(b_3) IP_y \ddot{n}_3 - \cos(b_4) IP_y \ddot{n}_4 \quad (A18)$$

$$l_{xz} = -\sin(n_1) \cos(b_1) IP_z \omega_1 - \sin(b_2) IP_z \omega_2 + \sin(n_3) \cos(b_3) IP_z \omega_3 + \sin(b_4) IP_z \omega_4 \quad (A19)$$

$$\rho_x = QRIB_{yy} - QRIB_{zz} - PdIB_x \quad (A20)$$

The coefficients of the torque dynamic equation in y -axis are as follows:

$$l_{yx} = \sin(n_1) \sin(b_1) IP_x \ddot{b}_1 - \sin(n_2) IP_x \ddot{b}_2 - \sin(n_3) \sin(b_3) IP_x \ddot{b}_3 + \sin(n_4) IP_x \ddot{b}_4 \quad (A21)$$

$$I_{yy} = -\cos(b_1)IP_y\ddot{n}_1 - \sin(n_2)\sin(b_2)IP_y\ddot{n}_2 + \cos(b_3)IP_y\ddot{n}_3 + \sin(n_4)\sin(b_4)IP_y\ddot{n}_4 \quad (A22)$$

$$I_{yz} = \sin(b_1)IP_z\omega_1 - \sin(n_2)\cos(b_2)IP_z\omega_2 - \sin(b_3)IP_z\omega_3 + \sin(n_4)\cos(b_4)IP_z\omega_4 \quad (A23)$$

$$\rho_y = PRIB_{zz} - PRIB_{xx} - QdIB_y \quad (A24)$$

The coefficients of the torque dynamic equation in z-axis are as follows:

$$I_{zx} = \sin(n_1)IP_x\ddot{b}_1 + \sin(n_2)IP_x\ddot{b}_2 + \sin(n_3)IP_x\ddot{b}_3 + \sin(n_4)IP_x\ddot{b}_4 \quad (A25)$$

$$I_{zy} = -\cos(n_1)\sin(b_1)IP_y\ddot{n}_1 - \cos(n_2)\sin(b_2)IP_y\ddot{n}_2 - \cos(n_3)\sin(b_3)IP_y\ddot{n}_3 - \cos(n_4)\sin(b_4)IP_y\ddot{n}_4 \quad (A26)$$

$$I_{zz} = \cos(n_1)\cos(b_1)IP_z\omega_1 + \cos(n_2)\cos(b_2)IP_z\omega_2 + \cos(n_3)\cos(b_3)IP_z\omega_3 + \cos(n_4)\cos(b_4)IP_z\omega_4 \quad (A27)$$

$$\rho_z = QPIB_{xx} - PQIB_{yy} - RdIB_z \quad (A28)$$

Appendix B. Inertia Values

Expanding Equation (7) yields the inertia values in the main axis as follows:

$$I_{Bxx} = I_{xx} + 2IP_{xx} + 2IP_{zz} + \lambda [\cos(2\eta_1) + \cos(2\eta_2) + \cos(2\eta_3) + \cos(2\eta_4)] + 2m_pL^2 \quad (A29)$$

$$I_{Byy} = I_{yy} + 2IP_{yy} + IP_{xx} + IP_{zz} - \frac{1}{2}\lambda [\cos(2\eta_1) + \cos(2\eta_2) + \cos(2\eta_3) + \cos(2\eta_4)] + \zeta [\cos(2\beta_1) + \cos(2\beta_2) + \cos(2\beta_3) + \cos(2\beta_4)] + \frac{1}{2}\lambda [\cos(2\eta_1)\cos(2\beta_1) + \cos(2\eta_2)\cos(2\beta_2) + \cos(2\eta_3)\cos(2\beta_3) + \cos(2\eta_4)\cos(2\beta_4)] + 2m_pL^2 \quad (A30)$$

$$I_{Bzz} = I_{zz} + 2IP_{yy} + IP_{xx} + IP_{zz} - \frac{1}{2}\lambda [\cos(2\eta_1) + \cos(2\eta_2) + \cos(2\eta_3) + \cos(2\eta_4)] - \zeta [\cos(2\beta_1) + \cos(2\beta_2) + \cos(2\beta_3) + \cos(2\beta_4)] - \frac{1}{2}\lambda [\cos(2\eta_1)\cos(2\beta_1) + \cos(2\eta_2)\cos(2\beta_2) + \cos(2\eta_3)\cos(2\beta_3) + \cos(2\eta_4)\cos(2\beta_4)] + 4m_pL^2 \quad (A31)$$

where the value of ζ and λ are defined as follows

$$\lambda = \frac{1}{2}(IP_{xx} - IP_{zz}) \quad (A32)$$

$$\zeta = \frac{1}{2}IP_{yy} - \frac{1}{4}IP_{xx} - \frac{1}{4}IP_{zz}$$

The last expression to be derived is the time derivative of the inertia matrix which is given by Equations (A29)–(A31). For each principal axis of the inertia, it yields:

$$\frac{d}{dt}(I_{Bxx}) = -2\lambda \left(\sin(2\eta_1) \left(\frac{d}{dt}(\eta_1) \right) \right) - 2\lambda \left(\sin(2\eta_2) \left(\frac{d}{dt}(\eta_2) \right) \right) - 2\lambda \left(\sin(2\eta_3) \left(\frac{d}{dt}(\eta_3) \right) \right) - 2\lambda \left(\sin(2\eta_4) \left(\frac{d}{dt}(\eta_4) \right) \right) \quad (A33)$$

$$\begin{aligned}
 \frac{d}{dt} (I_B y y) = & \lambda \sin(2\eta_1) \frac{d}{dt}(\eta_1) + \lambda \sin(2\eta_2) \frac{d}{dt}(\eta_2) + \lambda \sin(2\eta_3) \frac{d}{dt}(\eta_3) \\
 & + \lambda \sin(2\eta_4) \frac{d}{dt}(\eta_4) - \lambda \sin(2\eta_1) \cos(2\beta_1) \frac{d}{dt}(\eta_1) - \lambda \sin(2\eta_2) \cos(2\beta_2) \frac{d}{dt}(\eta_2) \\
 & - \lambda \sin(2\eta_3) \cos(2\beta_3) \frac{d}{dt}(\eta_3) - \lambda \sin(2\eta_4) \cos(2\beta_4) \frac{d}{dt}(\eta_4) \\
 & - 2 \left(\zeta + \frac{1}{2} \lambda \cos(2\eta_1) \right) \sin(2\beta_1) \frac{d}{dt}(\beta_1) - 2 \left(\zeta + \frac{1}{2} \lambda \cos(2\eta_2) \right) \sin(2\beta_2) \frac{d}{dt}(\beta_2) \\
 & - 2 \left(\zeta + \frac{1}{2} \lambda \cos(2\eta_3) \right) \sin(2\beta_3) \frac{d}{dt}(\beta_3) - 2 \left(\zeta + \frac{1}{2} \lambda \cos(2\eta_4) \right) \sin(2\beta_4) \frac{d}{dt}(\beta_4)
 \end{aligned} \tag{A34}$$

$$\begin{aligned}
 \frac{d}{dt} (I_B z z) = & \lambda \sin(2\eta_1) \frac{d}{dt}(\eta_1) + \lambda \sin(2\eta_2) \frac{d}{dt}(\eta_2) + \lambda \sin(2\eta_3) \frac{d}{dt}(\eta_3) \\
 & + \lambda \sin(2\eta_4) \frac{d}{dt}(\eta_4) + \lambda \sin(2\eta_1) \cos(2\beta_1) \frac{d}{dt}(\eta_1) + \lambda \sin(2\eta_2) \cos(2\beta_2) \frac{d}{dt}(\eta_2) \\
 & + \lambda \sin(2\eta_3) \cos(2\beta_3) \frac{d}{dt}(\eta_3) + \lambda \sin(2\eta_4) \cos(2\beta_4) \frac{d}{dt}(\eta_4) \\
 & + 2 \left(\zeta + \frac{1}{2} \lambda \cos(2\eta_1) \right) \sin(2\beta_1) \frac{d}{dt}(\beta_1) + 2 \left(\zeta + \frac{1}{2} \lambda \cos(2\eta_2) \right) \sin(2\beta_2) \frac{d}{dt}(\beta_2) \\
 & + 2 \left(\zeta + \frac{1}{2} \lambda \cos(2\eta_3) \right) \sin(2\beta_3) \frac{d}{dt}(\beta_3) + 2 \left(\zeta + \frac{1}{2} \lambda \cos(2\eta_4) \right) \sin(2\beta_4) \frac{d}{dt}(\beta_4)
 \end{aligned} \tag{A35}$$

References

1. Mahony, R.; Kumar, V. Aerial Robotics and the Quadrotor. *IEEE Robot. Autom. Mag.* **2012**, *19*, 19. [\[CrossRef\]](#)
2. Doherty, P.; Rudol, P. A UAV Search and Rescue Scenario with Human Body Detection and Geolocalization. In Proceedings of the Australian Conference on Artificial Intelligence, Gold Coast, Australia, 2–6 December 2007; pp. 1–13. [\[CrossRef\]](#)
3. Mahony, R.; Kumar, V.; Corke, P. Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor. *IEEE Robot. Autom. Mag.* **2012**, *19*, 20–32. [\[CrossRef\]](#)
4. Hua, M.D.; Hamel, T.; Morin, P.; Samson, C. A Control Approach for Thrust-Propelled Underactuated Vehicles and its Application to VTOL Drones. *IEEE Trans. Autom. Control* **2009**, *54*, 1837–1853.
5. Hua, M.D.; Hamel, T.; Morin, P.; Samson, C. Introduction to feedback control of underactuated VTOL vehicles: A review of basic control design ideas and principles. *IEEE Control Syst.* **2013**, *33*, 61–75. [\[CrossRef\]](#)
6. Lo, C.H.; Shin, H.S.; Tsourdos, A.; Kim, S. Modeling and Simulation of Fault Tolerant Strategies for A Quad Rotor UAV. In Proceedings of the AIAA Modeling and Simulation Technologies Conference, Minneapolis, MI, USA, 13–16 August 2012.
7. Morozov, Y.V. Emergency Control of a Quadcopter in Case of Failure of Two Symmetric Propellers. *Autom. Remote Control* **2018**, *79*, 463–478. [\[CrossRef\]](#)
8. Adir, V.G.; Stoica, A.M.; Marks, A.; Whidborne, J.F. stabilization and single motor failure recovery of a 4Y octorotor. In Proceedings of the IASTED International Conference on Intelligent Systems and Control (ISC 2011), Cambridge, UK, 11–13 July 2011; pp. 82–87.
9. Michieletto, G.; Ryll, M.; Franchi, A. Control of statically hoverable multi-rotor aerial vehicles and application to rotor-failure robustness for hexarotors. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 2747–2752. [\[CrossRef\]](#)
10. Cutler, M.; Ure, N.K.; Michini, B.; How, J. Comparison of Fixed and Variable Pitch Actuators for Agile Quadrotors. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, Portland, OR, USA, 8–11 August 2011.
11. Oner, K.T.; Cetinsoy, E.; Unel, M.; Aksit, M.F.; Kandemir, I.; Gulez, K. Dynamic model and control of a new quadrotor unmanned aerial vehicle with tilt-wing mechanism. *World Acad. Sci. Eng. Technol.* **2008**, *45*, 58–63.
12. Jiang, G.; Voyles, R. Hexrotor UAV platform enabling dextrous interaction with structures-flight test. In Proceedings of the 2013 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Linkoping, Sweden, 21–26 October 2013; pp. 1–6. [\[CrossRef\]](#)
13. Gasco, P.S. Development of a Dual Axis Tilt Rotorcraft. Master's Thesis, Cranfield University, Bedford, UK, 2012.
14. Gasco, P.S.; Al-Rihani, Y.; Shin, H.S.; Savvaris, A. A novel actuation concept for a multi rotor UAV. In Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, 28–31 May 2013; pp. 373–382. [\[CrossRef\]](#)
15. Nemati, A.; Kumar, M. Modeling and control of a single axis tilting quadcopter. In Proceedings of the American Control Conference, Portland, OR, USA, 4–6 June 2014; pp. 3077–3082. [\[CrossRef\]](#)

16. Scholz, G.; Trommer, G.F. Model based control of a quadrotor with tiltable rotors. *Gyrosc. Navig.* **2016**, *7*, 72–81. [CrossRef]
17. Lim, K.; Shin, J.Y.; Moerder, D.; Cooper, E. A New Approach to Attitude Stability and Control for Low Airspeed Vehicles. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, Providence, RI, USA, 16–19 August 2004. [CrossRef]
18. Gress, G. Using Dual Propellers as Gyroscopes for Tilt-Prop Hover Control. In Proceedings of the Biennial International Powered Lift Conference and Exhibit, Williamsburg, VA, USA, 5–7 November 2002.
19. Gress, G. Lift fans as gyroscopes for controlling compact VTOL air vehicles: Overview and development status of Oblique Active Tilting. In Proceedings of the American Helicopter Society 63th Annual Forum, Virginia Beach, VA, USA, 1–3 May 2007.
20. Sanchez, A.; Escareño, J.; Garcia, O.; Lozano, R. Autonomous Hovering of a Noncyclic Tiltrotor UAV: Modeling, Control and Implementation. *IFAC Proce. Vol.* **2008**, *41*, 803–808. [CrossRef]
21. Klippstein, H.; Diaz De Cerio Sanchez, A.; Hassanin, H.; Zweiri, Y.; Seneviratne, L. Fused Deposition Modeling for Unmanned Aerial Vehicles (UAVs): A Review. *Adv. Eng. Mater.* **2017**, 1700552. [CrossRef]
22. Klippstein, H.; Hassanin, H.; Diaz De Cerio Sanchez, A.; Zweiri, Y.; Seneviratne, L. Additive Manufacturing of Porous Structures for Unmanned Aerial Vehicles Applications. *Adv. Eng. Mater.* **2018**, 1800290. [CrossRef]
23. Pragada, L.K.D.; Katukam, R. 3d printing of quadcopter: A Case Study. *Int. J. Latest Trends Eng. Technol.* **2015**, *5*, 131–140.
24. Hooi, C.G. Design, Rapid Prototyping and Testing of a Ducted Fan Microscale Quadcopter. In Proceedings of the American Helicopter Society 70th Annual Forum, Montreal, QC, Canada, 20–22 May 2014.
25. Alzu'bi, H.; Allateef, I.; Zweiri, Y.; Alkhateeb, B.; Al-Masarwah, I. Quad tilt rotor Vertical Take Off and Landing (VTOL) Unmanned Aerial Vehicle (UAV) with 45 degree rotors. Patent US20130105635, 2 May 2013.
26. Ryll, M.; Bühlhoff, H.H.; Giordano, P.R. First flight tests for a quadrotor UAV with tilting propellers. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 295–302. [CrossRef]
27. Ryll, M.; Bühlhoff, H.H.; Giordano, P.R. A Novel Overactuated Quadrotor Unmanned Aerial Vehicle: Modeling, Control, and Experimental Validation. *IEEE Trans. Control Syst. Technol.* **2015**, *23*, 540–556. [CrossRef]
28. Zortrax. Zortrax M200. 2016. Available online: <https://zortrax.com/printers/zortrax-m200/> (accessed on 28 September 2017).
29. Guzman, M. Modeling and Simulation of an Over-Actuated Dual Tilting Rotors Quadcopter. Master's Thesis, Kingston University, London, UK, 2015.
30. Lim, H.; Park, J.; Lee, D.; Kim, H.J. Build Your Own Quadrotor: Open-Source Projects on Unmanned Aerial Vehicles. *IEEE Robot. Autom. Mag.* **2012**, *19*, 33–45. [CrossRef]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Design of a Lockable Spherical Joint for a Reconfigurable 3-URU Parallel Platform

Matteo Palpacelli, Luca Carbonari, Giacomo Palmieri and Massimo Callegari *

Department of Industrial Engineering and Mathematical Sciences, Polytechnic University of Marche, 60131 Ancona, Italy; m.c.palpacelli@univpm.it (M.P.); l.carbonari@univpm.it (L.C.); g.palmieri@univpm.it (G.P.)

* Correspondence: m.callegari@univpm.it

Received: 13 June 2018; Accepted: 30 July 2018; Published: 2 August 2018

Abstract: This article deals with the functional and preliminary design of a reconfigurable joint for robotic applications. Such mechanism is a key element for a class of lower mobility parallel manipulators, allowing a local reconfiguration of the kinematic chain that enables a change in platform's mobility. The mechanism can be integrated in the kinematic structure of a 3-URU manipulator, which shall accordingly gain the ability to change mobility from pure translation to pure rotation. As a matter of fact, special kinematics conditions must be met for the accomplishment of this task. Such peculiar requirements are described and properly exploited for the design of an effective reconfigurable mechanism. A detailed description of the joint operational principle is provided, also showing how to design it when is physically located at the fixed base of the manipulator.

Keywords: parallel robot; reconfigurable joint; flexible robotics; mechanism; kinematics

1. Introduction

Reconfigurable manipulators may represent an answer to the request of flexibility in the nowadays manufacturing industry. A *reconfigurable* or *metamorphic* manipulator is a machine that is able to change its end-effector mobility according to a local change of kinematics. Several methods can be used to modify the kinematic structure of a manipulator; the most common one is the use of lockable joints, i.e., joints with some degrees of freedom (DOF), one of which can be selectively locked in order to reconfigure the resulting mobility. The family of Parallel Kinematics Machines (PKMs) in particular can significantly take profit by the development of reconfigurable joints since a little modification of legs kinematics may lead to substantial modifications of the global mobility. The limited availability of a dextrous workspace is one of the major drawbacks of PKMs, therefore the possibility of locally (or temporarily) modifying the motion capabilities of a particular kinematic structure can trigger a significant interest.

As is well known, many closed loop joints' topologies, typical of PKMs, can present several *assembly* or *working modes*, often characterized by different motion capabilities, corresponding to different branches of the solution robot's kinematics [1–3]. It results that the exploitation of two, three, or even more assembly modes of a parallel kinematic machine could enormously enhance its usability in an industrial application. Some studies focused on the possibility of modifying robot mobility by changing the assembly mode but avoiding the disassembly of the manipulator [4–8]. Further attention has been paid to temporary modification of the kinematics structure, aimed at exploitation of the superior capabilities of under-actuated redundant structures: in such cases a lockable joint properly actuated during motion can improve the ability of performing positioning tasks [9–11]. Moreover, the typical modularity of serial robots can be searched also in reconfigurable PKMs [12,13], where a combination of elemental branches, sometimes provided with lockable joints, can be exploited to assemble manipulators characterized by different motion capabilities. Moreover, a 6-DOF full mobility

parallel robot equipped with six motors can also take advantage of lockable joints to behave like a legged manipulator [14], which can perform six-axis machining operations but also walk and reach other working stations. Scientific literature already provides few examples of lockable joints [15–19]; in particular, multiple DOF's kinematic pairs are obtained by the composition of elementary kinematic pairs which are then locked alternately; however, the change of configuration of the joints is often managed manually, without taking care of the behaviour of the manipulator during the transition. In fact, it may happen that the robot passes through an under-constrained configuration, temporarily gaining degrees of freedom: in this case, only an external manual support allows for holding the robot in its pose.

The researchers at the Robotics Lab of the Polytechnic University of Marche in Ancona (Italy) in recent years developed several studies on reconfigurable robots exploiting the concept of lockable joints. Firstly, they focused on the Cylindrical-Prismatic-Universal three legged architecture (3-CPU): it allows the design of a reconfigurable tripod able to perform pure translational or pure rotational motions, depending on the orientation of the axes of the universal joints [20,21]. The switch between the two mobilities by means of a lockable joint was then faced in [22]. Notwithstanding the discussed metamorphic capabilities, the 3-CPU architecture also owns some intrinsic drawbacks. In particular, the main issue that affects such architecture is the need of reconfiguring the passive joints which connect the end-effector to limbs structure: it implies that the lockable mechanism must be part of leg structure, thus making more complex the mechanical design (for the limitation of weight and size of the device) and leading a reduction of the payload.

Consequently, the authors studied different topologies, which have in common with the 3-CPU manipulator the same types of mobilities [23] among them the 3-URU (Universal-Revolute-Universal) architecture was selected to design a novel reconfigurable manipulator [24,25]. The 3-URU manipulator can be derived from the 3-SRU (Spherical-Revolute-Universal) topology: the spherical joint connecting the leg to the fixed platform can be thought as the composition of three revolute joints with concurrent axes; locking one of them selectively, the joint turns into a universal joint with different sequences of rotations. Two of the possible sequences, as described in the following section, provide the mobile platform with the capability to yield motions of pure rotation and pure translation respectively.

2. The 3-URU Reconfigurable Robot

The main scope of this manuscript is to introduce a novel lockable spherical joint, designed to manage the reconfiguration capabilities of a 3-SRU under-actuated parallel kinematics machine. Such joint is realized as a combination of revolute pairs (see Figure 1); a locking system allows for alternatively locking one of the first two revolute joints, giving the machine different 3-URU kinematic configurations which correspond to different types of mobility. For a deeper understanding of how the joint will affect the robot mobility, a description of its structure is provided in the following.

The kinematic architecture of each leg is composed of a Spherical-Revolute-Universal (SRU) joints chain. The spherical pair, which must be reduced to a universal joint in order to guarantee the functionality of the machine, connects the first body of each leg to the chassis. The second link is connected to the first one by a revolute joint. At last, a universal joint connects the end-effector to the leg. The mutual arrangement of the joints, which is of great importance for the mobility of the moving platform, is shown in Figure 1. As visible, the three spherical pairs are located so that their centers lie on the axes of the fixed reference frame $\{0\}$ at the same distance, called a , from the origin. The three rotations that compose each pair are realized by three perpendicular revolute joints, whose axes concur in a point (namely the center of the spherical pair). At such point, three reference frames have a common origin: such frames, called $\{l_{1,i}\}$, $\{l_{2,i}\}$, $\{l_{3,i}\}$ for the i -th leg, are fixed to the bodies sequentially connected to the three revolute joints. In the home configuration, Figure 1, where the mobile platform frame $\{1\}$ is coincident with the fixed frame $\{0\}$, it is $\{l_{1,i}\} \equiv \{l_{2,i}\} \equiv \{l_{3,i}\}$. Given the description of local frames, the disposition of the three axes composing the spherical joint can be defined as follows:

- the first revolute joint is coaxial to the respective axis of the reference frame $\{0\}$. In the i -th leg local frame $\{l_{1,i}\}$, the unit vector describing such axis is ${}^{l_{1,i}}\mathbf{r}_{1,i} = [1 \ 0 \ 0]^T$;
- the second revolute joint is perpendicular to the first one and it is directed along the first body of the limb at home configuration. In the frame $\{l_{2,i}\}$, it is ${}^{l_{2,i}}\mathbf{r}_{2,i} = [0 \ 0 \ 1]^T$;
- the axis of the third and last rotation is perpendicular to both axes of the first and the second rotations at home configuration. With respect to frame $\{l_{3,i}\}$, it is ${}^{l_{3,i}}\mathbf{r}_{3,i} = [0 \ 1 \ 0]^T$.

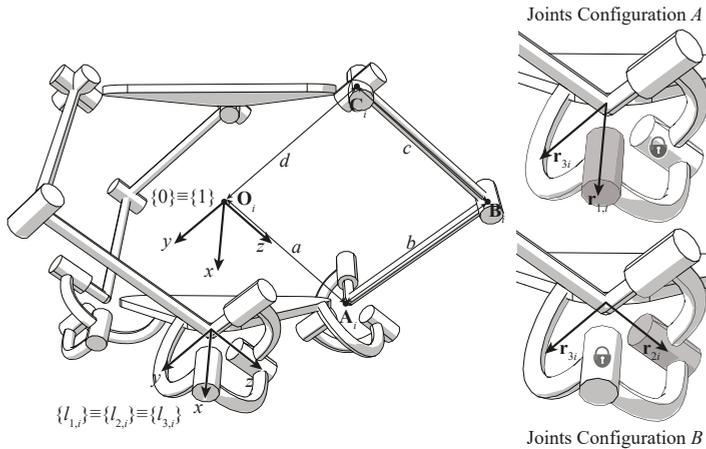


Figure 1. Setting of joints axes of the reconfigurable 3-URU kinematic architecture.

The locking mechanism is used in this case to produce two different configurations, here called Joints Configuration A and Joints Configuration B (see Figure 1). In configuration A, the first rotation of the spherical pair is allowed and, in particular, it represents the actuated degree of freedom of the i -th leg. The second rotation is locked, while the third one is free. In configuration B, the first rotation is locked, the second one is actuated and the third one is free. As demonstrated in [24,25], such joints setting allows the generation of motions of pure rotation with Joint Configuration A and motions of pure translation with Joint Configuration B without further modification to the arrangement of the passive joints.

The rest of the kinematic chain of each leg is composed by a revolute joint, parallel to the last revolute pair of the reconfigurable spherical joint at a distance b , and by a universal joint that connects the second link of the leg to the end-effector. As demonstrated by Palpacelli et al. in [23], the arrangement of the last joint of the chain is also crucial for the definition of end-effector’s mobility. In particular, for the reconfigurable 3-URU parallel robot, the last universal joint is arranged so that the first rotation is parallel to previous revolute joint at a distance c , thus it is perpendicular to the leg plane π_i (identified by points A_i , B_i , and C_i). The second rotation lies on π_i and is perpendicular to the previous one; on the end effector, such axes are mutually perpendicular and concurrent at the origin of the moving frame $\{1\}$. Moreover, such axes coincide with the axes of $\{1\}$. With respect to such reference frame, the three attachment points C_i are identified by vectors ${}^{\{1\}}\mathbf{C}_1 = [0 \ 0 \ -d]^T$, ${}^{\{1\}}\mathbf{C}_2 = [-d \ 0 \ 0]^T$, and ${}^{\{1\}}\mathbf{C}_3 = [0 \ -d \ 0]^T$.

The locking mechanism is designed so as to allow alternatively the motion of the first and the second revolute joint. A similar solution was already proposed by authors in [25] for the reconfiguration of a 3-CPU robot. Nonetheless, the mechanical solution proposed in the present

article is significantly different since previous design was used to modify the mobility of a passive universal joint. In the present case, instead, the reconfiguration implies that the actuated joints of the robot must be changed. The mechanical design of the reconfigurable joint is obviously affected by that. On the one hand, the need of connecting an actuator to different joints relatively complicates the problem. On the other hand, the fact that the lockable device is attached at the ground frame and it is not moving together with the end effector relaxes the design parameters in terms of weight and dimension, and consequently in terms of costs.

3. Conceptual Design of the Reconfigurable Joint

The conceptual design is based on a bevel gear coupling, as shown in Figure 2. Such solution is similar to the one proposed by the authors in [23]. However, the present locking mechanism has been re-conceived in order to allow the actuation of the bevel gear *D*, which must be connected to the motor. The motion of the sliding cursor *C* reconfigures the spherical joint alternatively locking one of the rotations of the spherical joint as described in the following paragraphs.

The cursor *C* is driven by an actuator in two different positions, which provide the joint as many different working modes. The cursor has a cylindrical shape with an external splined shaft and an internal splined hub at the top. In the Configuration *A* (Figure 2, top), the cursor *C* engages the splined hub of the fork *B*. In this case *B*, *C* and *D* globally behave like a rigid body. Due to that, the second fork *E* does not rotate with respect to *B*. The body *F* remains free to rotate since it is not constrained anyhow. Such configuration turns to allow rotations about axes *x* and *y* as required by the 3-URU of pure rotation.

If the cursor is moved downwards to Configuration *B* (Figure 2, down), the splined shaft engages with both bodies *A* and *B*. As a consequence, the rotation of *B* remains locked. The rotation of the actuated bevel gear is transmitted instead to the second fork *E*, which now represents the actuated degree of freedom of the leg. Thus, configuration *B* allows an actuated rotation about axis *z* and a free rotation around *y*, exactly as required for the pure translation configuration.

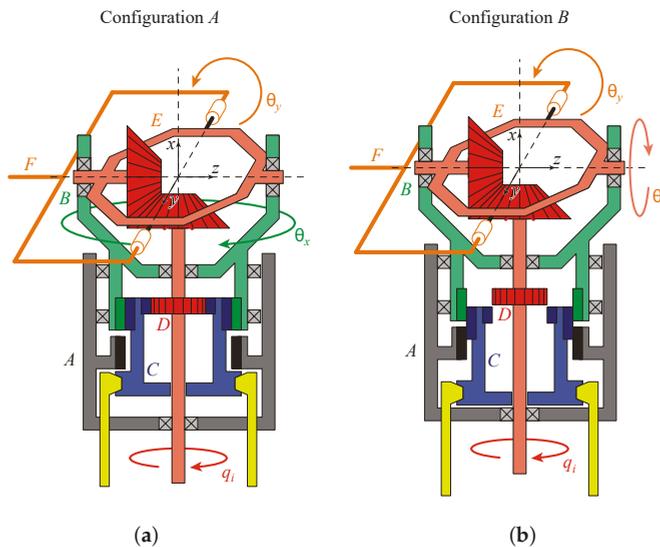


Figure 2. Functional scheme of the reconfigurable joint for spherical (a) and translational (b) motions.

4. Functional Design of the Reconfigurable Joint

Starting from the concept described in the previous section, a functional mechanical design is here proposed and discussed. The exterior aspect of the joint is sketched in Figure 3. Both the actuator of the leg and the actuator of the switching cursor are located externally, fixed to the ground. This represents a crucially important aspect of the present version of the reconfigurable joint, which makes it a significantly enhanced version of the 3-URU reconfigurable device: in [22], the actuator was dedicated to reconfiguring a passive joint within the kinematic chain, with a series of consequent non-avoidable issues. Firstly, the actuator was moved around in the robot workspace, solidly with the last joint of the leg (thus solidly with the robot end-effector). On the other hand, in the 3-URU case study, there was no need to actuate the reconfigured degree of freedom.

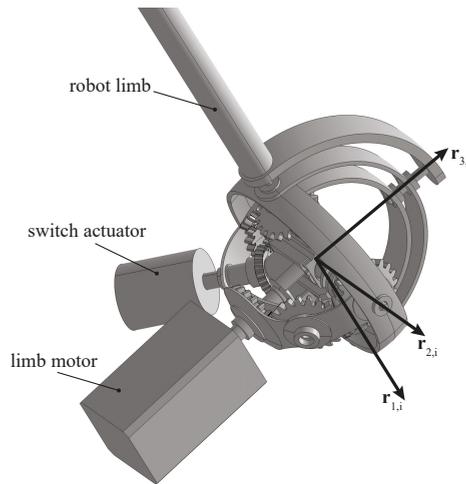


Figure 3. Design of the reconfigurable joint.

Such considerations led to a different functional design of the spherical joint, based on a series of spherical links connected by a system of gears, as described in this section. Globally, the mobility is characterized by a rotation of the limb around $r_{3,i}$ that is always free; such rotation is preceded by a first rotation, whose axis can be selected among $r_{1,i}$ or $r_{2,i}$ acting on the switch.

4.1. The Operational Principle

The joint exploits some bevel gear couplings, as in a previous concept; nevertheless, the rise of new additive manufacturing technologies for fast prototyping of small custom-made parts pushed the authors to adopt different shapes for the design of such gears. In particular, the classical bevel gears previously adopted were extremely binding in terms of dimension and assembly. Thus, a different solution has been adopted with gears obtained from spherical shells in order to maximize the room available within the gears coupling; such space can be used to accommodate the switching cursor. The components of the joint are assembled on different layers, as shown in Figure 4.

The following lines, making reference to Figure 4a, explain how the reconfigurable joint works:

- Joints Configuration A, Figure 4a: the actuator of the limb is fixed to the grounded body 0 (dark grey in figure) and it is connected to the bevel gear 1 which moves gears 2 and 3. In this configuration, the locking mechanism, which is described later, solidly connects the gear 3 to the body 4 (colored in red in Figure 4a). Thus, the rotation of gear 3 moves the robot limb around the axis $r_{1,i}$, as required by Joints Configuration A. The gear 5, which rotates solidly with 3,

does not affect the mobility since it is not connected to the body 4, as well as gears 6, 7, 8, and 9. The rotation around axis $r_{2,i}$ is restricted by the fact that gears 7 and 9 are always solid with the fork 10. Such gears rotate around $r_{1,i}$ by means of the motion of body 4, avoiding them to rotate around their own axes. Rotation about $r_{3,i}$ is no way controlled, as required by the robot kinematics.

- Joints Configuration B, Figure 4b: in this configuration, the locking mechanism connects the body 4 to the ground 0, avoiding the rotation around $r_{1,i}$. The actuation starts again from gear 1 and it is transmitted to 2, 3 and then to 8 and 9. In this case, the rotation of gear 9 puts in rotation the fork 10 around $r_{2,i}$, as prescribed by the kinematics required by joints configuration B. Gears 5, 6 and 7 work exactly in the same way providing a symmetric actuation to the fork 10. Again, rotation about $r_{3,i}$ is no way controlled.

4.2. The Locking Mechanism

The whole functionality of the reconfigurable joint is based on a locking mechanism which is able to alternatively lock body 4 to the bevel gear 3 (Joints Configuration A) or to ground 0 (Joints Configuration B). Such mechanism, which is shown in detail in Figure 5 for both joints configurations A and B, is operated as follows:

- Joints Configuration A, Figure 5a: the actuation of the locking mechanism comes through the bevel gear *a*, which moves the gear *b*, solidly connected to a screw. Such screw puts into translation a nut connected to the fork *c* which pushes the cursor *d*. The aim of such body is that of constraining the splined shaft *e* together with the bevel gear *f*, whose function has already been explained previously (see gear 3 in Figure 4a). It should be noted that the splined shaft *e* is a part of the body indicated with the number 4 in Figure 4a.
- Joints Configuration B, Figure 5b: in this case, the cursor constrains the shaft *e* together with the element *g* which is solid with the ground. In such way, body 4 of Figure 5b is connected with the ground, as required by the kinematics of the robot.

It is worth mentioning that the locking mechanism shall be moved by a dedicated actuator which must be as simple as possible. The particular task it shall fulfill, in fact, is just to rotate the gear *a* through two given positions, for it is not necessary to assume intermediate poses. As a consequence, a simple rotational pneumatic, hydraulic, or solenoid binary actuator will accomplish the purpose without considerably affecting the cost of the whole robot.

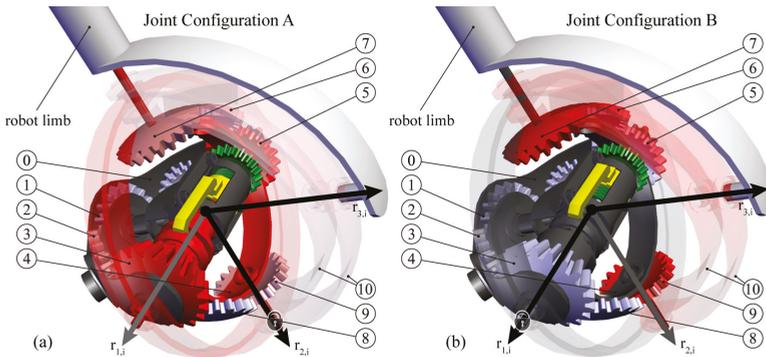


Figure 4. Details of the reconfigurable joint arranged for Joints Configuration A (a), and for Joints Configuration B (b).

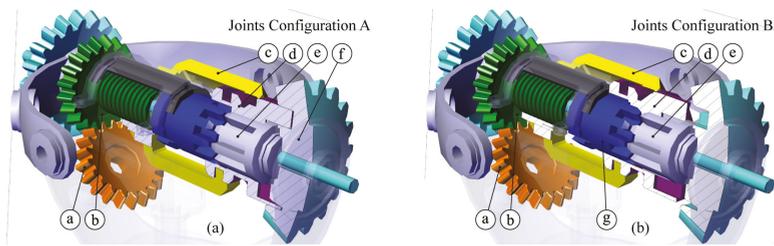


Figure 5. Nut-screw switching mechanism arranged for Joints Configuration A (a), and for Joints Configuration B (b).

5. Conclusions

The article has described the functional design of a reconfigurable universal joint. A switching cursor, which can be driven electronically, is used to change the joint configuration so that two different universal joints can be obtained. Such feature can be exploited to realize a reconfigurable 3-URU manipulator, in which the three universal joints at the fixed base are reconfigurable. A gear system is proposed as a mechanical solution to allow an effective and reliable change of the joint configuration. Future works will be focused on the structural design of the joint, in order to properly design all components in terms of dimensions and materials; such study will start from the static and dynamic analysis of the whole manipulator, in order to assess the forces acting on the joint. In that phase, the joint parts and gears will be also manufactured by rapid prototyping, with the aim of testing their strength and accuracy when assembled in a whole device.

Author Contributions: Conceptualization, L.C. and M.P.; Writing—Original Draft Preparation, G.P.; Supervision, M.C.; Project Administration, M.P.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chablat, D.; Wenger, P. Working modes and aspects in fully parallel manipulators. In Proceedings of the IEEE 1998 International Conference on Robotics and Automation, Leuven, Belgium, 20–20 May 1998; Volume 3, pp. 1964–1969.
2. Chablat, D.; Kong, X.; Zhang, C. Kinematics, workspace and singularity analysis of a multi-mode parallel robot. In Proceedings of the ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Cleveland, OH, USA, 6–9 August 2017; Volume 5A.
3. Au, W.; Chung, H.; Chen, C. Path planning and assembly mode-changes of 6-DOF Stewart-Gough-type parallel manipulators. *Mech. Mach. Theory* **2016**, *106*, 30–49. [[CrossRef](#)]
4. Tian, C.; Fang, Y.; Guo, S.; Qu, H. A class of reconfigurable parallel mechanisms with five-bar metamorphic linkage. *Proc. Inst. Mech. Eng. Part C* **2017**, *231*, 2089–2099. [[CrossRef](#)]
5. Gan, D.; Dai, J.; Dias, J.; Seneviratne, L. Reconfigurability and unified kinematics modeling of a 3rTPS metamorphic parallel mechanism with perpendicular constraint screws. *Robot. Comput. Integr. Manuf.* **2013**, *29*, 121–128. [[CrossRef](#)]
6. Khalid, A.; Mekid, S. Design synthesis of a three legged SPS parallel manipulator. In Proceedings of the 36th International MATADOR Conference, Manchester, UK, 14–16 July 2010; pp. 169–173.
7. Ye, W.; Fang, Y.; Guo, S. Structural synthesis of a class of metamorphic parallel mechanisms with variable mobility. In *Advances in Reconfigurable Mechanisms and Robots I*; Springer: London, UK, 2012; pp. 119–126.
8. Finistauri, A.; Xi, F. Reconfiguration analysis of a fully reconfigurable parallel robot. *J. Mech. Robot.* **2013**, *5*, 041002. [[CrossRef](#)]

9. Grosch, P.; Di Gregorio, R.; López, J.; Thomas, F. Motion planning for a novel reconfigurable parallel manipulator with lockable revolute joints. In Proceedings of the IEEE 2010 International Conference on Robotics and Automation (ICRA), Anchorage, AK, USA, 3–7 May 2010; pp. 4697–4702.
10. Taherifar, A.; Alasty, A.; Salarieh, H.; Boroushaki, M. Path planning for a hyper-redundant manipulator with lockable joints using PSO. In Proceedings of the First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), Tehran, Iran, 13–15 February 2013; pp. 224–229.
11. Aukes, D.; Heyneman, B.; Ulmen, J.; Stuart, H.; Cutkosky, M.; Kim, S.; Garcia, P.; Edsinger, A. Design and testing of a selectively compliant underactuated hand. *Int. J. Robot. Res.* **2014**, *33*, 721–735. [[CrossRef](#)]
12. Xi, F.; Li, Y.; Wang, H. A module-based method for design and analysis of reconfigurable parallel robots. In Proceedings of the International Conference on Mechatronics and Automation (ICMA), Xi'an, China, 4–7 August 2010; pp. 627–632.
13. Zhang, T.; Zhang, W.; Gupta, M. An underactuated self-reconfigurable robot and the reconfiguration evolution. *Mech. Mach. Theory* **2018**, *124*, 248–258. [[CrossRef](#)]
14. Yang, H.; Baradat, C.; Krut, S.; Pierrot, F. An agile manufacturing system for large workspace applications. *Int. J. Adv. Manuf. Technol.* **2016**, *85*, 25–35. [[CrossRef](#)]
15. Zhang, W.J. A new class of linkage with synthesis method for exact dwell motion at 1 and/or 2 limit positions of output link. In Proceedings of the 24th ASME Mechanisms Conference, Irvine, CA, USA, 18–22 August 1996; pp. 1–5.
16. Chen, W.; Zhang, J.; Quan, J.; Lv, T. A novel spherical joint designed for metamorphic mechanism. In Proceedings of the 2008 IEEE Conference on Robotics, Automation and Mechatronics, Chengdu, China, 21–24 September 2008; pp. 976–981.
17. Gan, D.; Dai, J.; Dias, J.; Seneviratne, L. Reconfiguration and static joint force variation of a 3rPS metamorphic parallel mechanism with 3R and 1T2R motion. *Mech. Mach. Sci.* **2016**, *36*, 213–222.
18. Zhang, T.; Zhang, W.J.; Gupta, M.M. A Novel Docking System for Modular Self-Reconfigurable Robots. *Robotics* **2017**, *6*, 25. [[CrossRef](#)]
19. Yuan, C.W.; Yin, R.X.; Zhang W.J.; Chen, G. A new under-actuated resilient robot. In Proceedings of the 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 5–8 October 2017; pp. 1202–1207.
20. Carbonari, L.; Callegari, M. The kinematotropic 3-CPU parallel robot: Analysis of mobility and reconfigurability aspects. In *Latest Advances in Robot Kinematics*; Springer: Berlin, Germany, 2012; pp. 373–380.
21. Carbonari, L.; Callegari, M.; Palmieri, G.; Palpacelli, M.C. A new class of reconfigurable parallel kinematic machines. *Mech. Mach. Theory* **2014**, *79*, 173–183. [[CrossRef](#)]
22. Palpacelli, M.C.; Carbonari, L.; Palmieri, G. Details on the design of a lockable spherical joint for robotic applications. *J. Intell. Robot. Syst.* **2016**, *81*, 169–179. [[CrossRef](#)]
23. Palpacelli, M.; Carbonari, L.; Palmieri, G.; Callegari, M. Mobility analysis of non-overconstrained reconfigurable parallel manipulators with 3-CPU/3-CRU kinematics. In *Advances in Reconfigurable Mechanisms and Robots II*; Springer: Berlin, Germany, 2016; pp. 189–200.
24. Carbonari, L.; Corinaldi, D.; Palpacelli, M.C.; Palmieri, G.; Callegari, M. Functional Design and Optimization of a Novel 3-URU Multimodal Reconfigurable Robot. In Proceedings of the ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Cleveland, OH, USA, 6–9 August 2017; p. V009T07A050.
25. Carbonari, L.; Corinaldi, D.; Palpacelli, M.; Palmieri, G.; Callegari, M. A novel reconfigurable 3-URU parallel platform. In Proceedings of the International Conference on Robotics in Alpe-Adria Danube Region, Torino, Italy, 21–23 June 2017; pp. 63–73.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Performance-Based Design of the CRS-RRC Schoenflies-Motion Generator

Raffaele Di Gregorio ^{1,*}, Mattia Cattai ¹ and Henrique Simas ²

¹ Department of Engineering, University of Ferrara, 44100 Ferrara, Italy; mattia.cattai@gmail.com

² Raul Guenther laboratory of Applied Robotics, Department of Mechanical Engineering, Federal University of Santa Catarina, Florianópolis, SC 88040-900, Brazil; henrique.simas@ufsc.br

* Correspondence: raffaele.digregorio@unife.it; Tel.: +39-0532974828

Received: 22 July 2018; Accepted: 10 September 2018; Published: 15 September 2018

Abstract: Rigid-body displacements obtained by combining spatial translations and rotations around axes whose direction is fixed in the space are named Schoenflies' motions. They constitute a 4-dimensional (4-D) subgroup, named Schoenflies' subgroup, of the 6-D displacement group. Since the set of rotation-axis' directions is a bi-dimensional space, the set of Schoenflies' subgroups is a bi-dimensional space, too. Many industrial manipulations (e.g., pick-and-place on a conveyor belt) require displacements that belong to only one Schoenflies' subgroup and can be accomplished by particular 4-degrees-of-freedom (4-DOF) manipulators (Schoenflies-motion generators (SMGs)). The first author has recently proposed a novel parallel SMG of type CRS-RRC ¹. Such SMG features a single-loop architecture with actuators on the base and a simple decoupled kinematics. Here, firstly, an organic review of the previous results on this SMG is presented; then, its design is addressed by considering its kinetostatic performances. The adopted design procedure optimizes two objective functions, one (global conditioning index (GCI)) that measures the global performance and the other (CI_{\min}) that evaluates the worst local performance in the useful workspace. The results of this optimization procedure are the geometric parameters' values that make the studied SMG have performances comparable with those of commercial SMGs. In addition, a realistic 3D model that solves all the manufacturing doubts with simple and cheap solutions is presented.

Keywords: parallel robot; Schoenflies-motion generator; dimensional synthesis; kinetostatic performances; conditioning index

1. Introduction

Schoenflies' motions are rigid-body displacements obtained by combining spatial translations and rotations around axes with a fixed direction. They constitute a 4-dimensional (4-D) subgroup, named Schoenflies' subgroup, of the 6-D displacement group [1,2]. Since the set of rotation-axis' directions is a bi-dimensional space, the set of Schoenflies' subgroups is a bi-dimensional space, too. Many industrial manipulations (e.g., pick-and-place on a conveyor belt) require displacements that belong to only one Schoenflies' subgroup and can be accomplished by particular 4-degrees-of-freedom (4-DOF) manipulators, named Schoenflies-motion generators (SMGs) [3].

The serial robot SCARA, presented in 1981, is the most known SMG, but many alternative serial architectures can be conceived [3]. The main drawback of serial architectures is the need of actuating

¹ Hereafter, R, P, U, S, and C stand for revolute pair, prismatic pair, universal joint, spherical pair, and cylindrical pair, respectively. With reference to a parallel architecture, which features the frame (base) and the end effector (platform) connected to each other by a number of kinematic chains (limbs), a string of capital letters denotes the sequence of joint types that are encountered by moving from the base to the platform on a limb. The hyphen separates the strings of the limbs and the underlining indicates the actuated joints. A serial architecture has only one limb and is denoted by only one string.

joints that connect mobile links, which brings to increase the mobile masses and, simultaneously, to reduce the dynamic performances. Parallel architectures can solve this issue and parallel SMGs have been presented (see, for instance [4–12]), too. Most of the proposed parallel SMGs feature four limbs (e.g., [5–7,9,12]) and one actuator per limb located on the base. Other parallel SMGs are simply obtained by adding a double Cardan shaft (i.e., a limb of RUPUR type), which connects the base to the platform, in a translational parallel manipulator.

The main drawback of parallel SMGs is their complex multi-loop structure that drastically reduces their workspace, usually brings cumbersome kinematics and control algorithms, and, often, does not allow a full rotation of the end effector (platform). Nevertheless, adopting two-limbed (i.e., single-loop) architectures with serial [4,13] or hybrid [8,11] limbs with two actuators per limb reduces the structure complexity while keeping the actuators on the base. Some design tricks [14,15] can yield end-effector's full rotations, and some architectures [12,13] can give the possibility of decoupling position and orientation (decoupled kinematics), which allows simpler and more intuitive control strategies. Moreover, not-overconstrained architectures [4] make it possible to avoid jamming without using small tolerances during manufacturing.

The first author has recently proposed a novel not-overconstrained parallel SMG of type CRS-RRC [13]. Such SMG features a single-loop architecture with actuators on the base and a simple decoupled kinematics. Here, firstly, an organic review of the previous results on this SMG is presented; then, its design is addressed by considering its kinetostatic performances. This design procedure will yield a realistic 3D model that solves all the manufacturing doubts with simple and cheap solutions.

The paper is organized as follows. Sections 2 and 3 state the notations and review the previous results presented in [13,16]. Section 4 addresses the analysis of the kinetostatic performances and the design based on them. Section 5 discusses the results and Section 6 draws the conclusions.

2. Notations and Background

Figure 1 shows the SMG of type CRS-RRC presented in [13]. The axes of the R and C pairs are all parallel. The platform is connected to the base through two limbs: one of type CRS and the other of type RRC. The actuated C pair of the CRS limb is obtained by mean of a PR chain with the sliding direction of the P pair parallel to the axis of the R pair. Such a kinematic chain can be actuated by keeping the motors on the base and using commercial components as shown in Figure 2. Also, the second actuated R pair of the RRC limb can be moved from the base by simply using a toothed-belt transmission.

With reference to Figure 1, $O_b x_b y_b z_b$ is a Cartesian reference fixed to the base; the direction of the z_b coordinate axis is the same as the R and C pair axes. In the platform, a_p is the constant distance of the center of the S pair from the axis of the passive C pair and is equal to the length of the segment $A_p A_2$. O_p is the reference point of the platform, and h is the length of the segment $A_p O_p$. The coordinates, $(x_p, y_p, z_p)^T$, of O_p , measured in $O_b x_b y_b z_b$, locate the position of the platform; whereas, the angle, φ , between the segment $A_p A_2$ and a line parallel to x_b and passing through A_p uniquely determines the platform orientation. In the CRS limb, point B_2 lies on the axis of the actuated C pair and is fixed to the output link of the C pair. The linear variable of the actuated C pair is the signed distance, d , of B_2 from O_b . The plane parallel to the $x_b y_b$ plane and passing through B_2 intersects the axis of the passive R pair at D_2 , the axis of the passive C pair at A_p , and the axis parallel to the z_b axis and passing through the center of the S pair at A_2 . a_3 and a_4 are the lengths of the segments $B_2 D_2$ and $D_2 A_2$, respectively; whereas, θ_3 and θ_4 are the angular variable of the actuated C pair and the joint variable of the passive R pair, respectively. The point A_p is fixed to the platform. In the RRC limb, the actuated-joint variables are θ_1 and θ_2 . The $x_b y_b$ plane intersects the axis of the R pair adjacent to the base at B_1 , the axis of the other R pair at D_1 , and the axis of the passive C pair at A_1 . a_0 , a_1 and a_2 are the lengths of the segments $O_b B_1$, $B_1 D_1$ and $D_1 A_1$, respectively. The RRC limb constrains the platform to perform Schoenflies motions with rotation axis parallel to the z_b axis; whereas, the CRS limb controls the coordinate z_p of point O_p through its linear variable, d , and, independently, the platform orientation through its angular variable, θ_3 .

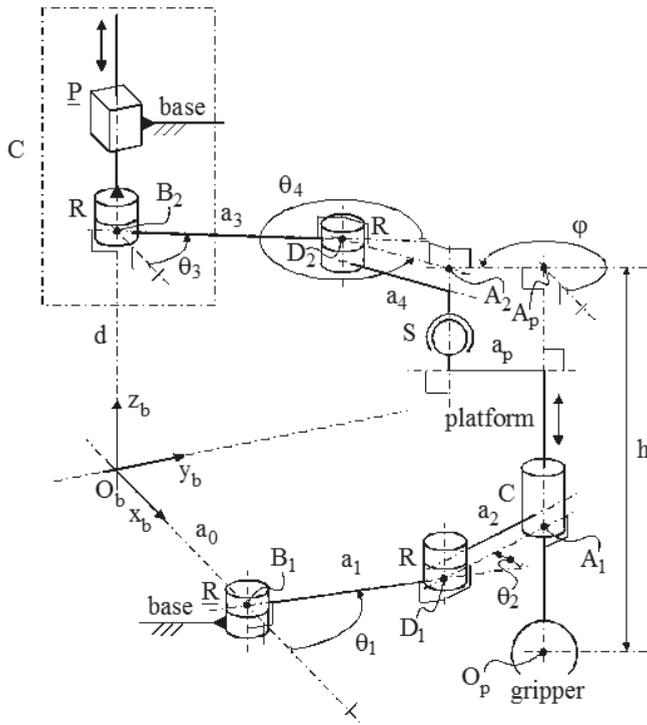


Figure 1. The Shoeflies-motion generators (SMG) of type CRS-RRC.

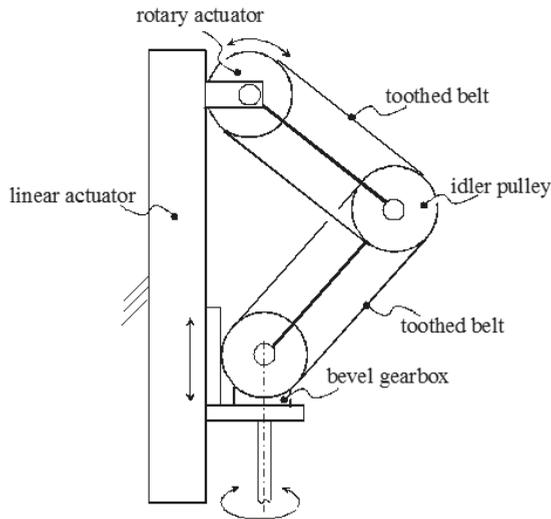


Figure 2. Constructive scheme of an actuated C pair with motors on the frame.

According to these notations, the 4-tuples $\mathbf{q} = (\theta_1, \theta_2, \theta_3, d)^T$ and $\mathbf{\kappa} = (x_p, y_p, z_p, \varphi)^T$ collect the actuated-joint (input) variables and the platform-pose (output) variables, respectively. The inspection of Figure 1 yields the following closure equations

$$x_p = a_0 + a_1 \cos \theta_1 + a_2 \cos (\theta_1 + \theta_2), \tag{1a}$$

$$y_p = a_1 \sin \theta_1 + a_2 \sin (\theta_1 + \theta_2), \tag{1b}$$

$$z_p = d - h \tag{1c}$$

$$(x_p + a_p \cos \varphi - a_3 \cos \theta_3)^2 + (y_p + a_p \sin \varphi - a_3 \sin \theta_3)^2 = a_4^2 \tag{1d}$$

If \mathbf{q} is assigned (direct position analysis (DPA)), Equation (1a–c) yield a unique platform position; whereas, Equation (1d) yields two platform orientations (i.e., the DPA has two solutions that share the same platform position). Vice versa, if $\mathbf{\kappa}$ is assigned (inverse position analysis (IPA)), Equation (1c) yields only one value for d ; whereas, Equation (1a,b,d) yield at most four values for $(\theta_1, \theta_2, \theta_3)$ which can be computed through explicit formulas (see [13] for the proof and the formulas), that is, the IPA has at most four solutions which share the same value of d .

The time derivatives of Equation (1a–d) yield

$$\dot{x}_p = -[a_1 \sin \theta_1 + a_2 \sin(\theta_1 + \theta_2)]\dot{\theta}_1 - a_2 \sin(\theta_1 + \theta_2) \dot{\theta}_2 \tag{2a}$$

$$\dot{y}_p = [a_1 \cos \theta_1 + a_2 \cos(\theta_1 + \theta_2)]\dot{\theta}_1 + a_2 \cos(\theta_1 + \theta_2) \dot{\theta}_2 \tag{2b}$$

$$\dot{z}_p = \dot{d} \tag{2c}$$

$$m_x \dot{x}_p + m_y \dot{y}_p + \dot{\varphi} a_p (m_y \cos \varphi - m_x \sin \varphi) = \dot{\theta}_3 a_3 (m_y \cos \theta_3 - m_x \sin \theta_3) \tag{2d}$$

where $m_x = x_p + a_p \cos \varphi - a_3 \cos \theta_3$ and $m_y = y_p + a_p \sin \varphi - a_3 \sin \theta_3$.

Since the RRC limb forbids the platform to perform displacements that do not belong to the above-mentioned Schoenflies' subgroup, the studied SMG has no constraint singularities² and $\dot{\mathbf{\kappa}} = (\dot{x}_p, \dot{y}_p, \dot{z}_p, \dot{\varphi})^T$ uniquely identifies the platform twist. As a consequence, Equation (2a–d) are sufficient to relate the platform twist to the actuated-joint rates, $\dot{\mathbf{q}}$, that is, they are the instantaneous input-output relationship (InI/O).

The InI/O of a manipulator is a linear mapping between platform twists and actuated-joint rates whose coefficient matrices (Jacobians) only depend on the configuration of the manipulator. The configurations (singularities) where these Jacobians have not full rank make the linear mapping not bijective and have relevant kinetostatic implications [17–21].

According to the above-deduced InI/O, if $\dot{\mathbf{q}}$ is assigned, $\dot{\mathbf{O}}_p = (\dot{x}_p, \dot{y}_p, \dot{z}_p)^T$ is always uniquely determined, but $\dot{\varphi}$ is not determined when the segments A_2D_2 and A_2A_p are aligned (i.e., a parallel singularity³ occurs). In addition, if $\mathbf{\kappa}$ is assigned, only two geometric conditions make one or more actuated-joint rates undetermined (i.e., a serial singularity⁴ occurs): (i) the 2-tuple $(\dot{\theta}_1, \dot{\theta}_2)$ is not

² Parallel manipulators with mobility lower than six that are designed to make the platform move inside a given displacement subgroup may have configurations (constraint singularities) where the platform can perform instantaneous displacements that do not belong to that subgroup [17].

³ Parallel singularities, also named type-II singularities [18], usually occur inside the workspace. They are configurations where the platform can perform instantaneous motions with locked actuators. At a type-II singularity, a load (even infinitesimal) applied to the platform needs infinitely-high generalized torques, in at least one actuator, to be balanced.

⁴ Serial singularities, also named type-I singularities [18], lie on (and identify) the workspace boundary. They are configurations where the platform stands still while the actuated joints perform instantaneous motions. At a type-I singularity, the platform can carry loads without needing that the actuators provide generalized torques to balance them.

determined when the segments A_1D_1 and D_1B_1 are aligned, and (ii) θ_3 is not determined when the segments A_2D_2 and D_2B_2 are aligned.

3. Workspace Analysis

According to the adopted notations, the geometric constants of the CRS-RRC SMG, which affect the SMG behavior, are $a_0, a_1, a_2, a_3, a_4, a_p,$ and h (Figure 1). Two of the authors, in [16], analyzed some workspace characteristics of this SMG. Such analysis brought to size some of those constants. This section summarizes and reviews the results deduced in [16].

If the two actuated R pairs of the RRC limb and the linear variable, d , of the actuated C pair are locked (i.e., by keeping z_p and A_1 fixed), the segments B_2A_p ($=O_bA_1 = \sqrt{x_p^2 + y_p^2}$), B_2D_2 ($=a_3$), D_2A_2 ($=a_4$), and A_pA_2 ($=a_p$) behave like frame, input link, coupler, and follower, respectively, of a four-bar linkage. θ_3 and φ are input and output variables, respectively, of this four-bar linkage, and the singularities of this four-bar correspond to the above-identified parallel singularity and serial singularity (ii). If this four-bar satisfies Grashof’s law [22] and A_pA_2 is the shortest bar, the platform can perform a complete rotation (i.e., the angle φ has no limitation). Nevertheless, only a double-crank four-bar can guarantee a full control of the platform rotation.

Let p and p denote the position vector (A_1-O_b) and its magnitude ($=\sqrt{x_p^2 + y_p^2}$), respectively. From an analytic point of view, the four-bar is a double-crank (i.e., the platform can perform a complete rotation fully controlled by θ_3) if and only if

$$(a_p + p) \leq (a_3 + a_4), \tag{3a}$$

$$|a_p - p| \geq |a_3 - a_4|, \tag{3b}$$

$$p = \min\{p, a_3, a_4, a_p\}. \tag{3c}$$

Inequalities (3a,b) impose that the platform can perform a full rotation and condition (3c) make the four-bar linkage double-crank.

Since there is no reason to have a_3 different from a_4 and a possible scaling factor does not affect the analysis, the choice $a_3 = a_4 = 1$ length unit (l.u.) is adopted. With this assumption, inequality (3b) become $|a_p - p| \geq 0$ and is identically satisfied; whereas, inequality (3a) and condition (3c) become

$$p \leq a_p \leq 2 - p \tag{4}$$

The blue area of Figure 3 indicates the values of the 2-tuple (p, a_p) that satisfy inequalities (4). Figure 3 shows that the maximum value, p_{Gr} , that p can assume depends only on a_p : when the values of a_p increase, p_{Gr} increases for $a_p \leq 1$ and, then, decreases for $a_p > 1$. The maximum value of p_{Gr} is 1 l.u. and is obtained with $a_p = 1$ l.u. Since p_{Gr} is the radius (see Figure 1) of the circle centered at O_b that is the region (double-crank region) where A_1 must be located to have a double-crank linkage, the choice $a_p = 1$ l.u. is adopted to maximize the double-crank region.

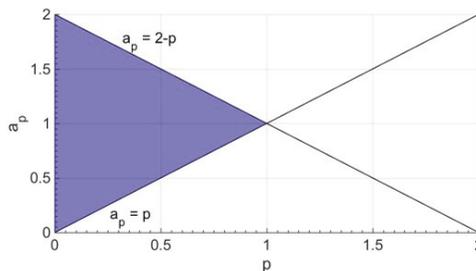


Figure 3. The blue region indicates the values of the 2-tuple (p, a_p) that satisfy inequalities (4).

Let $\Delta d (=d_{\max} - d_{\min})$ be the maximum linear stroke of the actuated C pair. The reachable workspace [23] referred to O_p is a right cylinder with height Δd , whose cross section is the intersection between two circles: one centered at O_b with radius $a_3 + a_4 + a_p$ and the other centered at B_1 with radius $a_1 + a_2$. The cross section of the reachable workspace is maximized if the smaller circle is located inside the other, that is, if

$$a_0 + a_1 + a_2 \leq a_3 + a_4 + a_p \tag{5a}$$

or

$$a_3 + a_4 + a_p \leq a_1 + a_2 - a_0. \tag{5b}$$

In addition, if such section contains the double-crank region, the dexterous workspace [23] referred to O_p is maximum, and it is equal to a right circular cylinder obtained by translating the double-crank region along the z_b axis of Δd . This condition is satisfied if

$$p_{Gr} + a_0 \leq a_1 + a_2, \tag{6a}$$

$$|a_1 - a_2| \leq a_0 - p_{Gr}, \tag{6b}$$

$$p_{Gr} \leq a_3 + a_4 + a_p. \tag{6c}$$

At the border of the double-crank region (i.e., for $p = p_{Gr}$), the four-bar linkage can still make the platform perform a complete rotation, but the linkage encounters two times the parallel singularity condition during the platform rotation. Since the integrity of parallel manipulators can be preserved only by keeping them work out and far from parallel singularities, a safe free-from-singularity dexterous workspace requires $p \ll p_{Gr}$.

Outside the double-crank region (i.e., for $p > p_{Gr}$), the four-bar linkage does not satisfy Grashof's law any longer; as a consequence, it can only be a rocker-rocker four-bar linkage and the rotation range, $\Delta\varphi$, of the platform is limited. Figure 4 shows the above-defined not-Grashof four-bar linkage at the configurations corresponding to the extreme values of φ . In a rocker-rocker four-bar, the whole $\Delta\varphi$ is swept by moving from a serial singularity (ii) to a parallel singularity (Figure 4a) or vice versa (Figure 4b). In both the cases, $\Delta\varphi$ does not change, but the configurations the mechanism passes through are different. The analysis of Figure 4 gives the following simple analytic formula for $\Delta\varphi$

$$\Delta\varphi = \pi + \cos^{-1} \left[\frac{(a_4 + a_p)^2 + p^2 - a_3^2}{2p(a_4 + a_p)} \right] - \cos^{-1} \left[\frac{(a_3 + a_4)^2 - p^2 - a_p^2}{2p a_p} \right] \tag{7}$$

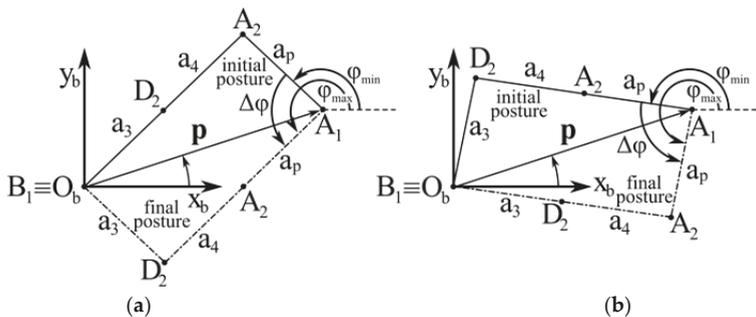


Figure 4. The above-defined four-bar linkage, with a p value that makes it a not-Grashof four-bar, at the configurations corresponding to the extreme values of φ : (a) $\Delta\varphi$ is swept by moving from a serial singularity (ii) to a parallel singularity, and (b) vice versa.

Formula (7) highlights that $\Delta\varphi$ depends only on p (i.e., the direction of the position vector p does not affect $\Delta\varphi$) and decreases when p increases. This symmetry with respect to the z_b axis and the fact that the decrease of $\Delta\varphi$ with the increase of p is not sharp [16] make wide regions of the reachable workspace that are out of the double-crank region usable for manipulation tasks that need not a complete platform rotation.

4. Kinetostatic-Performance Analysis and Dimensional Synthesis

The previous sections brought to choose $a_3 = a_4 = a_p = 1$ l.u. All the other geometric constants (i.e., $a_0, a_1, a_2,$ and h) can be used to match the adoption of a useful workspace adequate to industrial tasks with satisfactory kinetostatic properties. Even though the CRS-RRC SMG can perform some tasks outside the above-defined dexterous workspace, the choice of the useful workspace has to take into account the generality of the industrial tasks of an SMG. Consequently, in this case, the useful workspace is chosen as a right circular cylinder with axis passing through O_b (see Figure 1) and radius, p_{uw} , that satisfies the condition $p_{uw} \ll p_{Gr}$, which makes it coincide with a safe free-from-singularity dexterous workspace.

The determination of p_{uw} can be done by imposing that the transmission angle [22,24], μ , of the above-defined four-bar linkage has an acceptable value during the platform rotation. In general, tasks that require the application of relevant forces to the gripper during motion (e.g., machining tasks like drilling) need values of $|\mu - 90|$ ($^\circ$) lower than 50° and low friction in the kinematic pairs. Other manipulation tasks with reduced force interaction can accept $|\mu - 90|$ values lower than 70° ; whereas, when force interaction is not present (e.g., pick-and-place tasks) $|\mu - 90|$ could be even larger than 70° . The formulas (see [24]) that give the minimum, μ_{min} , and the maximum, μ_{max} , transmission angles, when particularized to the studied case, become ($\mu \in [0^\circ, 180^\circ]$)

$$\mu_{min} = \cos^{-1} \left[\frac{a_4^2 + a_p^2 - (p - a_3)^2}{2 a_4 a_p} \right] \tag{8a}$$

$$\mu_{max} = \cos^{-1} \left[\frac{a_4^2 + a_p^2 - (p + a_3)^2}{2 a_4 a_p} \right] \tag{8b}$$

which, for $a_3 = a_4 = a_p = 1$ l.u., give the diagrams of Figure 5.

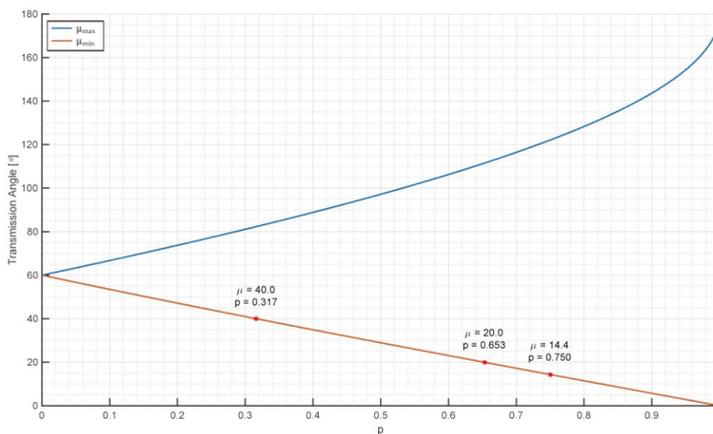


Figure 5. The minimum, μ_{min} (Equation (8a)), and the maximum, μ_{max} (Equation (8b)), transmission angles as a function of p for $a_3 = a_4 = a_p = 1$ (l.u.).

Figure 5 shows that $|\mu - 90| \leq 50^\circ$ corresponds to $p = 0.317$ and $|\mu - 90| \leq 70^\circ$ corresponds to $p = 0.653$; whereas, for $p = 0.75$, $|\mu - 90|$ exceeds 70° only in a small neighborhood of μ_{\min} which is equal to 14.4° .

Since SMGs are mainly employed in pick-and-place tasks, the choice $p_{uw} = 0.75$ is adopted by confining the force interaction tasks in region with $p \leq 0.653$ or with $p \leq 0.316$ according to the type of task.

A commercial SMG is the pickstar YS02N of Kawasaki [25]. Its useful workspace is a right circular cylinder with a base radius of 300 mm and a height of 200 mm. Hereafter, this right circular cylinder is chosen as useful workspace for the dimensional synthesis of the CRS-RRC SMG. This choice together with the previous choice $p_{uw} = 0.75$ yield 1 (l.u.) = 400 mm, that is, $a_3 = a_4 = a_p = 400$ mm.

Figure 6 shows the region of the $x_b y_b$ plane swept by the above-defined four-bar linkage with $a_3 = a_4 = a_p = 400$ mm when point A_1 is moved along the x_b axis from O_b (i.e., $A_1 = (0, 0)$) to the border of the useful workspace (i.e., $A_1 = (300, 0)$ mm). The whole region swept by the four-bar when the direction of the motion of A_1 changes can be obtained by making the region highlighted in Figure 6 rotate around O_b in the $x_b y_b$ plane. Such rotation yields a circle with a radius of 700 mm. Consequently, by taking into account the physical sizes of the links, the choice $a_0 = 800$ mm is adopted in order to avoid interferences between the frame and the four-bar during motion.

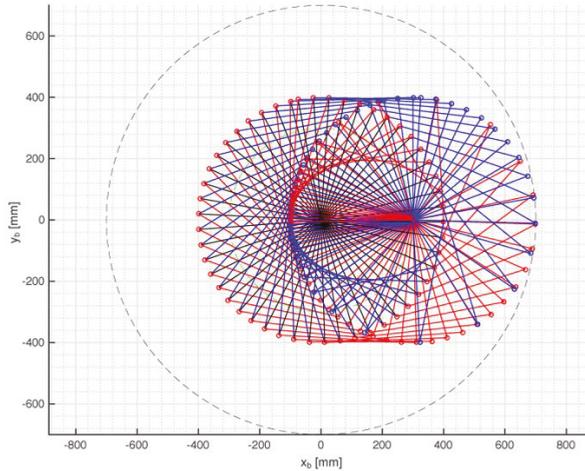


Figure 6. Region of the $x_b y_b$ plane swept during motion by the above-defined four-bar linkage with $a_3 = a_4 = a_p = 400$ mm and a useful workspace with a base radius of 300 mm.

4.1. Measure of the Kinetostatic Performances

The matrix form of system (2) (i.e., of the InI/O) is

$$J_k \dot{k} = J_q \dot{q} \tag{9}$$

with

$$J_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ m_x & m_y & 0 & a_p(m_y \cos \varphi - m_x \sin \varphi) \end{bmatrix} \tag{10a}$$

$$J_q = \begin{bmatrix} -[a_1 \sin \theta_1 + a_2 \sin(\theta_1 + \theta_2)] & -a_2 \sin(\theta_1 + \theta_2) & 0 & 0 \\ [a_1 \cos \theta_1 + a_2 \cos(\theta_1 + \theta_2)] & a_2 \cos(\theta_1 + \theta_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_3(m_y \cos \theta_3 - m_x \sin \theta_3) & 0 \end{bmatrix} \quad (10b)$$

Kinestatic performances of manipulators can be evaluated by using indices built by using the Jacobians J_k and J_q of Equation (9). Two of these indices are the “conditioning index” (CI) [26] of the matrix $J = (J_q)^{-1}J_k$, which is defined as the inverse of the condition number of J , and its average value on the useful workspace, named “global conditioning index” (GCI). The computation of the CI and GCI needs that the entries of the Jacobians J_k and J_q be homogeneous [9,27–29]. Jacobian homogenization can be obtained with the introduction of a characteristic length, λ , through a change of variables [27], even though which λ should be used is an open problem.

In the studied case, the introduction of the new homogeneous variables $\dot{\kappa}_h = \left(\frac{\dot{x}_p}{\lambda}, \frac{\dot{y}_p}{\lambda}, \frac{\dot{z}_p}{\lambda}, \dot{\varphi} \right)^T$ and $\dot{q}_h = \left(\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \frac{\dot{d}}{\lambda} \right)^T$ into Equation (9) yields

$$J_h \dot{\kappa}_h = \dot{q}_h \quad (11)$$

where $J_h = (J_{qh})^{-1}J_{kh}$ with

$$J_{kh} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{m_x}{\lambda} & \frac{m_y}{\lambda} & 0 & \frac{a_p}{\lambda^2}(m_y \cos \varphi - m_x \sin \varphi) \end{bmatrix} \quad (12a)$$

$$J_{qh} = \begin{bmatrix} \frac{-[a_1 \sin \theta_1 + a_2 \sin(\theta_1 + \theta_2)]}{\lambda} & -\frac{a_2}{\lambda} \sin(\theta_1 + \theta_2) & 0 & 0 \\ \frac{[a_1 \cos \theta_1 + a_2 \cos(\theta_1 + \theta_2)]}{\lambda} & \frac{a_2}{\lambda} \cos(\theta_1 + \theta_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{a_3}{\lambda^2}(m_y \cos \theta_3 - m_x \sin \theta_3) & 0 \end{bmatrix} \quad (12b)$$

J_h is a homogeneous Jacobian with dimensionless entries. The condition number, χ , of J_h is, by definition, $\chi = \|J_h\| \|J_h^{-1}\|$ where $\|(\cdot)\|$ stands for any matrix norm of the argument, if the Frobenius norm [21] is adopted $\|J_h\| = \sqrt{\text{trace}(J_h J_h^T)}$. Consequently, the conditioning index, $CI = 1/\chi$, depends both on the SMG configuration and on the geometric constants a_1 and a_2 that have not been determined, yet. The CI ranges from 0, at parallel singularities to 1 at isotropic configurations (i.e., SMG configurations where J_h is proportionate to the identity matrix), which are the farthest from parallel singularities. The CI is a local index that measures the kinestatic performance of the SMG at a configuration; whereas, the GCI (i.e., its average value on the useful workspace) gives a score to the kinestatic performance of the SMG and could be used to compare different manipulators.

Optimizing the kinestatic performances of a manipulator by using the CI and the GCI means determining the available geometric constants (in this case, a_1 and a_2) so that the minimum value, CI_{\min} , of the CI and the GCI are as high as possible for that architecture. Such optimization is presented in the following subsection.

4.2. Dimensional Synthesis

The admissible values of a_1 and a_2 must satisfy inequalities (6a,b) with $a_0 = 800$ mm and $p_{Gr} = a_p = 400$ mm. Moreover, a reasonable choice for the RRC limb would be $a_2 \leq a_1$. All these inequalities are satisfied in the region highlighted in blue of Figure 7.

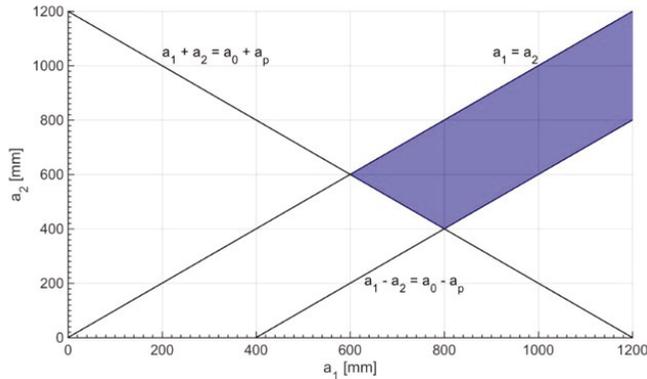


Figure 7. The values of (a_1, a_2) that satisfy inequalities (6a), (6b) and $a_2 \leq a_1$ with $a_0 = 800$ mm and $p_{Gr} = a_p = 400$ mm are those belonging to the blue region.

By setting $\lambda = 400$ mm (i.e., equal to the length of a_p , a_3 and a_4 and to the arithmetic mean of the diameter and the height of the useful workspace), a numerical algorithm has been used to compute the GCI and the CI_{min} referred to the useful workspace for each admissible values of (a_1, a_2) . The results of these computations are shown in Figure 8.

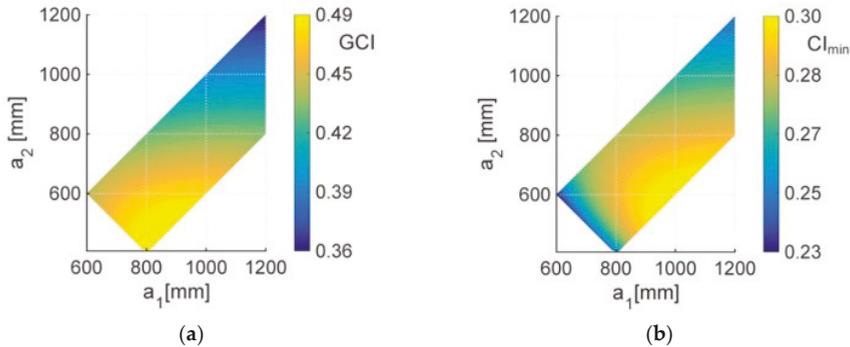


Figure 8. Kinetostatic performance as a function of a_1 and a_2 : (a) global conditioning index (GCI) evaluated on the useful workspace, and (b) CI_{min} in the useful workspace.

The analysis of the GCI values displayed in Figure 8a reveals that the maximum GCI is equal to 0.48795 and is obtained with $a_1 = 869$ mm and $a_2 = 469$ mm, which correspond to $CI_{min} = 0.28682$ (Figure 8b). Also, Figure 8a shows that the maximum GCI falls in a smooth region (more or less flat) which allows large variation of (a_1, a_2) with small reductions of the GCI.

On the other side, the analysis of the CI_{min} values displayed in Figure 8b reveals that the maximum CI_{min} is equal to 0.29806 and is obtained with $a_1 = 993$ mm and $a_2 = 593$ mm, which correspond to $GCI = 0.4763$ (Figure 8a). Moreover, Figure 8b shows that also the maximum CI_{min} falls in a smooth region (more or less flat). These results bring to choose $a_1 = 950$ mm and $a_2 = 600$ mm which correspond to a good compromise with $GCI = 0.482$ and $CI_{min} = 0.29743$ that are values near enough to their maxima. The chosen values of a_1 and a_2 yield the minimum CI values at each A_1 position inside the useful workspace shown in Figure 9. Figure 9 highlights that most of the useful workspace has $CI \geq 0.45$, which makes the CI distribution acceptable.

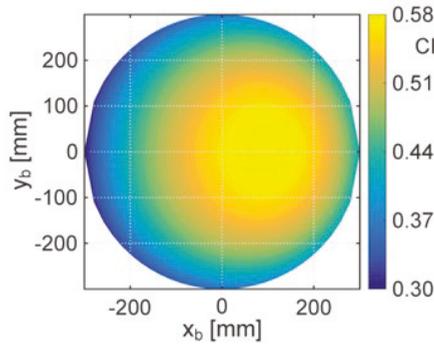


Figure 9. Minimum conditioning index (CI) values at each A_1 position inside the useful workspace for $a_1 = 950$ mm and $a_2 = 600$ mm.

4.3. The 3D Model

The dimensional synthesis, with useful workspace assigned as a right circular cylinder with a radius of 300 mm and a height of 200 mm, brought to choose the following values of the geometric constants: $a_0 = 800$ mm, $a_1 = 950$ mm, $a_2 = 600$ mm, $a_3 = 400$ mm, $a_4 = 400$ mm, and $a_p = 400$ mm. Figure 10 shows the region of the $x_b y_b$ plane swept by the RRC limb, with $a_0 = 800$ mm, $a_1 = 950$ mm, and $a_2 = 600$ mm, when point A_1 is moved on the whole circular boundary of the useful workspace and the limb is assembled in either of the two assembly modes the IPA identifies [13]. The analysis of Figure 10 highlights that no link interference occurs with the chosen geometric data. By combining Figures 6 and 10 the overall region of the $x_b y_b$ plane that is swept by both the limbs is obtained. Figure 11 shows such region.

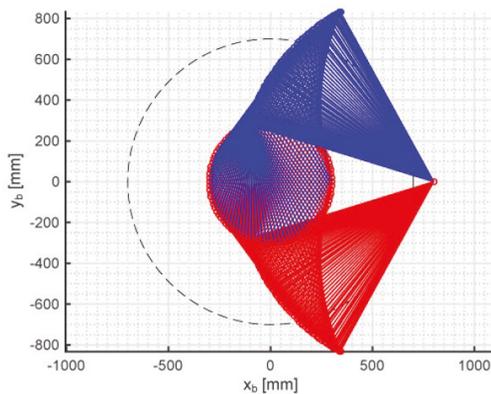


Figure 10. Region of the $x_b y_b$ plane swept by the RRC limb, with $a_0 = 800$ mm, $a_1 = 950$ mm, and $a_2 = 600$ mm, when point A_1 is moved on the whole circular border of the useful workspace and the limb is assembled in either of the two assembly modes the inverse position analysis (IPA) identifies.

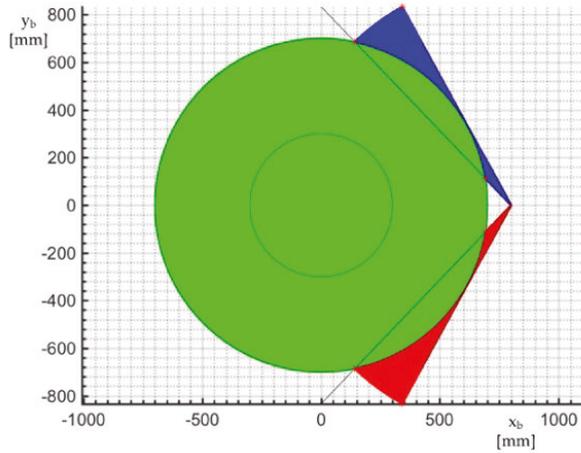


Figure 11. Overall region of the x_b, y_b plane that is swept by both the limbs.

In order to solve all the doubts about the actual feasibility of the machine, a CAD model of the CRS-RRC SMG has been built with the chosen geometric data. Figure 12 shows the 3D view (Figure 12a) and the lateral view (Figure 12b) of this model. Such model has an L-shaped base, only commercial actuators that are all mounted on the base, rolling bearings in all the R pairs to avoid jamming of the above-defined four-bar linkage at low transmission angles, a spherical roller bearings that implements the S-pair constraint⁵, and a linear ball bearing that implements the constraint of the passive C-pair. The bevel gearbox and the toothed belts (Figures 2 and 12) are commercial products, too.

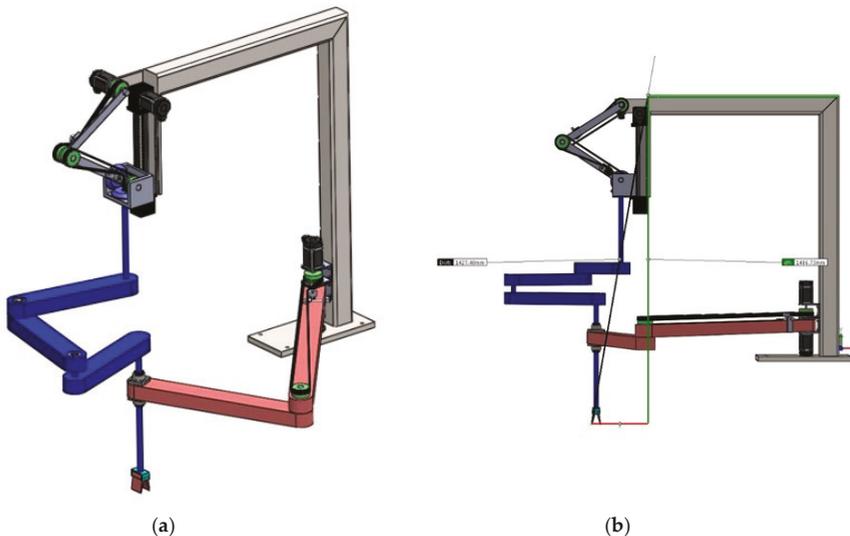


Figure 12. CAD model: (a) 3D view, and (b) lateral view.

⁵ In the studied SMG, the S pair has the only role of compensating errors of parallelism among the R-pair axes.

5. Summary of the Results and Discussion

The kinematic analysis [13] of the CRS-RRC SMG showed that both the DPA and IPA are solvable with simple and explicit formulas, and that platform's translation and rotation are decoupled. Therefore, the control algorithms are easy to implement and require very low computation time, which makes it possible to build a fast online control system.

The above-reviewed workspace analysis (see also Ref. [16]) highlighted that the reachable workspace has axially-symmetric properties. In particular, a safe free-from-singularity dexterous workspace, which is a right circular cylinder, is easy to identify in a wider region of the reachable workspace where the platform can perform complete rotations. Moreover, parallel singularities occur only for two known platform orientations and, in the reachable workspace, they are located on a circle that is the boundary of the dexterous workspace. Thus, the platform can pass through the parallel-singularity locus simply by changing its orientation and, if necessary, can accomplish particular tasks in region where the platform rotation is bounded.

The kinetostatic analysis based on the transmission angle and the conditioning index revealed that the machine can be so dimensioned that the GCI and the CI_{\min} are acceptable in a useful workspace, equal to those of commercial SMGs, with transmission angles adequate to the generality of industrial tasks.

The CAD model of the CRS-RRC SMG showed that its manufacture needs only commercial components. This result proves the actual feasibility of the machine with cheap production processes. Moreover, it highlights that the L-shaped base when fixed on a rotating frame allows an easy way for setting up the machine with respect to different machining planes (e.g., belt conveyors).

6. Conclusions

Kinetostatic indices have been used to complete the dimensional synthesis of the SMG of type CRS-RRC previously presented by the first author. In particular, a safe free-from-singularity useful workspace equal to that of a commercial SMG (the pickstar YS02N of Kawasaki [25]) has been chosen and all the geometric constants of this machine have been determined so that the minimum CI and the GCI are maximized.

Then, the so-determined geometric constants have been used to build a realistic 3D model that involves only commercial components, and implements an original architecture for actuated C pairs that brings all the actuators on the frame.

The obtained results positively close the validation of the machine concept and open to the structural and dynamic analyses/checks necessary to match static and dynamic requirements for an assigned payload. These analyses together with stiffness and accuracy analyses are the next steps of this project.

Author Contributions: Data curation, M.C. and H.S.; Formal analysis, R.D.G.; Funding acquisition, R.D.G. and H.S.; Investigation, R.D.G., M.C. and H.S.; Methodology, R.D.G.; Project administration, R.D.G.; Software, M.C.

Funding: This work has been developed at the Laboratory of Advanced Mechanics (MECH-LAV) of Ferrara Technopole, supported by FAR2017 UNIFE funds, by Regione Emilia Romagna (District Councillorship for Productive Assets, Economic Development, Telematic Plan) POR-FESR 2007-2013, Attività I.1.1, and by CNPq—Conselho Nacional de Desenvolvimento Científico e Tecnológico (National Council for Scientific and Technological Development), Brazil.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Hervé, J.M. The Lie group of rigid body displacements, a fundamental tool for mechanism design. *Mech. Mach. Theory* **1999**, *34*, 719–730. [[CrossRef](#)]

2. Bottema, O.; Roth, B. *Theoretical Kinematics*; Courier Dover Publications: Amsterdam, The Netherlands, 1979; ISBN 0-444-85124-0.
3. Lee, C.-C.; Hervé, J.M. Type Synthesis of Primitive Schoenflies-Motion Generators. *Mech. Mach. Theory* **2009**, *44*, 1980–1997. [[CrossRef](#)]
4. Lee, C.-C.; Hervé, J.M. Isoconstrained Parallel Generators of Schoenflies Motion. *ASME J. Mech. Robot.* **2011**, *3*, 021006. [[CrossRef](#)]
5. Pierrot, F.; Company, O. H4: A New Family of 4-dof Parallel Robots. In Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM'99), Atlanta, GA, USA, 19–23 September 1999; pp. 508–513.
6. Krut, S.; Company, O.; Benoit, M.; Ota, H.; Pierrot, F. I4: A new parallel mechanism for SCARA motions. In Proceedings of the 2003 International Conference on Robotics and Automation, Taipei, Taiwan, 14–19 September 2003; pp. 1875–1880.
7. Nabat, V.; de la O. Rodriguez, M.; Company, O.; Krut, S.; Pierrot, F. Par4: very high speed parallel robot for pick-and-place. In Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, AB, Canada, 2–6 August 2005; pp. 553–558.
8. Angeles, J.; Caro, S.; Khan, W.; Morozov, A. The Design and Prototyping of an Innovative Schoenflies Motion Generator. *Proc. IMechE-Part C J. Mech. Eng.* **2006**, *220*, 935–944. [[CrossRef](#)]
9. Li, Z.; Lou, Y.; Zhang, Y.; Liao, B.; Li, Z. Type Synthesis, Kinematic Analysis, and Optimal Design of a Novel Class of Schoenflies-Motion Parallel Manipulators. *IEEE Trans. Autom. Sci. Eng.* **2013**, *10*, 674–686.
10. Kong, X.; Gosselin, C.M. Type Synthesis of 3T1R 4-dof Parallel Manipulators Based on Screw Theory. *IEEE Trans. Robot. Autom.* **2004**, *20*, 181–190. [[CrossRef](#)]
11. Company, O.; Pierrot, F.; Nabat, V.; Rodriguez, M. Schoenflies Motion Generator: A New Non Redundant Parallel Manipulator with Unlimited Rotation Capability. In Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 18–22 April 2005; pp. 3250–3255.
12. Richard, P.-L.; Gosselin, C.M.; Kong, X. Kinematic Analysis and Prototyping of a Partially Decoupled 4-DOF 3T1R Parallel Manipulator. *ASME J. Mech. Des.* **2007**, *129*, 611–616. [[CrossRef](#)]
13. Di Gregorio, R. A Novel Single-Loop Decoupled Schoenflies-Motion Generator: Concept and Kinematics Analysis. In *Advances in Service and Industrial Robotics*; Series: Mechanisms and Machine Science; Ferraresi, C., Quaglia, G., Eds.; Springer: New York, NY, USA, 2018; Volume 49, pp. 11–18. ISBN 978-3-319-61275-1. [[CrossRef](#)]
14. Pierrot, F.; Company, O.; Krut, S.; Nabat, V. Four-Dof PKM with Articulated Travelling-Plate. In *Parallel Kinematic Machines in Research and Practice, Proceedings of the the 5th Chemnitz Parallel Kinematics Seminar, Chemnitz, Germany, 25–26 April 2006*; Neugebauer, R., Ed.; Verlag Wissenschaftliche Scripten: Auerbach, Germany, 2006; Volume 33, pp. 677–694. ISBN 9783937524405. Available online: <https://hal-lirmm.ccsd.cnrs.fr/lirmm-00105558> (accessed on 14 September 2019).
15. Simas, H.; Di Gregorio, R. Kinematics of a Particular 3T1R Parallel Manipulator of Type 2PRPU. In Proceedings of the ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Cleveland, OH, USA, 6–9 August 2017; Paper No. DETC2017-67174. Volume 5A, ISBN 978-0-7918-5817-2. [[CrossRef](#)]
16. Simas, H.; Di Gregorio, R. Workspace Analysis and Dimensional Synthesis of the PRRS-RRC Schoenflies-Motion Generator. In *Mechanism Design for Robotics, Proceedings of the 4th IFToMM Symposium on Mechanism Design for Robotics, Udine, Italy, 11–13 September 2018*; Gasparetto, A., Ceccarelli, M., Eds.; Springer: New York, NY, USA, 2018; pp. 2011–2018.
17. Zlatanov, D.; Bonev, I.A.; Gosselin, C.M. Constraint singularities of parallel mechanisms. In Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA 2002), Washington, DC, USA, 11–15 May 2002; pp. 496–502.
18. Gosselin, C.M.; Angeles, J. Singularity analysis of closed-loop kinematic chains. *IEEE Trans. Robot. Autom.* **1990**, *6*, 281–290. [[CrossRef](#)]
19. Ma, O.; Angeles, J. Architecture singularities of platform manipulators. In Proceedings of the 1991 IEEE International Conference on Robotics and Automation (ICRA 1991), Sacramento, CA, USA, 9–11 April 1991; pp. 1542–1547.
20. Zlatanov, D.; Fenton, R.G.; Benhabib, B. A unifying framework for classification and interpretation of mechanism singularities. *ASME J. Mech. Des.* **1995**, *117*, 566–572. [[CrossRef](#)]

21. Meyer, C.D. *Matrix Analysis and Applied Linear Algebra*; Society for Industrial & Applied Mathematics (SIAM): Philadelphia, PA, USA, 2000; ISBN 978-0898714548.
22. Shigley, J.E.; Uicker, J.J. *Theory of Machines and Mechanisms*, 2nd ed.; McGraw-Hill: New York, NY, USA, 1995; pp. 33–36. ISBN 9780071137475.
23. Siciliano, B.; Sciavicco, L.; Villani, L.; Oriolo, G. *Robotics: Modelling, Planning and Control*; Springer: London, UK, 2009; p. 85. ISBN 978-1-84628-641-4.
24. Pennestri, E.; Valentini, P.P. A review of simple analytical methods for the kinematic synthesis of four-bar and slider-crank function generators for two and three prescribed finite positions. In *Buletin Stiintific Seria Mecanica Aplicata*; University of Pitesti: Pitești, Romania, 2009; pp. 128–143. [[CrossRef](#)]
25. Kawasaki. Y-Series: Ultra-High Speed Delta Robot. 2012. Available online: http://larrarioz.com/_lib/pdf/Kawasaki/DS_Robot_YS002N.pdf (accessed on 16 July 2018).
26. Gosselin, C.; Angeles, J. A global performance index for the kinematic optimization of robotic manipulators. *ASME J. Mech. Des.* **1991**, *113*, 220–226. [[CrossRef](#)]
27. Ma, O.; Angeles, J. Optimum architecture design of platform manipulators. In Proceedings of the 5th International Conference Advanced Robotics (ICAR1991), Pisa, Italy, 19–22 June 1991; pp. 1130–1135.
28. Kim, S.; Ryu, J. New dimensionally homogeneous Jacobian matrix formulation by three end-effector points for optimal design of parallel manipulators. *IEEE Trans. Robot. Autom.* **2003**, *19*, 731–736.
29. Kim, S.M.; Kim, W.; Yi, B.-J. Kinematic Analysis and Design of a New 3T1R 4-DOF Parallel Mechanism with Rotational Pitch Motion. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 5167–5172.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Cable Robot Performance Evaluation by Wrench Exertion Capability

Giovanni Boschetti * and Alberto Trevisani

Department of Management and Engineering, Università degli Studi di Padova, 36100 Vicenza, Italy; alberto.trevisani@unipd.it

* Correspondence: giovanni.boschetti@unipd.it; Tel.: +39-0444-99-8748

Received: 22 February 2018; Accepted: 25 March 2018; Published: 27 March 2018

Abstract: Although cable driven robots are a type of parallel manipulators, the evaluation of their performances cannot be carried out using the performance indices already developed for parallel robots with rigid links. This is an obvious consequence of the peculiar features of flexible cables—a cable can only exert a tensile and limited force in the direction of the cable itself. A comprehensive performance evaluation can certainly be attained by computing the maximum force (or torque) that can be exerted by the cables on the moving platform along a specific (or any) direction within the whole workspace. This is the idea behind the index—called the Wrench Exertion Capability (*WEC*)—which can be employed to evaluate the performance of any cable robot topology and is characterized by an efficient and simple formulation based on linear programming. By significantly improving a preliminary computation method for the *WEC*, this paper proposes an ultimate formulation suitable for any cable robot topology. Several numerical investigations on planar and spatial cable robots are presented to give evidence of the *WEC* usefulness, comparisons with popular performance indices are also provided.

Keywords: cable driven robot; performance index; Wrench Exertion Capability

1. Introduction

Cable driven robots, or simply cable robots, are relatively simple parallel manipulators, operating in planar or spatial arrangements, formed by attaching multiple cables to a moving platform, on which the end-effector is fitted. In cable robots, the cables are driven by motors which can extend or retract the cables by winding or unwinding them from pulleys (also called winches or drums). In this sense cables are usually said to be active. Cable robots have several desirable advantages compared to conventional serial and parallel robots. For this reason they have been studied thoroughly since the early 90's [1] and promise to significantly increase performances of today's industrial robots in terms of payload, workspace and dynamic performances: they can be designed to have a very large workspace, a very high load capacity, or to generate very high speed motions [2], always with considerable energy efficiency. Their unique features, arising from parallel kinematics combined to minimal moving masses, make them amongst the most promising robotic devices in the industrial and service field, as it is proved by the ever-growing number of cable robot families that has been developed by research institutions and private companies [3,4].

Very often, cable robots are designed to be redundant (i.e., with more active cables and hence motors, than degrees of freedom (dofs) of the moving platform, see for example, Reference [2]), however, fully actuated (i.e., with a number of active cables equal to the dofs, see for example, Reference [5]) and underactuated (i.e., with less active cables than dofs, see for example, Reference [6]) topologies have been studied too. Moreover, cable robots can exhibit a hybrid design (i.e., with both cables and rigid link mechanisms [3,7]). Redundant cable robots are the sole robots which can

completely restrain the moving platform of a cable robot: in order to fully constrain the moving platform of a cable robot, it is required that the number of cables is greater by one than the number of dofs of the moving platform (see for example, [8]), hence a redundant configuration is needed. The condition on the number of cables is only necessary but not sufficient: a cable robot can be underconstrained, even if the number of cables is greater than the number of dofs. It is typically the case of the so-called cable suspended robots. Generally speaking, a high number of cables may lead to overconstrained configurations (see for example, Reference [2]), while a lower number of cables leads necessarily to underconstrained robots, which must rely on gravity to keep positive tensions in the cables (see for example, References [9,10]). Indeed, contrary to fully constrained or overconstrained cable robots, the underconstrained ones cannot take advantage of redundant cables to set a desired tension distribution in the cables. This makes operating underactuated cable robots particularly challenging.

A major requirement that has to be met in cable robots is ensuring that during operation all the cables are under adequate tension (at least cable slackness must be prevented in all the cables) and that such a tension is below the maximum permissible value related to the torque limits of the winch motors or to the tensile force limits of the cables [5]. In practice, this makes it necessary to take into account explicitly the bilateral bounds on cable tensions reflecting both the unilateral nature of cables as actuators (cables can pull but are unable to push the end-effector) and the additional constraints posed, on the upper bound, by cable and motor physical properties and, on the lower bound, by safety margins or end-effector stiffness requirements [11,12]. The latter requirements usually suggest imposing a lower bound for cables forces greater than 0. Clearly, the evaluation of the performances of a cable robot cannot neglect such peculiarities of cable robots and the complexity arising from the need of keeping bounded cable tensions. As a result, though cable driven robots are basically parallel robots, the traditional performance indices developed for parallel robots (see for example, [13–16]) are inadequate and cannot be employed straightforwardly.

So far, just a few examples of performance indices for cable robots have been proposed in literature. They have mainly been conceived as extensions to cable robots of traditional Jacobian-based performance indices. In [17] an evolution of the Yoshikawa manipulability has been proposed. In [18] the condition number has instead been applied as is, by restricting the analysis to a specific workspace. In [19] the Kinematic Sensitivity Index [20] has been modified in order to achieve the best workspace region of cable robots. An evolution of the isotropy index, called tension factor, has been proposed in [21]: the tension factor is an isotropy index, defined in the joint space, which evaluates the ratio between minimum and maximum cable tensions. Another interesting isotropy index has been defined in [9] for evaluating the inertial properties of two cable robots designed for rehabilitation. In the same work, a maximum isotropic force has been defined in order to find the minimum force that can be exerted in any direction. The index in [8] has been then extended for application to reconfigurable cable robots [22,23].

In [24] a novel approach to cable robot performance evaluation has been proposed and applied to solely redundant cable robots. The approach is based on the computation of the maximum force which can be exerted by the active cables on the moving platform along a specific direction. By extending the reasoning behind such an approach, in [25] a novel performance index called Wrench Exertion Capability (*WEC*) has been firstly introduced. The reason for referring the evaluation to a given direction comes from a typical practical need when designing a cable robot: predicting the maximum force or torque that can be exerted on the moving platform along a direction of interest, usually keeping either null or limited wrench components, both in terms of forces and torques, along the other directions. This is basically what is meant by evaluation of the *WEC* of a cable robot along a direction. In [25] a preliminary formulation of the method has been introduced, as well as a validation restricted to planar cable robots. This paper improves such a formulation and extends it to spatial cable robots therefore setting an ultimate formulation of the method.

The *WEC* appears more versatile than other performance indices since it can be used to perform various analyses. Not only can the *WEC* be employed for maximum force/torque evaluations but also to compute the minimum force/torque values which can be guaranteed throughout the workspace, irrespective of the direction and for isotropy evaluations. All these analyses allow getting considerable insight into cable robot performances and give the possibility to perform comprehensive comparisons among the performances of cable robots with different topologies and alternative cable layouts: indeed, such comparative investigations often need to be carried out since it is apparent that not only are the performances of cable robots influenced by the number of active cables but also by their geometrical arrangement. A challenging example of *WEC* employment is given by some recent recovery strategies in case of cable failure [26,27].

The paper is organized as follows. The *WEC* formulation is first developed in Section 2: the formulation is based on the theory developed in [25] which is here revised and extended to generic spatial robots. In Section 3 three illustrative examples of computation of the *WEC* are provided. Firstly, the *WEC* is employed to compare the performances of two fully actuated planar cable robots with different cable layouts. A comparison is also made with state-of-the-art performance indices. Secondly, performance changes due to variations in the number of cables are investigated by referring to a planar and cable suspended robot. Thirdly, an investigation is proposed to show the benefits related to the use of the *WEC* in the evaluation of the performances of a spatial cable robot. The conclusions are stated in Section 4.

2. Wrench Exertion Capability

Suppose that for a given pose of the moving platform of a cable robot you were interested in evaluating the maximum force or torque that cables can exert on the platform, along a certain direction. Such an interest could be motivated by a variety of reasons, including, for example, the need to identify the regions of the workspace where the robot best performs in terms of initial acceleration, payload capacity or capability to react to external disturbances (forces or torques). Clearly, the maximum force or torque that the cables can exert on the moving platform, along a given direction, depends on the maximum force that each cable can exert. Less obviously, such a wrench exertion capability also depends on the minimum tension of the cables that must be guaranteed to avoid cable slackness or to meet a desired stiffness requirement for the robot platform. At least, cable tension must be greater than zero to guarantee that cable forces can be maintained tensile. Computing the *WEC* index for a cable robot basically consists in performing the aforementioned evaluation taking into account cable tension limits explicitly.

The computation of the *WEC* suggested in this work is based on the solution of a linear programming problem involving cable tensions, cable tension limits and a novel representation of the so-called wrench matrix. The wrench matrix S of a cable robot, also called structure matrix, is usually defined as the matrix relating the wrench w_c exerted by the cable forces on the moving platform to the tension vector τ containing the cable forces τ_i (see Figure 1). It can be immediately recognized that in the most general case of a spatial cable robot driven by m cables, it holds: $w_c = S\tau$, where the structure matrix S takes the following form:

$$S = \begin{bmatrix} u_1 & u_2 & \dots & u_m \\ r_1 \times u_1 & r_2 \times u_2 & \dots & r_m \times u_m \end{bmatrix} \tag{1}$$

As schematically shown in Figure 1, vectors u_i and r_i are respectively the unit vector running along the i th cable, (oriented from the moving platform, that is, the box painted in grey in the scheme, towards the i th cable output point on the fixed frame) and the vector from the centre of mass (G) of the moving platform to the point where the i th cable is connected to the moving platform.

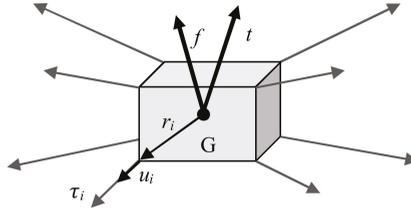


Figure 1. Schematic representation of the moving platform of a spatial cable robot and of the vectors involved in the computation of the wrench exerted by the cables on the platform.

The structure matrix S only allows computing the cable wrench w_c exerted by the cables on the moving platform. In general, this is not the sole wrench applied to the moving platform. In order to compute the total wrench $w := \begin{bmatrix} f^T & t^T \end{bmatrix}^T$ applied to the moving platform, external loading, including, for example, gravity force, should be taken into account. In the previous definition of w vectors f and t are respectively the overall forces and torques exerted on the moving platform by the cables and the external forces. To account explicitly for external forces, a novel definition for the wrench matrix (denoted by W) is introduced, which is obtained by simply aggregating the structure matrix S and the external wrench w_e :

$$w = w_c + w_e = S\tau + w_e = [S \ w_e] \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} := W \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \tag{2}$$

Once the matrix definition of W in Equation (2) is introduced, it is possible to develop cable robot performance analysis following a well-established approach. It is common knowledge that in the performance analysis of parallel manipulators it has been proved convenient to split Jacobian matrices into their “translational” and “rotational” parts [13] in order to evaluate independently the translational and rotational capabilities of parallel robots. By applying the same idea to the novel definition of the wrench matrix of a cable robot, it is here suggested to split W into two parts, namely W_f and W_t (where $W := \begin{bmatrix} W_f^T & W_t^T \end{bmatrix}^T$) to analyze separately force and torque exertion capabilities.

The proposed analysis is particularly useful when it is referred to a specific direction of interest not necessarily coinciding with an axis of the absolute reference frame. Since the force and torque components in vector w are expressed in an absolute reference frame, in order to refer the evaluation to a specific direction d , a rotation matrix R can be introduced to define such a direction of interest univocally in the absolute reference frame adopted [13]. Once the direction d is defined, symbols $o1$ and $o2$ are used to denote two orthogonal directions which, combined with d , provide a Cartesian reference frame. The following expressions can be adopted to rotate matrices W_f and W_t :

$$\begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix}^T \begin{Bmatrix} W_f \\ W_t \end{Bmatrix} := \begin{bmatrix} w_{fd}^T & w_{fo1}^T & w_{fo2}^T & w_{td}^T & w_{to1}^T & w_{to2}^T \end{bmatrix}^T \tag{3}$$

Then, for example, the WEC of a fully constrained cable robot can be expressed in terms of the maximum force w_{fd} that can be exerted along the direction d while keeping bounded cable tensions and given values \tilde{w}_R of the other wrench components. Such a WEC may be referred to as WEC_{cd}^f , since it involves a force evaluation along a direction of interest d and can be computed by solving the

following linear programming problem (henceforth, the symbol \preceq stands for the component-wise inequality):

$$WEC_d^f := \max \left(w_{f_d} = w_{f_d}^T \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \right) \text{ s.t. : } \left\{ \begin{array}{l} \begin{bmatrix} w_{f_{o1}}^T \\ w_{f_{o2}}^T \\ w_{f_d}^T \\ w_{t_{o1}}^T \\ w_{t_{o2}}^T \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} := A \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} := \tilde{w}_R \\ \tau_{min} \preceq \tau \preceq \tau_{max} \end{array} \right. \quad (4)$$

Similarly, if the torque exertion capability along a direction d is to be evaluated, the WEC_d^t can be computed by simply solving the linear programming problem stated as follows:

$$WEC_d^t := \max \left(w_{t_d} = w_{t_d}^T \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \right) \text{ s.t. : } \left\{ \begin{array}{l} \begin{bmatrix} w_{f_d}^T \\ w_{f_{o1}}^T \\ w_{f_{o2}}^T \\ w_{t_{o1}}^T \\ w_{t_{o2}}^T \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = A \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = \tilde{w}_R \\ \tau_{min} \preceq \tau \preceq \tau_{max} \end{array} \right. \quad (5)$$

In general, a default value for \tilde{w}_R should be 0 because, when a cable robot is designed, a practical need may consist in predicting the maximum force that can be exerted along a prescribed direction while keeping null wrench components, both in terms of forces and torques, along the other directions. For example, this is the case when it is necessary to accelerate the moving platform along a specific direction belonging to a path, while keeping the platform orientation unaltered. This is coherent with the typical investigation objectives presented at the beginning of this section.

The possibility of meeting imposed requirements on the full set of values \tilde{w}_R of the wrench components excluding the one which is maximized, may only be assured in fully actuated and redundant cable robots. Conversely, when a cable robot is underactuated this is generally impossible. Nonetheless, a suitable redefinition of the linear programming problem allows extending the application of the WEC index to such cable robots. Indeed, in underactuated cable robots it is impossible to apply the proposed optimization unless enough equations in the linear problem $A \{ \tau^T \ 1 \}^T = \tilde{w}_R$ are removed. This is a consequence of the fact that it is impossible to assign finite values to all the \tilde{w}_R components but only to $m - 1$ of them, where m is the number of active cables. Instead of just removing from the linear programming problem the proper number of equality constraints, we suggest replacing them with inequality constraints imposing upper and lower bounds to the wrench components to which finite values cannot be assigned. As an example, consider the problem statement in Equation (6) which refers to a spatial cable robot with 6 dofs driven by 3 active cables: in order to compute a WEC_d^f , finite values are assigned only to 2 force components (\tilde{w}_R) while the torque components of the overall wrench are limited by upper and lower bounds (\tilde{w}_B).

$$WEC_d^f := \max \left(w_{f_d} = w_{f_d}^T \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \right) \text{ s.t. : } \left\{ \begin{array}{l} \begin{bmatrix} w_{f_{o1}}^T \\ w_{f_{o2}}^T \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = A \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = \tilde{w}_R \\ \begin{bmatrix} w_{t_d}^T \\ w_{t_{o1}}^T \\ w_{t_{o2}}^T \\ -w_{t_d}^T \\ -w_{t_{o1}}^T \\ -w_{t_{o2}}^T \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = \begin{bmatrix} B \\ -B \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \preceq \tilde{w}_B \\ \tau_{min} \preceq \tau \preceq \tau_{max} \end{array} \right. \quad (6)$$

In particular, if we compare Equation (6) with the formulation in Equation (4), it is apparent that the constraints in the form of equalities removed from $A\{\tau^T \ 1\}^T = \tilde{w}_R$ have been replaced by a suitable set of constraints in the form of inequalities $[B^T - B^T]^T \{\tau^T \ 1\}^T \preceq \tilde{w}_B$.

The presented extension of the WEC definition to underactuated cable robots addresses a more general problem and gives the opportunity to make comparisons among considerably different robot topologies keeping a goal-driven approach.

Equation (7) provides a conclusive and general formulation of the WEC, suitable to any cable robot topology, where Γ is a generalized force (i.e., Γ can be either a force or torque component of the wrench vector w projected along the direction d of interest).

$$WEC_d^\Gamma := \max(w_{\Gamma,d}) \text{ s.t. : } \begin{cases} A \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = \tilde{w}_R \\ \begin{bmatrix} B \\ -B \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \preceq \tilde{w}_B \\ \tau_{min} \preceq \tau \preceq \tau_{max} \end{cases} \quad (7)$$

It is worth highlighting that in Equation (7) the dimensions of the matrices A and B are related to the cable robot topology and to the specific constraints defined for the analysis, reflecting operational requirements or specific features of the application. Let k be the number of rows of matrix A (i.e., the number of linear equality constraints). The following inequality must always hold: $k \leq \min(n - 1, m - 1)$, where n is the number of degrees of freedom of the moving platform and m is the number of active cables. The inequality holding for k reflects the practical need that the maximum number of equality constraints cannot exceed $n - 1$ (being 1 the force or torque to be maximized and n the overall number of wrench components) or $m - 1$ in the case of underactuated robots. The maximum number of rows l of matrix B is instead equal to $n - 1 - k$: since one inequality constraint can be introduced only if it is impossible, or not necessary for the given application, to provide $n - 1$ constraints in equality form (i.e., if $k < n - 1$, then l can be greater than zero). As for the number of columns of A and B , they are both equal to $m + 1$, that is, the size of the column vector $\{\tau^T \ 1\}^T$.

In conclusion, the WEC definition is strictly related to the constraints defined for the linear programming problem. Such constraints depend on the topology and on the application. It is important to notice that the formulation with inequality constraints can be always adopted, while exact values to all the wrench components apart from the one maximized (i.e., a formulation with a full set of $n - 1$ equality constraints) can be imposed only with fully actuated or redundant robot.

3. Results of WEC Application

The WEC computation presented in the previous section is here applied to the performance analysis of different topologies of cable robots. The objective is to provide a clear proof of the effectiveness and usefulness of the WEC when it is employed at the design stage to find the regions of the workspace where a cable robot can best perform, or to compare different cable robot topologies or cable layouts. Three representative investigations are carried out, addressing:

1. performance changes when robots have identical topologies but different cable layouts;
2. performance changes when a cable robot topology is altered by increasing the number of active cables;
3. performance evaluation of a spatial redundant cable robot.

3.1. Analysis of Cable Robots with Different Cable Layout

The first investigation concerns the group of cable robots shown in Figure 2 which comprises two redundant, planar and fully constrained cable robots with three dofs. The topology of the two robots

analysed is identical but the points where cables are attached to the moving platform are different. Therefore, it is said that the cable layouts are different. As shown in Figure 2, the moving platform of the robots is rectangular (0.4 m × 0.2 m) and is driven by four cables attached to the four platform vertices. The cable output points are located at the four vertices of a square. The coordinates (x, y) of the cable output points, expressed in a reference frame located at the square centroid, are: A(−1 m, −1 m), B(1 m, −1 m), C(1 m, 1 m) and D(−1 m, 1 m). It is assumed that the platform moves in a horizontal plane (plane xy) and that the overall design of the platform the drive pulleys and the winches allow avoiding cable interference in the robot with crossed cables (Figure 2, on the right). For both the robots, the WEC can be computed by employing the formulations proposed in Equations (4) and (5), since both the robots are redundant and the platform can be fully constrained. In this example, it has been chosen to impose null wrench components ($\tilde{w}_R = 0$) in the directions orthogonal to the one along which a force or torque is maximized. Additionally, since no external wrench is assumed to be applied on the platform, w_e is a null vector too.

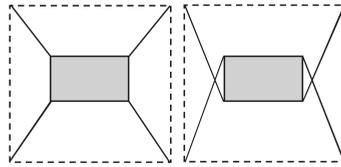


Figure 2. Two cable robots with same topology and different cable layout.

This choice is mainly meant to simplify the comparative analysis of the results. The maximization of all the three wrench components along the Cartesian axes x, y, z has been investigated, that is, two forces acting along the positive directions of the axes x and y in the plane of motion and a torque about the positive direction of axis z . For example, the WEC, in terms of maximum torque about the positive direction of axis z , has been computed as follows:

$$WEC_z^t := \max \left(w_{tz} = w_{tz}^T \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \right) \text{ s.t. : } \begin{cases} \begin{bmatrix} w_{fx}^T \\ w_{fy}^T \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = A \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \\ \tau_{min} \preceq \tau \preceq \tau_{max} \end{cases} \quad (8)$$

WEC_x^f and WEC_y^f , with the obvious meaning for symbols, have been computed similarly, always starting from the general formulation in Equation (7). Figure 3 collects all the results achieved. In Figure 3a, the sketches of the two cable robots can be recognized: dash-dotted lines are employed to connect the four cable output points; cables are represented by blue lines and the moving platform is depicted in solid black line. A green quadrangle delimits the Static Equilibrium Workspace (SEW) defined as the set of poses of the moving platform for which static equilibrium can be obtained while maintaining positive tensions in all the cables. The SEW has been computed with the shown orientation of the platform, that is, with the sides parallel to the x and y axes. As an example, the WEC has been evaluated at point $P(-0.2 \text{ m}, 0.2 \text{ m})$ and refers to the force exertion capability along the positive direction of axis x (see the red arrows and the red numbers). As far as the range of tensions that can be resisted by the cables is concerned, without loss of generality, the maximum value has been set to 100 N while the minimum to 5 N. The arrows in bold line overlapped to the cables provide a scale representation of the cable forces which allow achieving the maximum force represented by the red arrow, whose module is the WEC_x^f value, in N, computed at point P , which is also written below the red arrow.

While the results shown in Figure 3a refer to a single point P , the other plots of Figure 3 extend the analysis to the whole SEW: WEC values have been computed only for the points belonging to the SEW and without altering the moving platform orientation. Figure 3b shows the values (in N), taken by the

WEC_x^f , while Figure 3c shows the WEC_y^f . Finally, Figure 3d addresses the torque exertion capabilities of the robots by showing the WEC_z^t (in Nm). In all the subplots from Figure 3b–d WEC values are represented by isolines. The regions where the best performances are achieved can be immediately recognized: they are the ones where the isolines take the highest values (red lines). The comparison of the WEC plots clearly highlights the superior performances that can be guaranteed by the robot with crossed cables. First of all, such a robot, which has a wider SEW too, guarantees the possibility of exerting high forces in the x and y directions in a wider subset of the SEW (notice the extension of the red and orange isolines in subplots (b) and (c)). In terms of very maximum values taken by the forces, there are no significant differences between the two robots. Conversely, the torque exertion capability is completely different (see subplot (d)): the behaviour of the robot with crossed cables is preferable since its cables can exert much higher torques.

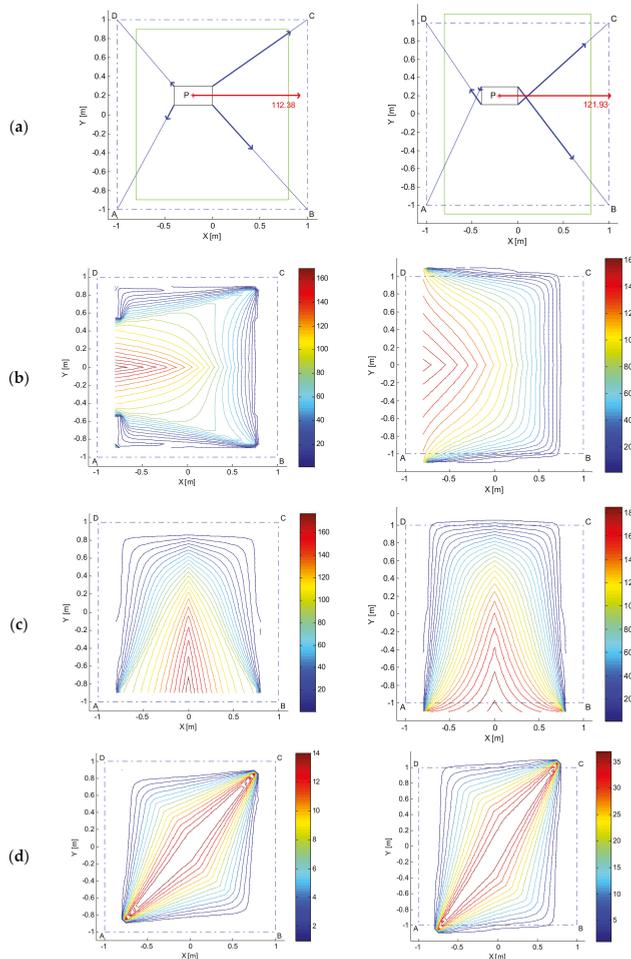


Figure 3. Wrench Exertion Capability (WEC) comparison for the two redundant cable robots sketched in Figure 2 in terms of analysis at point P (a), WEC_x^f (b), WEC_y^f (c) and WEC_z^t (d).

The WEC formulation can be further exploited to compute the minimum force values which can be guaranteed along any direction and at any point of the SEW and to perform an isotropy evaluation

based on comparing such minimum force values with the very maximum ones that can be exerted at any point of the SEW. As an example, such analyses are presented in Figure 4 with reference to the sole robot with crossed cables. Figure 4a represents such a robot at a generic point $P(-0.3\text{ m}, -0.3\text{ m})$.

The results of several repetitions of WEC_d^f computations at point P made considering different directions d are collected and depicted through a red polygon which provides a scale representation of the maximum force that can be exerted along any radial direction d around P . An angular resolution by 1° has been adopted to trace such a polygon, in other words, the polygon summarizes the output of 360 distinct WEC_d^f computations. The wrench exertion capabilities of the robot referred to all the possible directions can be immediately inferred. The maximum and minimum exertable forces can also be easily found: they are traced in green lines.

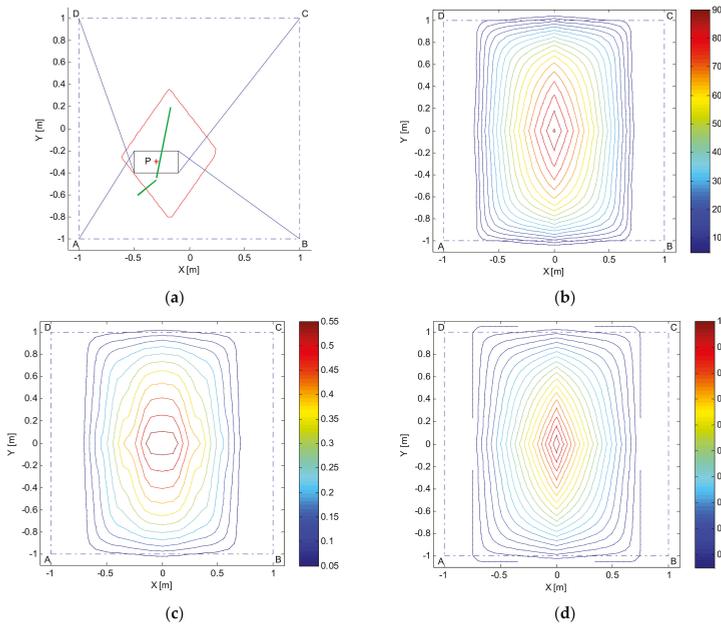


Figure 4. Scale representation of the maximum force that can be exerted along any direction at a point P (a); minimum guaranteed force (b); isotropy evaluation (c) and Tension Factor (d).

If the same WEC_d^f computation (i.e., repeated along any direction) is performed at each point of the SEW, a minimum force value which can be guaranteed irrespective of the direction can be found. Figure 4b shows such a value plotted through isolines. Getting such information at the design stage may obviously be of great practical usefulness.

If, at any point, the ratio between the minimum and maximum exertable forces (e.g., those traced in green lines for point P in the subplot (a)) is computed, isotropy can be evaluated effectively. Figure 4c shows such ratios, which can be straightforwardly compared to the popular isotropy index called Tension Factor (TF), proposed in [17]. The TF is the ratio between the minimum and the maximum cable tension values achieved when the platform is in static equilibrium. In Figure 4d the TF computed at each point of the SEW has been plotted by isolines. Apparently, subplots (c) and (d) provide different indications in terms of robot isotropy, however it is the authors' opinion that the one based on the WEC is more useful in practice, since the TF provides a measure of robot isotropy in the joint space, rather than in the Cartesian space.

3.2. Analysis of Cable Robots with Different Number of Cables

The second investigation concerns the group of cable robots shown in Figure 5 which comprises four underconstrained (or “cable suspended”) 3-dof cable robots differing in the number of cables and/or in the cable layout. The robots are assumed to move in a vertical plane, hence, their platforms are under the influence of gravity, which is essential to maintain tension in the cables in static conditions. Hence, for this comparison, the plane of motion is plane xz , the vertical one. The moving platforms of all the robots are assumed to be identical and share the same shape and dimensions of the of the cable robots presented in Section 3.1. The cables are all attached to the platform upper vertices. The cables output points are instead located at the two upper vertices of a square. The coordinates (x, z) of such points, expressed in a reference frame located at the square centroid, are: $A(-1\text{ m}, 1\text{ m})$ and $B(1\text{ m}, 1\text{ m})$. The first robot only has two cables and hence it is also underactuated. The other robots are instead fully actuated or redundant (i.e., with four cables).

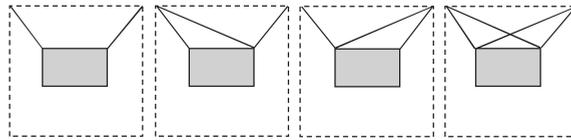


Figure 5. Cable suspended robots investigated comparatively by the WEC.

By comparing the performances of these robots, the effect of increasing the number of active cables can be appreciated, also in relation to the cable layout adopted. This is the objective of such a comparative analysis, whose results are collected in Figure 6. Robot performances are compared by referring to the WEC_x^f obtained by assuming that:

- cable tensions are to be kept in the range 5–100 N;
- a null overall torque must be exerted on the platform;
- a limited vertical force in the upwards direction must be applied to the platform. The lower and upper bounds of such a vertical force have been set equal to, respectively, 0 and 5 N.

The mentioned constraints are coherent with the field of application where cable suspended robots are very likely to be employed in the future: high speed pick and place manipulations (e.g., over-the-belt packaging). Indeed, if a cable suspended robot has to be employed for such tasks, it is of apparent interest evaluating which is the maximum horizontal force that can be exerted on the platform and hence applied to the picked object, while keeping a null torque on it (not to induce rotations) and a bounded upward force (if a given force value in the vertical direction cannot be set due to the limited number of cables available). Not only does imposing an upward force meet the basic requirement of lifting the picked objects during the manipulation (a lower bound higher than zero could be imposed to this purpose) but setting an upper bound could allow preventing the load from being dropped at the start of the motion.

In other words, the upper bound on the vertical force could reflect a limitation to the maximum vertical acceleration. In the comparative analysis, although the sole 2-cable robot is actually underactuated, identical constraints in the form of inequalities have been set for all WEC computations to make the comparison between different cable robot topologies fair. The problem is therefore stated as follows for all the robots:

$$WEC_x^f := \max \left(w_{f_x} = w_{f_x}^T \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \right) \quad s.t. : \begin{cases} \begin{bmatrix} w_{t_y}^T \\ \tau \\ 1 \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = A \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = 0 \\ \begin{bmatrix} w_{f_z}^T \\ -w_{f_z}^T \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = \begin{bmatrix} B \\ -B \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \preceq \tilde{w}_B \\ \tau_{min} \preceq \tau \preceq \tau_{max} \end{cases} \quad (9)$$

Generally speaking, the WEC_x^f can be computed at any point achievable by the moving platform statically or dynamically. In the subplots on the left of Figure 6, the computation is referred to a generic point $P(-0.2\text{ m}, 0.2\text{ m})$. The lines employed in these plots have the same meaning of the corresponding ones in Figure 4a. Here, however, a vertical green arrow is also adopted to provide a scale representation of the external wrench w_e : the force of gravity acting on the platform. The mass of the platform has been set equal to 5 kg.

In the subplots on the right of Figure 6, the analysis has been extended to all the points of the Statically Feasible Workspace (SFW) defined as the set of the mobile platform poses for which static equilibrium against gravity can be obtained using a limited range of cable tensions [3].

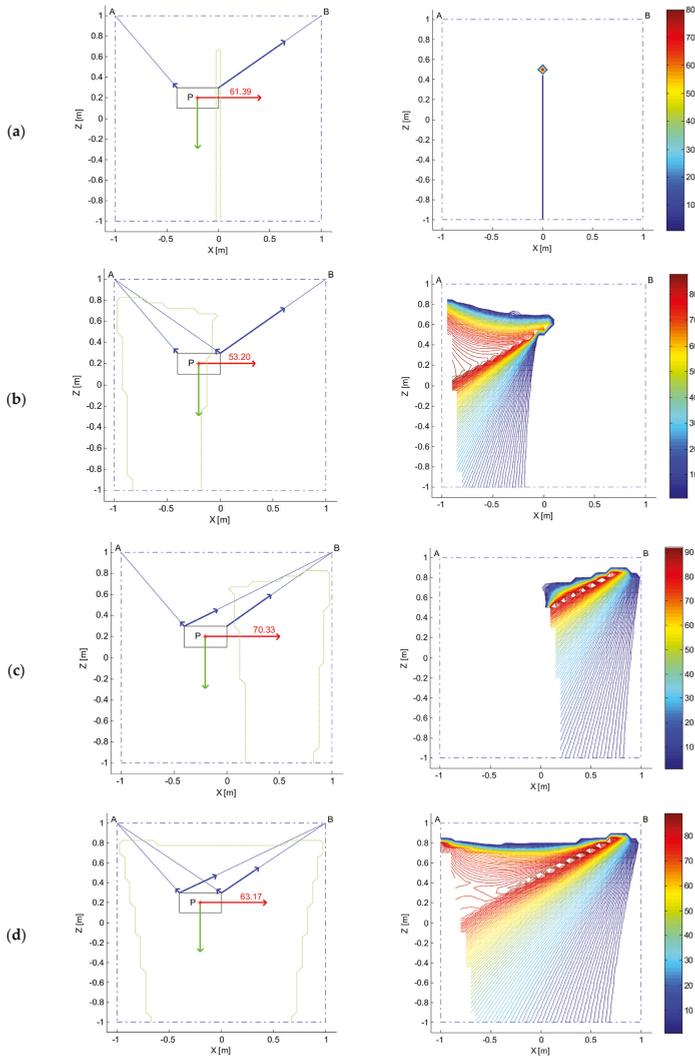


Figure 6. WEC_x^f (a–d) for each of the different topologies of cable suspended robots presented in Figure 5. WEC_x^f are either computed at a single point P (plots on the left) or throughout the statically feasible workspace (SFW) (plots on the right).

The SFWs of all the robots, computed keeping the platform horizontal, are represented in the subplots on the left of Figure 6, for clarity, delimited by green solid lines. The SFWs have been geometrically bounded by the square box with vertices at A and B . The isolines in the plots highlight the greatly different behaviors of the four cable driven robots: while for the underactuated robot (subplot (a)) it is possible to find a solution to the problem stated in Equation (9) in just one point of the SFW, by increasing the number of active cables, the force exertion capabilities improve considerably. The subplots (b) and (c) prove that, for the given problem of maximizing a rightward force, the two fully actuated robots behave very differently in their SFWs. The SFWs of the robots are very different (basically symmetrical about the z axis) and not overlapped, which complicates performing a straightforward comparison between the robots. Nonetheless, the plots provide clear hints about the regions where these robots can best perform. Clearly, the most effective cable layout could be identified once the geometrical features of the tasks to be executed and of the workcell were known: in general, the cable arrangement of Figure 6b seems preferable since the WEC_x^f takes high values in a wider region of the workspace. Finally, in Figure 6d, the WEC_x^f of the redundant robot is plotted. Obviously, the availability of a fourth cable allows extending the SFW and improving the performances within it. In particular, the rightward force exertion capability shows that this robot merges the benefits of the fully actuated robots discussed earlier, at the expense of an increased cost, design complexity and cable obstruction in the workspace.

3.3. Analysis of a 6-dof Overconstrained Spatial Cable Robot

The third investigation concerns the cable robot shown in Figure 7 which is a spatial redundant cable robot with six degrees of freedom. The robot design and cable layout recall those of several prototypes developed worldwide [1,2]. The moving platform of the robot is a parallelepiped ($0.4 \text{ m} \times 0.1 \text{ m} \times 0.2 \text{ m}$) that weighs 0.5 kg and is driven by eight cables attached to the eight platform vertices. The cable output points are located at the eight vertices of a cube. The coordinates (x, y, z) of the cable output points, expressed in a reference frame located at the cube centroid, are: $A(-1 \text{ m}, -1 \text{ m}, -1 \text{ m})$, $B(1 \text{ m}, -1 \text{ m}, -1 \text{ m})$, $C(1 \text{ m}, 1 \text{ m}, -1 \text{ m})$, $D(-1 \text{ m}, 1 \text{ m}, -1 \text{ m})$, $E(-1 \text{ m}, -1 \text{ m}, 1 \text{ m})$, $F(1 \text{ m}, -1 \text{ m}, 1 \text{ m})$, $G(1 \text{ m}, 1 \text{ m}, 1 \text{ m})$ and $H(-1 \text{ m}, 1 \text{ m}, 1 \text{ m})$.

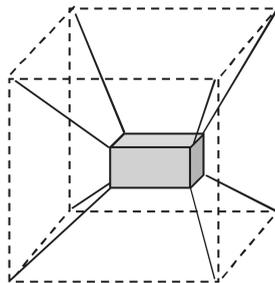


Figure 7. The 6-dof overconstrained spatial cable robot investigated.

For this robot, the WEC can be computed by employing the formulations proposed in Equations (4) and (5), since the robot is redundant and the platform can be fully constrained. Also in this case, it has been chosen to impose null wrench components ($\tilde{w}_R = 0$) in the directions orthogonal to the one along which a force or torque is maximized. Additionally, since gravity force is applied to the platform, the third row of the w_e vector is equal to the gravity force, while the other rows contain null values. The maximization of three representative wrench components along the Cartesian axes x, y, z has been investigated. In particular, it has been chosen to maximize the two forces acting along, respectively, the negative direction of the y axis and the positive direction of z axis. Finally, the torque acting about the negative direction of the axis y has been analysed. As an example, the WEC, in terms of maximum torque about the negative direction of axis y , has been computed as follows:

$$WEC_{-y}^t := \max \left(w_{t-y} = w_{t-y}^T \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} \right) \text{ s.t. : } \begin{cases} \begin{bmatrix} w_{f-y}^T \\ w_{fO1}^T \\ w_{fO2}^T \\ w_{tO1}^T \\ w_{tO2}^T \end{bmatrix} \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = A \begin{Bmatrix} \tau \\ 1 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \\ \tau_{min} \preceq \tau \preceq \tau_{max} \end{cases} \quad (10)$$

where the direction d is achieved by setting the matrix R defined in Equation (3) equal to an elementary rotation matrix of -90 degrees about the z axis ($R_z(-90)$).

Figure 8 collects the results achieved. For the sake of clarity, in the left subplots of Figure 8, the origin of the rotated reference frame is set at point $(0 \text{ m}, 0 \text{ m}, -1 \text{ m})$ and the direction d is highlighted by means of a red arrow. The results shown in the subplots on the left of Figure 8 refer to a single point P . Dash-dotted lines are employed to connect the eight cable output points and hence to represent the robot typical workspace; cables are represented by light-blue lines and the moving platform is depicted in solid black line. As an example, the WEC has been evaluated at point $P(0.2 \text{ m}, 0.4 \text{ m}, 0.0 \text{ m})$. As far as the range of tensions that can be resisted by the cables is concerned, without loss of generality, the maximum value has been set to 10 N while the minimum to 0.5 N . The arrows in bold blue line overlapped to the cables provide a scale representation of the cable forces which allow achieving the maximum force/torque represented by the red (in case of forces) or green (in case of torque) arrows. In the subplots on the right, each analysis is extended to the whole SEW: only the results achieved in some representative horizontal planes of the SEW are represented for clarity. As usual, in these subplots WEC values are represented by isolines, the red ones identifying the regions where the best performances are achieved (i.e., where the highest WEC values are achieved). Figure 8a shows the values (in N), taken by the WEC_{-y}^f , while Figure 8b shows the WEC_z^f (N). Finally, Figure 8c addresses the torque exertion capabilities of the robot by showing the WEC_{-y}^t (Nm).

As it has already stated for planar cable robots, the computed forces or torques can be exerted statically by the end-effector on the environment or alternatively they can be interpreted as forces or torques that the cables can apply to the end-effector to accelerate it. According to the latter interpretation, the results plotted give immediate evidence of the workspace locations where the highest dynamic performances can be achieved.

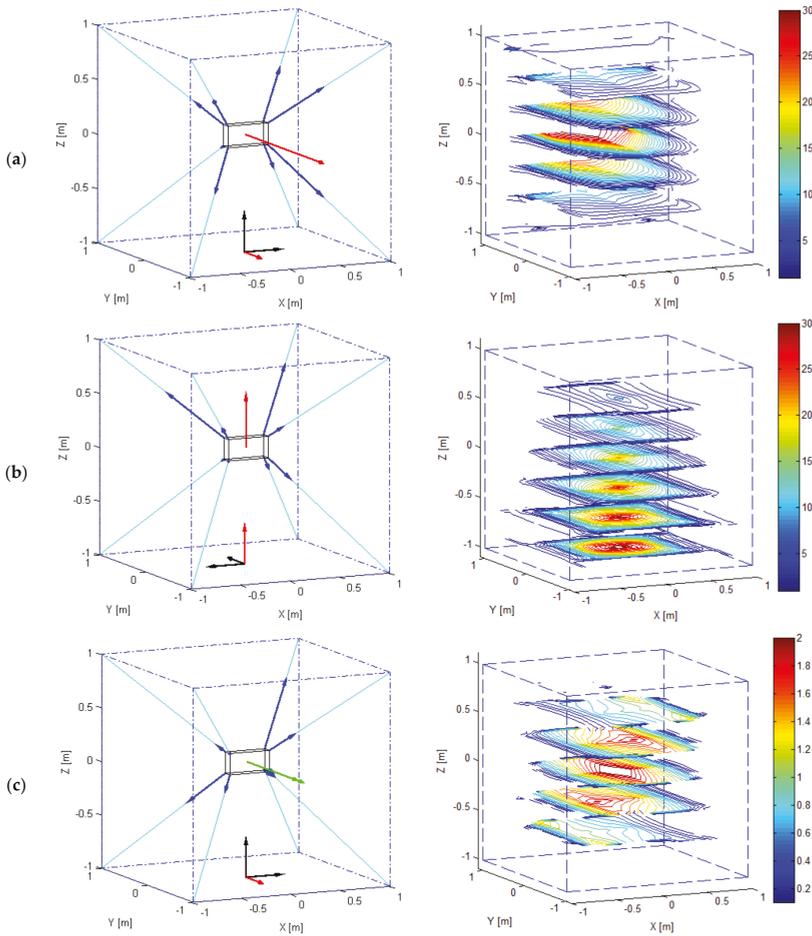


Figure 8. WEC_{-y}^f (a) WEC_z^f (b) and WEC_{-y}^t (c) of the spatial cable robot presented in Figure 7. WEC are either computed at a single point P (plots on the left) or throughout the static equilibrium workspace (SEW) (plots on the right).

4. Conclusions

A comprehensive evaluation of the performances of any cable driven robot can be carried by the proposed performance index named Wrench Exertion Capability (WEC), which allows evaluating the maximum force or torque a cable robot can exert along a direction of interest. The WEC accounts explicitly for the intrinsic cable tension limits and for the constraints which can be imposed to the wrench components that are not maximized.

A linear programming problem is solved to compute the WEC . The problem makes use of suitable partitions of a novel definition of the wrench matrix which has been introduced to simplify the inclusion of external wrenches in the analysis. The WEC formulation proposed is general enough to allow analyzing redundant, fully actuated and underactuated spatial cable robots. To this purpose, the use of constraints in both the form of equalities and inequalities has been suggested and discussed.

Finally, by a set of representative examples, it has been proved that the WEC can be adopted to carry out complete and effective evaluations of the performances of cable robots either in absolute or comparative terms.

Author Contributions: G.B. and A.T. conceived the study and designed the experiments; G.B. performed the experiments; G.B. and A.T. analyzed the data; G.B. and A.T. wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Albus, J.S.; Bostelman, R.; Dagalakis, N.G. The NIST ROBOCRANE. *J. Robot. Syst.* **1993**, *10*, 709–724. [[CrossRef](#)]
2. Pott, A.; Mütterich, H.; Kraus, W.; Schmidt, V.; Miermeister, P.; Verl, A. IPAnema: A family of Cable-Driven Parallel Robots for Industrial Applications. In *Cable-Driven Parallel Robots*; Bruckmann, T., Pott, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 119–134.
3. Trevisani, A. Planning of dynamically feasible trajectories for translational, planar, and underconstrained cable-driven robots. *J. Syst. Sci. Complex.* **2013**, *26*, 695–717. [[CrossRef](#)]
4. Abdolshah, S.; Zanutto, D.; Rosati, G.; Agrawal, S. Performance evaluation of a new design of cable-suspended camera system. In Proceedings of the International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 3728–3733.
5. Trevisani, A. Underconstrained planar cable-direct-driven robots: A trajectory planning method ensuring positive and bounded cable tensions. *Mechatronics* **2010**, *20*, 113–127. [[CrossRef](#)]
6. Carricato, M.; Abbasnejad, G. Direct geometrico-static analysis of under-constrained cable-driven parallel robots with 4 cables. In *Cable-Driven Parallel*; Bruckmann, T., Pott, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 269–285.
7. Zi, B.; Sun, H.; Zhang, D. Design, analysis and control of a winding hybrid-driven cable parallel manipulator. *Robot. Comput. Integr. Manuf.* **2017**, *48*, 196–208. [[CrossRef](#)]
8. Williams, R.L.; Gallina, P.; Vadia, J. Planar translational cable direct driven robots. *J. Robot. Syst.* **2003**, *20*, 107–120. [[CrossRef](#)]
9. Rosati, G.; Secoli, R.; Zanutto, D.; Rossi, A.; Boschetti, G. Planar robotic systems for upper-limb post-stroke rehabilitation. In Proceedings of the ASME 2008 International Mechanical Engineering Congress and Exposition (IMECE 2008), Boston, MA, USA, 31 October–6 November 2008.
10. Hernandez, E.; Valdez, S.I.; Carbone, G.; Ceccarelli, M. Design optimization of a cable-driven parallel robot in upper arm training-rehabilitation processes. *Mech. Mach. Sci.* **2018**, *54*, 413–424.
11. Surdilovic, D.; Radojicic, J.; Krüger, J. Geometric stiffness analysis of wire robots: A mechanical approach. In *Cable-Driven Parallel Robots*; Bruckmann, T., Pott, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 389–404.
12. Abdolshah, S.; Shojaei Barjuei, E. Linear quadratic optimal controller for cable-driven parallel robots. *Front. Mech. Eng.* **2015**, *10*, 344–351. [[CrossRef](#)]
13. Merlet, J.P. Jacobian, manipulability, condition number, and accuracy of parallel robots. *ASME J. Mech. Des.* **2006**, *128*, 199–206. [[CrossRef](#)]
14. Gao, F.; Liu, X.; Gruver, W.A. Performance Evaluation of Two-degree-of-freedom Planar Parallel Robots. *Mech. Mach. Theory* **1998**, *33*, 661–668. [[CrossRef](#)]
15. La Mura, F.; Romanò, P.; Fiore, E.; Giberti, H. Workspace Limiting Strategy for 6 DOF Force Controlled PKMs Manipulating High inertia Objects. *Robotics* **2018**, *7*, 10. [[CrossRef](#)]
16. Seriani, S.; Gallina, P.; Gasparetto, A. A performance Index for Planar Repetitive Workspace Robots. *J. Mech. Robot.* **2014**, *6*, 031005. [[CrossRef](#)]
17. Rosati, G.; Gallina, P. Manipulability of a planar wire driven active design. *Mech. Mach. Theory* **2002**, *37*, 215–228.
18. Pusey, J.; Fattah, A.; Agrawal, S.; Messina, E. Design and workspace analysis of a 6-6 cable suspended parallel robot. *Mech. Mach. Theory* **2004**, *39*, 761–778. [[CrossRef](#)]

19. Khalilpour, S.A.; Lololei, A.Z.; Taghirad, H.D.; Masouleh, M.T. Feasible Kinematic Sensitivity in Cable Robots Based on Interval Analysis. In *Cable Driven Parallel Robots*; Bruckmann, T., Pott, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 223–249.
20. Cardou, P.; Bouchard, S.; Gosselin, C. Kinematic-Sensitivity Indices for Dimensionally Nonhomogeneous Jacobian Matrices. *IEEE Trans. Robot.* **2010**, *26*, 166–173. [[CrossRef](#)]
21. Pham, C.B.; Yeo, S.H.; Yang, G.; Chen, I.-M. Workspace analysis of fully restrained cable-driven manipulators. *Robot. Auton. Syst.* **2009**, *57*, 901–912. [[CrossRef](#)]
22. Barbazza, L.; Oscari, F.; Minto, S.; Rosati, G. Trajectory planning of a suspended cable driven parallel robot with reconfigurable end effector. *Robot. Comput. Integr. Manuf.* **2017**, *48*, 1–11. [[CrossRef](#)]
23. Abdolshas, S.; Zanotto, D.; Rosati, G.; Agrawal, S.K. Optimizing Stiffness and Dexterity of Planar Adaptive Cable-Driven Parallel Robots. *ASME J. Mech. Robot.* **2017**, *9*, 031004:1–031004:11.
24. Boschetti, G.; Trevisani, A. Performance evaluation for cable direct driven robot. In Proceedings of the 12th Biennial Conference on Engineering Systems Design and Analysis ESDA 2014, Copenhagen, Denmark, 251–27 June 2014.
25. Boschetti, G.; Trevisani, A. On the Use of the Wrench Exertion Capability as a Performance Index for Cable Driven Robot. In Proceedings of the Thematic Conference on Multibody Dynamics ECCOMAS 2015, Barcelona, Spain, 29 June–2 July 2015.
26. Berti, A.; Gouttefarde, M.; Carricato, M. Dynamic recovery of cable-suspended parallel robots after a cable failure. In *Advances in Robot Kinematics 2016*; Lenarčič, J., Merlet, J.P., Eds.; Springer: Cham, Switzerland, 2018; Volume 4, pp. 331–339.
27. Boschetti, G.; Passarini, C.; Trevisani, A. A recovery strategy for cable driven robots in case of cable failure. *Int. J. Mech. Control* **2017**, *18*, 41–48.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Optimal Kinematic Design of a 6-UCU Kind Gough-Stewart Platform with a Guaranteed Given Accuracy

Guojun Liu

School of Mechanical Engineering, Hunan Institute of Science and Technology, Yueyang 414006, China; liuguojun_ist@163.com; Tel.: +86-138-7309-3067

Received: 24 April 2018; Accepted: 15 June 2018; Published: 18 June 2018

Abstract: The 6-UCU (U-universal joint; C-cylinder joint) kind Gough-Stewart platform is extensively employed in motion simulators due to its high accuracy, large payload, and high-speed capability. However, because of the manufacturing and assembling errors, the real geometry may be different from the nominal one. In the design process of the high-accuracy Gough-Stewart platform, one needs to consider these errors. The purpose of this paper is to propose an optimal design method for the 6-UCU kind Gough-Stewart platform with a guaranteed given accuracy. Accuracy analysis of the 6-UCU kind Gough-Stewart platform is presented by considering the limb length errors and joint position errors. An optimal design method is proposed by using a multi-objective evolutionary algorithm, the non-dominated sorting genetic algorithm II (NSGA-II). A set of Pareto-optimal parameters was found by applying the proposed optimal design method. An engineering design case was studied to verify the effectiveness of the proposed method.

Keywords: accuracy analysis; optimal design; multi-objective evolutionary algorithms; NSGA-II; 6-UCU kind Gough-Stewart platform

1. Introduction

With the advantages of high rigidity, high precision, and large carrying capacity, Gough-Stewart platforms (GSPs) are extensively used in virtual-reality motion simulators [1]. Many scholars studied the optimal design of the 6-UPS kind GSP [1], where U stands for the universal joint, P for the prismatic joint, and S for the spherical joint. When compared with spherical joints, universal joints can bear more tension [2]. Universal joints are extensively used as the passive joints of GSPs to connect hydraulic cylinders or electric cylinders to the moving platform and fixed base. Universal joints are used as the passive joints of the universal tyre test machine designed by Gough et al. [3–5], the motion simulator patented by Cappel [6], commercial flight simulators [7], the Ampelmann system [8], the Moog FCS 5000E motion base [9], the VARIAX machine tool [10], and the AMiBA hexapod telescope mount [11]. Universal joints are also used as the passive joints of the docking test system, and about 60 motion simulators designed by the team of Professor Junwei Han at the Harbin Institute of Technology, China. The cylinders of the hydraulic and electric actuators that can not only translate along the axis, but also rotate along the axis are cylindrical joints instead of prismatic ones. These GSPs are 6-UCU parallel manipulators [12,13], where C stands for the cylinder joint.

In highly accurate positioning applications, such as the docking test systems, high accuracy is required to achieve good simulation results. In China, the translational motion errors of the moving platform must be less than 1 mm in the whole workspace of the docking test system [14]. Many researchers studied the accuracy analysis of GSPs. Wang and Masory studied the effects of manufacturing errors on the accuracy of a GSP by modeling the GSP as serial legs based on the Denavit–Hartenberg method [15]. Ropponen and Arai presented the error model of a modified GSP

by using the differentiation method [16]. Patel and Ehmann established an error model of a GSP, addressing all possible sources of errors [17]. Wang and Ehmann developed first- and second-order error models of a GSP [18]. Masory et al. presented an effective method of identifying kinematic parameters of a GSP using pose measurements [19]. Cong et al. developed a kinematic calibration method of a GSP by using a three-dimensional coordinate measuring machine containing actuator errors and passive joint errors [20]. Dai et al. proposed an accuracy analysis method for a docking test system [14]. Merlet and Daney proposed an optimal method of parallel manipulator considering manufacturing errors by using interval analysis [21–23]. Because of the dependence of the variables, a high level of expertise in interval analysis is needed for the proper usage of parallel manipulators [22]. To the best of our knowledge, there are few papers considering the errors of a GSP in optimal design processes.

In the optimal design process of a high-precision GSP, one needs to find the optimal solutions which meet the given accuracy requirement. An optimal design method for a GSP is proposed in this paper, so as to meet these accuracy requirements.

In this paper, the error model of the 6-UCU kind GSP is derived in Section 2. The optimal design method is proposed in Section 3. Section 4 presents a case study to illustrate the effectiveness of the proposed optimal design. The conclusions are given in Section 5.

2. Differential Error Model

In this section, the relationship between the structural errors and end-point errors of a 6-UCU kind GSP is discussed. The 6-UCU kind GSP consisted of a moving platform, a fixed base, and six limbs, as shown in Figures 1 and 2. The i th limb was connected to the fixed base via a universal joint, whose joint center point is denoted as B_i , and connected to the mobile platform via a universal joint with center P_i , as shown in Figure 2. For convenience, a Cartesian coordinate frame, $O_1 - X_1Y_1Z_1$, was attached to the moving platform with origin O_1 . A Cartesian coordinate frame, $O - XYZ$, was attached to the fixed base with origin O .

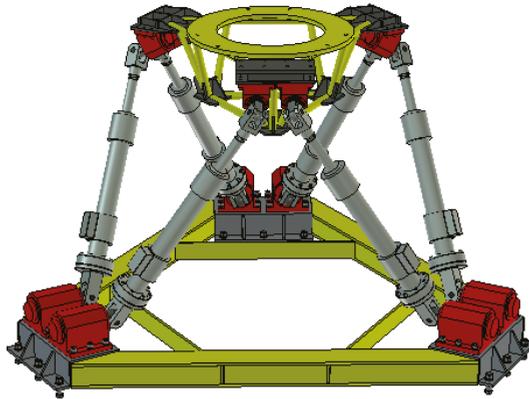


Figure 1. The 6-UCU (U—universal joint; C—cylinder joint) kind Gough-Stewart platform (GSP).

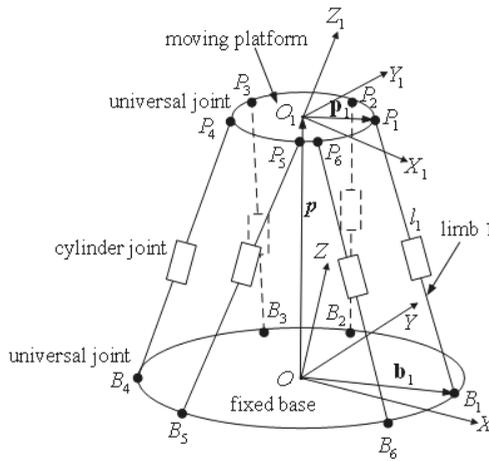


Figure 2. Schematic of the 6-UCU kind GSP.

The positions of the universal joints attached to the moving platform and the fixed base of the GSP formed semi-hexagons, as shown in Figure 3, where r_p is the radius of the platform-joint attachment circle, d_p is the distance between the shorter edges of the attachment point semi-hexagon on the moving platform, r_b is the radius of the base-joint attachment circle, and d_b is the distance between the shorter edges of the attachment point semi-hexagon on the fixed base.

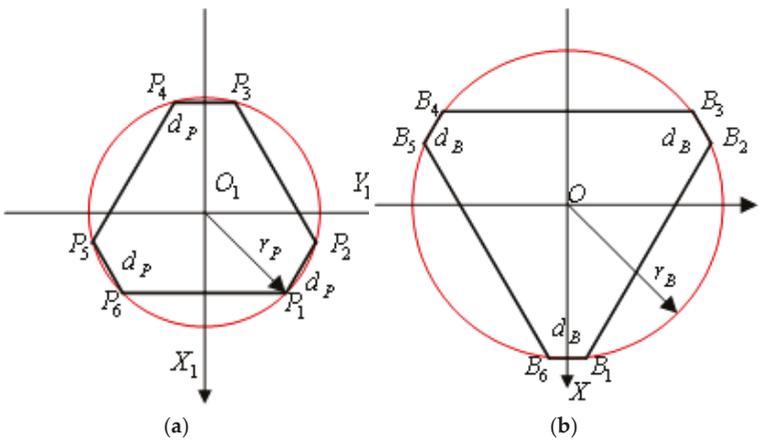


Figure 3. Schematic of the universal joint positions. (a) Moving platform; and (b) fixed base.

Referring to Figures 1 and 2, the position vector of P_i expressed in the frame $O - XYZ$ could be written as

$$\mathbf{p} + \mathbf{R}^L \mathbf{p}_i = \mathbf{b}_i + l_{1i} \mathbf{n}_{1i} + l_{2i} \mathbf{n}_{2i}, \quad (1)$$

where \mathbf{R} is the rotation matrix from the frame $O_1 - X_1 Y_1 Z_1$ to $O - XYZ$, \mathbf{p} is the position coordinate of the point O_1 measured in the frame $O - XYZ$, \mathbf{b}_i is the position coordinate of the point B_i measured in the frame $O - XYZ$, \mathbf{p}_i is expressed as ${}^L \mathbf{p}_i$ in the frame $O - XYZ$, ${}^L \mathbf{p}_i$ is the coordinate of the point P_i measured in the frame $O_1 - X_1 Y_1 Z_1$, l_{1i} is the distance from the point B_i to the lower plane of the

piston, l_{2i} is the distance from the point P_i to the lower plane of the piston, \mathbf{n}_i is the unit vector of $\vec{B_iP_i}$, \mathbf{n}_{1i} and \mathbf{n}_{2i} are the unit vectors of \mathbf{n}_i fixed to the cylinder and piston, respectively, and $\mathbf{n}_i = \mathbf{n}_{1i} = \mathbf{n}_{2i}$. ${}^L\mathbf{p}_i$ and \mathbf{b}_i are

$${}^L\mathbf{p}_i = \begin{bmatrix} r_P \cos \theta_{Pi} & r_P \sin \theta_{Pi} & h_P \end{bmatrix}^T, \tag{2}$$

$$\mathbf{b}_i = \begin{bmatrix} r_B \cos \theta_{Bi} & r_B \sin \theta_{Bi} & h_B \end{bmatrix}^T, \tag{3}$$

where T is the matrix transpose, h_P is the z coordinate of the upper universal joint of the i th limb in the frame $O_1 - X_1Y_1Z_1$, h_B is the z coordinate of the lower universal joint of the i th limb in the frame $O - XYZ$, and θ_{Pi} and θ_{Bi} are as follows:

$$\begin{aligned} \boldsymbol{\theta}_P &= \begin{bmatrix} \theta_{P1} & \theta_{P2} & \theta_{P3} & \theta_{P4} & \theta_{P5} & \theta_{P6} \end{bmatrix}^T \\ &= \begin{bmatrix} \eta_A & \frac{2\pi}{3} - \eta_A & \frac{2\pi}{3} + \eta_A & \frac{4\pi}{3} - \eta_A & \frac{4\pi}{3} + \eta_A & -\eta_A \end{bmatrix}^T, \end{aligned} \tag{4}$$

$$\begin{aligned} \boldsymbol{\theta}_B &= \begin{bmatrix} \theta_{B1} & \theta_{B2} & \theta_{B3} & \theta_{B4} & \theta_{B5} & \theta_{B6} \end{bmatrix}^T \\ &= \begin{bmatrix} \eta_B & \frac{2\pi}{3} - \eta_B & \frac{2\pi}{3} + \eta_B & \frac{4\pi}{3} - \eta_B & \frac{4\pi}{3} + \eta_B & -\eta_B \end{bmatrix}^T, \end{aligned} \tag{5}$$

where

$$\eta_A = \frac{\pi}{3} - \arcsin\left(\frac{d_P}{2r_P}\right), \tag{6}$$

$$\eta_B = \arcsin\left(\frac{d_B}{2r_B}\right). \tag{7}$$

For the i th leg, the limb length l_i was derived as

$$l_i = l_{1i} + l_{2i}, \tag{8}$$

where l_i is the length of $\vec{B_iP_i}$.

Differentiating Equation (1), we obtained

$$\delta\mathbf{p} + \delta\mathbf{R}{}^L\mathbf{p}_i + \mathbf{R}\delta{}^L\mathbf{p}_i = \delta\mathbf{b}_i + \delta l_{1i}\mathbf{n}_{1i} + l_{1i}\delta\mathbf{n}_{1i} + \delta l_{2i}\mathbf{n}_{2i} + l_{2i}\delta\mathbf{n}_{2i}, \tag{9}$$

where $\delta\mathbf{R}$ could be rewritten as follows, according to Reference [16]:

$$\delta\mathbf{R} = \delta\boldsymbol{\theta} \times \mathbf{R} = \begin{bmatrix} 0 & -\delta\theta_Z & \delta\theta_Y \\ \delta\theta_Z & 0 & -\delta\theta_X \\ -\delta\theta_Y & \delta\theta_X & 0 \end{bmatrix} \mathbf{R} = \boldsymbol{\Omega}\mathbf{R}, \tag{10}$$

where $\delta\boldsymbol{\theta} = \begin{bmatrix} \delta\theta_X & \delta\theta_Y & \delta\theta_Z \end{bmatrix}^T$ is the orientation error vector of the moving platform in the frame $O - XYZ$, and $\delta\mathbf{p} = \begin{bmatrix} \delta p_X & \delta p_Y & \delta p_Z \end{bmatrix}^T$ is the translational error vector of the moving platform in the frame $O - XYZ$.

Multiplication of \mathbf{n}_i^T on both sides of Equation (10) yielded

$$\begin{aligned} \mathbf{n}_i^T \delta\mathbf{p} + \mathbf{n}_i^T \delta\boldsymbol{\theta} \times \mathbf{R}{}^L\mathbf{p}_i + \mathbf{n}_i^T \mathbf{R} \delta{}^L\mathbf{p}_i \\ = \mathbf{n}_i^T \delta\mathbf{b}_i + \mathbf{n}_i^T \delta l_{1i}\mathbf{n}_{1i} + \mathbf{n}_i^T l_{1i}\delta\mathbf{n}_{1i} + \mathbf{n}_i^T \delta l_{2i}\mathbf{n}_{2i} + \mathbf{n}_i^T l_{2i}\delta\mathbf{n}_{2i}. \end{aligned} \tag{11}$$

Since $\mathbf{n}_i^T \mathbf{n}_i = 1$, $\mathbf{n}_i = \mathbf{n}_{1i} = \mathbf{n}_{2i}$, $\delta l_{2i} = 0$, and $\delta l_i = \delta l_{1i}$, we had $\mathbf{n}_i^T \delta \mathbf{n}_i = 0$, $\mathbf{n}_i^T \delta \mathbf{n}_{1i} = 0$, and $\mathbf{n}_i^T \delta \mathbf{n}_{2i} = 0$. Therefore, Equation (11) could be rewritten as

$$\begin{aligned} & \mathbf{n}_i^T \delta \mathbf{p} + \mathbf{n}_i^T \delta \theta \times \mathbf{R}^L \mathbf{p}_i + \mathbf{n}_i^T \mathbf{R} \delta^L \mathbf{p}_i \\ &= \mathbf{n}_i^T \delta \mathbf{b}_i + \mathbf{n}_i^T \delta l_{1i} \mathbf{n}_i \\ &= \mathbf{n}_i^T \delta \mathbf{b}_i + \delta l_{1i} \\ &= \mathbf{n}_i^T \delta \mathbf{b}_i + \delta l_i \end{aligned} \quad (12)$$

δl_i could be derived as

$$\begin{aligned} \delta l_i &= \mathbf{n}_i^T \delta \mathbf{p} + \mathbf{n}_i^T \delta \theta \times \mathbf{R}^L \mathbf{p}_i + \mathbf{n}_i^T \mathbf{R} \delta^L \mathbf{p}_i - \mathbf{n}_i^T \delta \mathbf{b}_i \\ &= \begin{bmatrix} \mathbf{n}_i^T & (\mathbf{R}^L \mathbf{p}_i \times \mathbf{n}_i)^T \end{bmatrix} \begin{bmatrix} \delta \mathbf{p} \\ \delta \theta \end{bmatrix} + \\ & \quad \begin{bmatrix} \mathbf{n}_i^T \mathbf{R} & (-\mathbf{n}_i^T) \end{bmatrix} \begin{bmatrix} \delta^L \mathbf{p}_i \\ \delta \mathbf{b}_i \end{bmatrix} \end{aligned} \quad (13)$$

Once all six limbs were assembled, the errors could be expressed as

$$\delta \mathbf{l} = \mathbf{J}_P \delta \mathbf{x} + \mathbf{J}_s \delta \mathbf{s}, \quad (14)$$

where

$$\delta \mathbf{l} = \begin{bmatrix} \delta l_1 & \dots & \delta l_6 \end{bmatrix}^T \in \mathfrak{R}^{6 \times 1}, \quad (15)$$

$$\delta \mathbf{x} = \begin{bmatrix} \delta \mathbf{p} \\ \delta \theta \end{bmatrix} \in \mathfrak{R}^{6 \times 1}, \quad (16)$$

$$\delta \mathbf{s} = \begin{bmatrix} \delta^L \mathbf{p}_i \\ \delta \mathbf{b}_1 \\ \vdots \\ \delta^L \mathbf{p}_6 \\ \delta \mathbf{b}_6 \end{bmatrix} \in \mathfrak{R}^{36 \times 1}, \quad (17)$$

$$\mathbf{J}_P = \begin{bmatrix} \mathbf{n}_1^T & (\mathbf{R}^L \mathbf{p}_1 \times \mathbf{n}_1)^T \\ \vdots & \vdots \\ \mathbf{n}_6^T & (\mathbf{R}^L \mathbf{p}_6 \times \mathbf{n}_6)^T \end{bmatrix} \in \mathfrak{R}^{6 \times 6}, \quad (18)$$

$$\mathbf{J}_s = \begin{bmatrix} \mathbf{n}_1^T \mathbf{R} & (-\mathbf{n}_1^T) & \dots & \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 3} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 3} & \dots & \mathbf{n}_6^T \mathbf{R} & (-\mathbf{n}_6^T) \end{bmatrix} \in \mathfrak{R}^{6 \times 36}, \quad (19)$$

$$\mathbf{0}_{1 \times 3} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}. \quad (20)$$

If the inverse of \mathbf{J}_P existed, then $\delta \mathbf{x}$ could be derived as

$$\delta \mathbf{x} = \mathbf{J}_P^{-1} \delta \mathbf{l} - \mathbf{J}_P^{-1} \mathbf{J}_s \delta \mathbf{s}. \quad (21)$$

The first term on the right side represents the actuation-induced error, and the second term is the error caused by the position errors of the joints [16].

3. Optimal Design Method

In engineering, we may assume that the manufacturing tolerances on the geometrical parameters are bounded and the maximum values of $\delta \mathbf{l}$ and $\delta \mathbf{s}$ are known as a function of the manufacturing

method. The maximum values of δx for the optimal design solutions had to be lower than the given accuracy by customers, before being dealt with as a constraint in the optimal design process in this paper.

Many scholars studied the optimal design of a GSP with only one optimal solution [24]. In practice, many different functional requirements of robots are intended to be satisfied [25]; thus, it is more appropriate to have multiple optimization solutions after the optimal kinematic design [24]. The condition number and determinant of the kinematic Jacobian matrix, J_P , of a GSP were extensively used as optimal design objectives [24]; therefore, they were chosen as objective functions in the optimal design process in this paper. Accordingly, they were defined as

$$\text{cond}(J_P) = \frac{\sigma_{\max}(J_P)}{\sigma_{\min}(J_P)}, \tag{22}$$

$$\omega = \sqrt{|\det(J_P J_P^T)|} = |\det(J_P)|, \tag{23}$$

where $\sigma_{\max}(J_P)$ and $\sigma_{\min}(J_P)$ are the maximum and minimum singular values of J_P at one pose, respectively, and $\det(J_P)$ is the determinant of J_P .

In the optimal design process, evolutionary algorithms are extensively used to search the optimal solutions [26]. In this paper, the elitist non-dominated sorting genetic algorithm version II (NSGA-II), developed by Deb et al. [27], was employed to solve the multi-objective optimal design problems of the 6-UCU kind GSP, due to its good spread of solutions and convergence to obtain the Pareto front. Real-coded NSGA-II with a simulated binary crossover (SBX) operator [28] and a polynomial mutation operator [29] were adopted in this paper to search the minimum values of the objective functions. The parameters of the NSGA-II are shown in Table 1.

Table 1. Parameters of the non-dominated sorting genetic algorithm version II (NSGA-II).

	Number of Iterations	1000
	Population size	50
	Crossover probability	0.9
	Mutation probability	0.1
	Distribution index for the simulated binary crossover (SBX)	20
	Distribution index for the polynomial mutation	20

The optimal design procedures were proposed as described below.

Step 1. The customers usually provided the maximum motion requirements for each degree of freedom (DOF), as shown in Table 2.

Table 2. Typical requirements by customers.

Payload			
Degree of Freedom (DOF)	Maximum Excursion	Speed	Acceleration
Roll	$\pm Ro_X(^{\circ})$	$\pm Rv_X(^{\circ}/s)$	$\pm Ra_X(^{\circ}/s^2)$
Pitch	$\pm Ro_Y(^{\circ})$	$\pm Rv_Y(^{\circ}/s)$	$\pm Ra_Y(^{\circ}/s^2)$
Yaw	$\pm Ro_Z(^{\circ})$	$\pm Rv_Z(^{\circ}/s)$	$\pm Ra_Z(^{\circ}/s^2)$
Surge	$\pm Tr_X(m)$	$\pm Tv_X(m/s)$	$\pm Ta_X(m/s^2)$
Sway	$\pm Tr_Y(m)$	$\pm Tv_Y(m/s)$	$\pm Ta_Y(m/s^2)$
Heave	$\pm Tr_Z(m)$	$\pm Tv_Z(m/s)$	$\pm Ta_Z(m/s^2)$

We transformed the requirements of the customer to 12 typical trajectories along six single axes [24]. Two typical travel trajectories of translation along the X axis were

$$S_X(t) = Tr_X \sin\left(\frac{Tv_X t}{Tr_X}\right), \tag{24}$$

$$S_X(t) = \frac{(Tv_X)^2}{Ta_X} \sin\left(\frac{Ta_X t}{Tv_X}\right), \tag{25}$$

where t is the run time.

Step 2. At each pose, $cond(\mathbf{J}_P)$ was calculated using Equation (22). When it was at or near the singularity, calculating $\delta\mathbf{x}$ using Equation (21) was potentially wrong. We used another method to solve this problem, as follows: if $cond(\mathbf{J}_P) > 10^6$, then $\delta\mathbf{x} = \begin{bmatrix} 10^7 & 10^7 & 10^7 & 10^7 & 10^7 & 10^7 \end{bmatrix}^T$; if $cond(\mathbf{J}_P) \leq 10^6$, $\delta\mathbf{x}$ was calculated using Equation (21). In the searching process, the guaranteed accuracy was given, and then it was dealt with as a constraint. The penalty function is the most often used technique in constrained optimization [26]; however, the penalty coefficients are very difficult to choose appropriately [30]. We handled the constraint as follows: if $\delta\mathbf{x}$ was in the required range, $cond(\mathbf{J}_P)$ was calculated using Equation (22), and ω was calculated using Equation (23). Otherwise, $cond(\mathbf{J}_P) = (10^7 + f_c)$ and $\omega = (-f_c)$. A subprogram was subsequently created to calculate the maximum value of $cond(\mathbf{J}_P)$ and the minimum value of ω for all 12 typical trajectories.

$$f_c = \left\| \frac{\delta p_X}{ac_{TX}} \right\| + \left\| \frac{\delta p_Y}{ac_{TY}} \right\| + \left\| \frac{\delta p_Z}{ac_{TZ}} \right\| + \left\| \frac{\delta \theta_X}{ac_{RX}} \right\| + \left\| \frac{\delta \theta_Y}{ac_{RY}} \right\| + \left\| \frac{\delta \theta_Z}{ac_{RZ}} \right\|, \tag{26}$$

where ac_{TX} , ac_{TY} , and ac_{TZ} are the required maximum linear displacement positioning errors along the X, Y, and Z axes, respectively; ac_{RX} , ac_{RY} , and ac_{RZ} are the required maximum angular displacement positioning errors along the X, Y, and Z axes, respectively.

Step 3. $cond(\mathbf{J}_P)_{\max}$ was set as the maximum value of $cond(\mathbf{J}_P)$ calculated in step 2, and ω_{\min} was set as the minimum value of ω calculated in step 2.

Step 4. The design variables were chosen as r_p , r_B , d_p , d_B , and H_0 , where H_0 was the height of the upper universal joint plane to the lower universal joint plane at the home position. $f_1 = cond(\mathbf{J}_P)_{\max}$ and $f_2 = -\omega_{\min}$. NSGA-II was used to minimize the objective functions, f_1 and f_2 , simultaneously.

For other requirements, the 12 typical trajectories were replaced by all the typical trajectories in the special applications. In the optimal kinematic design of a flight simulator, 31 typical trajectories of a Boeing 747 were considered in the design process by Advani [31]. If one desires a singularity-free workspace, as for motion simulators, the 12 typical trajectories in Step 2 would be replaced by all the typical trajectories and the 64 extreme positions [24,31].

4. Case Study

A case study is presented in this section to show the effectiveness of the proposed optimal design method. For this specific case, the requirements of the customer are shown in Table 3.

From a practical consideration, the ranges of the parameters were chosen as follows:

$h_p = -0.45(\text{m})$, $h_B = (-0.45 - H_0)(\text{m})$, $0.75(\text{m}) \leq r_B \leq 3.0(\text{m})$, $0.75(\text{m}) \leq r_p \leq 3.0(\text{m})$, $r_p \leq r_B$, $1.0(\text{m}) \leq H_0 \leq 3.5(\text{m})$, $0.25(\text{m}) \leq d_B \leq r_B$, and $0.22(\text{m}) \leq d_p \leq r_p$. The range of every element of $\delta\mathbf{l}$ and $\delta\mathbf{s}$ were in the range $[-0.3, 0.3](\text{mm})$.

Table 3. Requirements of the customer.

Payload		10,000 kg	
Maximum Position Errors		Linear Travel	1 mm
		Angular Travel	0.1°
DOF	Maximum Excursion	Speed	Acceleration
Roll	±25°	±20°/s	±210°/s ²
Pitch	±25°	±20°/s	±210°/s ²
Yaw	±30°	±20°/s	±210°/s ²
Surge	±1 m	±0.7 m/s	±10 m/s ²
Sway	±1 m	±0.7 m/s	±10 m/s ²
Heave	±0.8 m	±0.6 m/s	±10 m/s ²

After applying the proposed optimization process, as outlined in Section 3, 50 optimal solutions and the Pareto-optimal front were found, as shown in Figure 4. Lastly, there were three optimum design points, labeled as a, b, and c, whose corresponding objective values and design parameters are shown in Table 4. As seen in Table 4, the maximum errors of all the linear displacements were lower than 1 mm, and the maximum errors of all the angular displacements were lower than 0.1°. Therefore, all the requirements by the customer, given in Table 3, were met. This showed the effectiveness of the proposed method for the optimal design purpose.

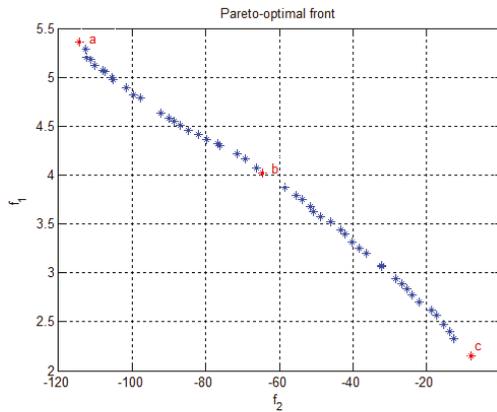


Figure 4. Pareto-optimal front of the solutions.

Table 4. Objective functions values from the Pareto sets and their corresponding design variables.

	a	b	c
d_B (m)	0.3409	0.3489	0.2500
d_P (m)	1.6915	0.5177	0.2200
r_B (m)	3.0000	2.9981	2.6552
r_P (m)	3.0000	2.4046	1.2128
H_0 (m)	2.4002	2.4044	2.4363
f_2	-114.2709	-64.5399	-8.0017
f_1	5.3615	4.0201	2.1464
δp_X (mm)	0.2	0.2	0.2
δp_Y (mm)	0.2	0.2	0.2
δp_Z (mm)	0.6	0.6	0.6
$\delta\theta_X$ (°)	0.0022	0.0025	0.0020
$\delta\theta_Y$ (°)	0.0018	0.0027	0.0026
$\delta\theta_Z$ (°)	0.0030	0.0029	0.0030

5. Conclusions

Motion simulators extensively use 6-UCU kind GSPs. A differential error model for a 6-UCU kind GSP was derived in this paper, and contained both the actuation-induced error and the error caused by the position errors of the joints. The guaranteed given accuracy was used as a constraint in the optimal design process of the high-accuracy motion simulators. An optimal kinematic design method was proposed by using the multi-objective evolutionary algorithm, NSGA-II. If J_p was near or at the singularity pose, the inverse of J_p potentially did not exist. δx could be wrongly calculated using Equation (21). In order to solve this problem, a method was presented where the condition number of J_p was compared with a large number. Another engineering method was proposed to handle the accuracy requirement constraints in the optimal design process. Multiple optimization solutions were found following implementation of the optimal kinematic design.

The effectiveness of the proposed method was verified through a practical optimal design case. The proposed optimal design method can be used as a guideline for the practical design of GSPs used in other applications with high-accuracy requirements.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Merlet, J.P. *Parallel Robots*, 2nd ed.; Springer: Dordrecht, The Netherlands, 2006; pp. 77–78. ISBN 978-1-4020-4132-7.
- Huang, Z.; Kong, L.F.; Fang, Y.F. *Mechanism Theory of Parallel Robotic Manipulator and Control*; China Mechanical Press: Beijing, China, 1997; pp. 306–308. ISBN 7-111-05812-7. (In Chinese)
- Fichter, E.; Kerr, D.; Rees-Jones, J. The Gough—Stewart platform parallel manipulator: A retrospective appreciation. *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* **2009**, *223*, 243–281. [[CrossRef](#)]
- Gough, V.E.; Whitehall, S.G. Universal Tyre Testing Machine. In Proceedings of the 9th International Automobile Technical Congress, Bursa, Turkey, 7–8 May 2018.
- Stewart, D. A Platform with six degrees of freedom. *Proc. Inst. Mech. Eng.* **1965**, *180*, 371–386. [[CrossRef](#)]
- Cappel, K.L. Motion Simulator. U.S. Patent 3,295,224, 3 January 1967.
- Ma, O. Mechanical Analysis of Parallel Manipulators with Simulation, Design and Control Applications. Ph.D. Thesis, McGill University, Montreal, QC, Canada, 1991.
- Cerda Salzman, D.J. Ampelmann: Development of the Access System for Offshore Wind Turbines. Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands, 2010.
- Blaise, J.; Bonev, I.; Monsarrat, B.; Briot, S.; Lambert, J.M.; Perron, C. Kinematic characterisation of hexapods for industry. *Ind. Robot Int. J.* **2010**, *37*, 79–88. [[CrossRef](#)]
- VARIAX: *The Machine Tool of the Future-Today!* Giddings & Lewis, Inc.: Fond Du Lac, WI, USA, 1994.
- Koch, P.M.; Kesteven, M.; Nishioka, H.; Jiang, H.; Lin, K.Y.; Umetsu, K.; Huang, Y.-D.; Raffin, P.; Chen, K.J.; Ibañez-Romano, F. The AMiBA hexapod telescope mount. *Astrophys. J.* **2009**, *694*, 1670–1684. [[CrossRef](#)]
- Liu, G.; Zheng, S.; Ogbobe, P.; Han, J. Inverse kinematic and dynamic analyses of the 6-UCU parallel manipulator. *Appl. Mech. Mater.* **2012**, *127*, 172–180. [[CrossRef](#)]
- Liu, G.; Qu, Z.; Liu, X.; Han, J. Singularity analysis and detection of 6-UCU parallel manipulator. *Robot Comput. Integr. Manuf.* **2014**, *30*, 172–179. [[CrossRef](#)]
- Dai, X.; Huang, Q.; Han, J.; Li, H. Accuracy Synthesis of Stewart Platform used in Testing System for Spacecraft Docking Mechanism. In Proceedings of the International Conference on Measuring Technology and Mechatronics Automation, ICMTMA'09, Zhangjiajie, China, 11–12 April 2009; Volume 3, pp. 7–10. [[CrossRef](#)]
- Wang, J.; Masory, O. On the Accuracy of a Stewart Platform-Part I: The Effect of Manufacturing Tolerances. In Proceedings of the IEEE International Conference on Robotics and Automation, Atlanta, GA, USA, 2–6 May 1993; pp. 114–120. [[CrossRef](#)]
- Ropponen, T.; Arai, T. Accuracy Analysis of a Modified Stewart Platform Manipulator. In Proceedings of the IEEE International Conference on Robotics and Automation, Nagoya, Japan, 21–27 May 1995; pp. 521–525. [[CrossRef](#)]

17. Patel, A.J.; Ehmann, K.F. Volumetric error analysis of a Stewart platform-based machine tool. *CIRP Ann. Manuf. Technol.* **1997**, *46*, 287–290. [[CrossRef](#)]
18. Wang, S.M.; Ehmann, K.F. Error model and accuracy analysis of a six-DOF Stewart platform. *J. Manuf. Sci. Eng.* **2002**, *124*, 286–295. [[CrossRef](#)]
19. Masory, O.; Wang, J.; Zhuang, H. On the Accuracy of a Stewart Platform—Part II: Kinematic Calibration and Compensation. In Proceedings of the IEEE International Conference on Robotics and Automation, Atlanta, GA, USA, 2–6 May 1993; pp. 725–731. [[CrossRef](#)]
20. Cong, D.; Yu, D.; Han, J. Kinematics accuracy analysis and error compensation of Stewart platform. *J. Eng. Des.* **2006**, *13*, 162–165. (In Chinese)
21. Merlet, J.P.; Daney, D. Dimensional Synthesis of Parallel Robots with a Guaranteed Given Accuracy over a Specific Workspace. In Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 2005, Barcelona, Spain, 18–22 April 2005; pp. 942–947. [[CrossRef](#)]
22. Merlet, J.-P.; Daney, D. Appropriate Design of Parallel Manipulators. In *Smart Devices and Machines for Advanced Manufacturing*; Wang, L., Xi, F., Eds.; Springer: London, UK, 2008; pp. 1–25. ISBN 978-1-84800-146-6.
23. Merlet, J.P. Interval analysis and reliability in robotics. *Int. J. Reliab. Saf.* **2009**, *3*, 104–130. [[CrossRef](#)]
24. Liu, G.; Qu, Z.; Han, J.; Liu, X. Systematic optimal design procedures for the Gough-Stewart platform used as motion simulators. *Ind. Robot* **2013**, *40*, 550–558. [[CrossRef](#)]
25. Angeles, J.; Park, F.C. Design and performance evaluation. In *Springer Handbook of Robotics*, 2nd ed.; Siciliano, B., Khatib, O., Eds.; Springer: Berlin, Germany, 2016; pp. 399–418. ISBN 978-3-319-32550-7.
26. Yu, X.; Gen, M. *Introduction to Evolutionary Algorithms*; Springer: London, UK, 2010; pp. 3–8. ISBN 978-1-84996-128-8.
27. Deb, K.; Pratap, A.; Agarwal, S.; Meyerivian, T.A.M.T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evolut. Comput.* **2002**, *6*, 182–197. [[CrossRef](#)]
28. Deb, K.; Agrawal, R.B. Simulated binary crossover for continuous search space. *Complex Syst.* **1994**, *9*, 1–34.
29. Deb, K.; Goyal, M. A combined genetic adaptive search (GeneAS) for engineering design. *Comput. Sci. Inf. Syst.* **1996**, *26*, 30–45.
30. Deb, K. An efficient constraint handling method for genetic algorithms. *Comput. Method Appl. Mech.* **2000**, *186*, 311–338. [[CrossRef](#)]
31. Advani, S.K. The Kinematic Design of Flight Simulator Motion Bases. PhD. Dissertation, TU Delft, Delft, The Netherlands, 1998; pp. 103–191.



© 2018 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Viability and Feasibility of Constrained Kinematic Control of Manipulators

Marco Faroni ¹, Manuel Beschi ², Nicola Pedrocchi ² and Antonio Visioli ^{1,*}

¹ Dipartimento di Ingegneria Meccanica e Industriale, University of Brescia, Via Branze 38, 25123 Brescia, Italy; m.faroni003@unibs.it

² Istituto di Sistemi e Tecnologie Industriali Intelligenti per il Manifatturiero Avanzato, National Research Council, Via Alfonso Corti 12, 20133 Milano, Italy; manuel.beschi@stiima.cnr.it (M.B.); nicola.pedrocchi@stiima.cnr.it (N.P.)

* Correspondence: antonio.visioli@unibs.it; Tel.: +39-030-3715426

Received: 13 June 2018; Accepted: 20 July 2018; Published: 23 July 2018

Abstract: Recent advances in planning and control of robot manipulators make an increasing use of optimization-based techniques, such as model predictive control. In this framework, ensuring the feasibility of the online optimal control problem is a key issue. In the case of manipulators with bounded joint positions, velocities, and accelerations, feasibility can be guaranteed by limiting the set of admissible velocities and positions to a viable set. However, this results in the imposition of nonlinear optimization constraints. In this paper, we analyze the feasibility of the optimal control problem and we propose a method to construct a viable convex polyhedral that ensures feasibility of the optimal control problem by means of a given number of linear constraints. Experimental and numerical results on an industrial manipulator show the validity of the proposed approach.

Keywords: manipulators; trajectory planning; kinematic constraints; optimization; viability; inverse kinematics

1. Introduction

Robotic systems typically present kinematic and/or dynamic limitations. For example, joint positions are generally bounded within an available range of motion, while actuators implicitly present velocity and acceleration/torque limits. Further constraints can be due to the environment where the robot has to operate (e.g., partial occupation of the robot workspace) or to safety reasons, which may determine velocity and acceleration limitations. Including such constraints in the development of planning and control methods is of utter importance, as their violation might lead to unrealizable motions (with consequent significant errors in the execution of the tasks) or to safety issues (for example, in the case workspace limits are not respected).

Constrained methods are widely diffused, for example, in the case of offline trajectory planning. The general approach consists in the formulation of a constrained optimization problem that should optimize a given objective, such as minimum execution time [1] or minimum energy consumption [2]. This is also the case of global redundancy resolution methods [3], where the robot has to perform an assigned task, and the redundancy can be exploited to optimize a desired objective, such as maximum manipulability [4], dexterity [5], or maximum joint range availability [6]. However, all these methods are typically performed offline as they need the prior knowledge of the task to be performed, and they lead to heavy computational burdens that do not permit their online execution. For this reason, they are not able to handle online trajectory generation and re-planning techniques and this is a relevant limitation, considering recent robotic applications such as collaborative robotics and robots in unstructured environments. To handle the robot constraints in online planning and control methods,

optimization-based techniques are becoming more and more widespread as the increasing computing power of modern processors allows their real-time implementation with small sampling periods.

Quadratic programming, for example, is intensively exploited in the resolution of the Inverse Kinematic (IK) problem of redundant manipulators [7–11]. In brief, a manipulator is termed kinematically redundant with respect to a task when it has a number of degrees of freedom greater than the one of the desired task. The IK problem does not present a unique solution, as the transformation between the joint and the task space results in an undetermined system. Such problem is usually addressed at differential level in order to exploit the linear relation between joint and task velocities given by the Jacobian of the robot. This can be formulated as a chain of Quadratic Programs (QP) with decreasing priorities: the tasks are written as cost functions and the robot limits are represented as linear constraints. The possibility of including robot constraints in the QP represents a great advantage with respect to other typical methods (such as the unconstrained projection of secondary tasks in the null space of the Jacobian [12,13]). Further developments in this field explicitly face the problem of possible saturations and task deformations by introducing a scaling variable into the QP to slow down the execution of the tasks if needed [14].

A further improvement of these methods is represented by the use of Model Predictive Control (MPC) techniques, which aim at overcoming issues related to the fact that the previous methods only consider the current state of the robot. In fact, MPC-based methods are able to take into account the future evolution of the system, the tasks, and the constraints, improving the behavior of the robot in terms of task satisfaction and smoothness [15,16]. MPC is also applied to constrained motion planning [17,18] or low-level control applications [19].

Disregarding the particular field of application, a fundamental issue to address when using optimization-based control techniques is the feasibility of the online optimization problem. In fact, feasibility must be ensured in any possible state of the system. Otherwise, a solution to the problem might not exist, and this could cause the algorithm to stop (unless specific but sub-optimal strategies are activated when infeasibility occurs).

Feasibility of the online optimization problem is strictly related to the concept of set viability [20]. A set is said to be viable if, given an initial state within such set, the state trajectory can be kept within the set by means of a proper and realizable input function. In other words, keeping the state within a viable set ensures feasibility for all future time instants. On the contrary, if the state exits the viable set, infeasibility of the control problem will surely occur at a certain time in the future. This is particularly relevant in the case of state-constrained systems, as infeasibility might occur if no viability conditions are added to the control problem but only state and input limits are taken into account.

For robot manipulator control, this issue is common to any control strategy that takes into account both position and acceleration/torque constraints. In this case, the simplistic imposition of box constraints on the position and the acceleration/torque of the robot does not ensure the existence of a feasible solution to the optimization problem. For example, if a robot joint approaches its position limit with high velocity, the position bound will be exceeded due to the bounded admissible deceleration. Few works addressed this issue by means of manually-tuned heuristic strategies that aimed at reducing the velocity of the robot when it approached its position limits [21–23]. Many other control strategies did not take into account position bounds, assuming that the reference trajectories were implicitly feasible [24,25]. Notice that the viability guarantee does not ensure the feasibility of the reference trajectory, which could be only verified offline when the whole trajectory is known a priori. Such approach does not apply to online methods, which should be able to handle online trajectory generation and re-planning. In these cases, we can only ensure the feasibility of the optimal control, which typically aims at minimizing the deviation of the performed motion with respect to the reference trajectory. This means that, although the desired motion might be infeasible, the optimal control problem remains feasible. In this case, a deformation with respect to the nominal trajectory could arise.

A formal viability guarantee was given in [26], which proposed an invariance control scheme for constrained robot manipulators. In particular, viability conditions were derived in order not to exceed workspace limitations with bounded joint accelerations. In [14], a local redundancy resolution technique was proposed: to respect a given deceleration limit in the braking phase, the position-velocity state region was limited by imposing that the robot was able to stop within the given joint limit (in fact, limiting the state region to a viable set). This approach was similarly adopted in [27] in the context of robust constrained motion planning. Recently, viability for joint-constrained manipulators was explicitly addressed in [28], with a particular focus on the discrete-time implementation of the viability constraints.

All the above-mentioned approaches were based on the derivation of an analytical viability condition for a double integral system with bounded input and output. The resulting condition is quadratic in the velocity state. However, as all the above-mentioned works refer to local/feedback methods, such condition could be easily linearized around the current system state, resulting in variable box constraints on the control actions. However, such linearization is not possible for predictive strategies. In this case, a linear approximation of the viability set should be obtained. For example, Faroni et al. [29] gives a sufficient condition to approximate the quadratic viable set with a linear constraint and maintaining the viability property. Although the approximation technique is easy to implement and does not remarkably increase the computational complexity of the problem, the resulting allowed state region could result to be significantly smaller than the original one.

In this paper, we propose a method to derive a viable convex polyhedron for a robotic system with bounded joint positions, velocities, and accelerations. In particular, a simple optimization problem is set up to determine the maximum polyhedron that approximates the original viable set. The original quadratic condition is approximated by a polyhedron with a given number of sides, by maximizing the area of the allowed velocity-position state-region. Moreover, viability of the resulting set is ensured by imposing that, for all points on the polyhedron boundary, there exists a realizable input action that keeps the next state within the polyhedron itself, which is also demonstrated to be convex and can be therefore rewritten as a linear inequality constraint in optimization-based algorithms (such as linear MPC techniques). The paper shows that, by increasing the number of sides, the polyhedron gives a better approximation of the original viable set, as expected. This means that a larger admissible state region can be exploited by the controller when the resulting constraints are included in the optimization problem. Consequently, the controller can obtain smaller tracking errors when the desired task requires the robot joint to get close to the maximal viable set boundary. In particular, numerical results show the enlargement of the admissible state region as the number of sides increases. Such improvement is more and more evident as the maximum acceleration values gets smaller. Finally, experimental results on a Universal Robots UR10 manipulator demonstrates the validity of the proposed approach. In particular, an MPC algorithm is applied to the tracking problem of a given joint-space trajectory and the results with different viability conditions are compared. Firstly, an experimental example shows that the use of an invariance condition is of vital importance to ensure both the feasibility of the online optimization problem and the satisfaction of the manipulator limits. Secondly, the results show that the viable sets obtained by means of the proposed method permits to obtain smaller (or null) tracking errors in the case the required motion is close to the viability boundary, increasing the performance of the MPC controller.

The paper is organized as follows. Section 2 introduces the concept of set viability applied to constrained kinematic control of robot manipulators. Section 3 illustrates the proposed method for the computation of the optimal viable polyhedron and gives viability and convexity proves. Numerical results demonstrate the effectiveness of the proposed approach in Section 4, while experimental results on a Universal Robot UR10 manipulator are shown in Section 5. Section 6 concludes the paper.

2. Feasibility of the Constrained Kinematic Control Problem

Consider a generic robot joint and denote with q, \dot{q}, \ddot{q} its position, velocity, and acceleration, respectively. The limits on q, \dot{q} , and \ddot{q} can be therefore expressed as:

$$q_{\min} \leq q \leq q_{\max}, \tag{1}$$

$$\dot{q}_{\min} \leq \dot{q} \leq \dot{q}_{\max}, \tag{2}$$

$$\ddot{q}_{\min} \leq \ddot{q} \leq \ddot{q}_{\max}, \tag{3}$$

where $q_{\min}, \dot{q}_{\min}, \ddot{q}_{\min}$, and $q_{\max}, \dot{q}_{\max}, \ddot{q}_{\max}$ are the minimum and maximum joint position, velocity, and acceleration, respectively.

Assume a discrete-time implementation with sampling period T in which the acceleration is considered as constant along each sampling period. At time k , kinematic limits at the next sampling time $k + 1$ can be easily written as linear inequalities in the joint acceleration \ddot{q}_k as:

$$q_{\min} \leq q_k + T \dot{q}_k + \frac{1}{2} T^2 \ddot{q}_k \leq q_{\max}, \tag{4}$$

$$\dot{q}_{\min} \leq \dot{q}_k + T \ddot{q}_k \leq \dot{q}_{\max}, \tag{5}$$

$$\ddot{q}_{\min} \leq \ddot{q}_k \leq \ddot{q}_{\max}. \tag{6}$$

However, the simplistic imposition of Equations (4)–(6) may result in an empty admissible set, as no feasible solution might exist that satisfies all constraints at the same time [28].

Indeed, the existence of a solution is guaranteed if Equations (4)–(6) are feasible for all the admissible states of the system. This is strictly correlated to the concept of set viability, as mentioned in Section 1. As each joint is modeled as a double integrator, the viability analysis traces back to the viability of a double integrator system with bounded input and output. By imposing the feasibility of Equations (4)–(6), the maximal viability set for the double integrator can be derived analytically [26]. Intuitively, it can be calculated by imposing that, applying the maximum deceleration \ddot{q}_{\min} , the joint stops at $q = q_{\max}$ with null velocity ($\dot{q} = 0$). The resulting condition (valid for the upper bound) is given by:

$$\begin{cases} q - \frac{\dot{q}^2}{2\ddot{q}_{\min}} - q_{\max} \leq 0 & \text{if } \dot{q} > 0 \\ q - q_{\max} \leq 0 & \text{otherwise} \end{cases} \tag{7}$$

which expresses a quadratic condition in the system states. An analogous condition for the lower bound can be derived likewise.

Such condition can be easily linearized around the current velocity and position, in the case of local methods, as in [14,27]. The inclusion of Equation (7) in the QP ensures that the states of the system (i.e., the joint velocities and positions) remain within a viable set for which the problem is feasible. More details about the discrete implementation in robotic systems can be found in [28].

The necessity of deriving a linear approximation of the original quadratic equation (Equation (7)) comes from the advantages given by the linear formulation in optimization-based controllers (in particular, the significant decrement of computational time and the ease of implementation of linear MPC with respect to the nonlinear one). However, in the case of predictive strategies, the linearization adopted for local methods is not applicable.

To tackle this issue, Faroni et al. [29] proposed a linear viability condition based on a single constraint for each joint. Firstly, it showed that approximating the quadratic constraint in Equation (7) by means of a straight line passing through the extreme points of the maximal viable set (i.e., $(q_{\max}, 0)$ and the intersection between Equation (7) and $\dot{q} = \dot{q}_{\max}$ in Figure 1) leads to a non-viable set. Then, it derives a linear viability condition by imposing that the maximum deceleration along the linear

constraint is exactly equal to the maximum admissible deceleration. In this way, the joint states can be always kept within the resulting set by means of a realizable acceleration. Such condition is given by:

$$-\frac{\ddot{q}_{\min}}{\dot{q}_{\max}} q + \dot{q} + \frac{\ddot{q}_{\min}}{\dot{q}_{\max}} q_{\max} \leq 0 \tag{8}$$

for the upper bound, and likewise for the lower one.

Inequalities (Equations (7) and (8)) are graphically represented in Figure 1. Linearity of Equation (8) is a great advantage, as it allows the straightforward inclusion in an optimization problem as a linear constraints. However, it is clear that Equation (8) implies a conservative reduction of the available state-space region compared to Equation (7), and such reduction worsens as the acceleration limits become smaller.

From a practical perspective, the reduction of the admissible state region shows its drawbacks when the robot is required to perform a motion that would violate Equation (8). In fact, the states laying between the linear and the quadratic constraints in Equations (7) and (8) are potentially realizable by the robot, but they are automatically excluded by the controller, with consequent deformation of the desired trajectory. Similarly, for redundant manipulators, shrinking the admissible state region could result in a worse satisfaction of the secondary objectives.

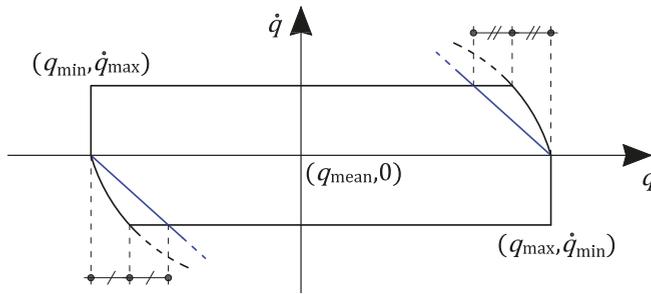


Figure 1. Viable admissible set for the double integrator system. Black: Maximal region, given by the quadratic inequality in Equation (7). Blue: Conservative linear inequality in Equation (8), as proposed in [29].

3. Proposed Method

As mentioned in the previous section, viability conditions are fundamental to ensure the feasibility of the control problem. The derivation of linear viability conditions (e.g., Equation (8)) allows their straightforward implementation in an optimization-based control framework, such as MPC algorithms. A strategy to enlarge the admissible region without waving its linearity is therefore proposed. It consists in increasing the number of linear constraints that compose the admissible set. We can therefore state the following problem.

Problem 1. *Given the acceleration, velocity, and configuration limits for each joint, find the maximal viable convex polyhedron determined by a given number of linear inequalities.*

We address this problem by converting it into an optimization problem. In particular, for each joint, two optimization problems are set up (one for the upper position bound and one for the lower position bound). Without loss of generality, consider the upper configuration bound for a generic joint (i.e., the first quarter in Figure 1). The optimization variables of the problem are given by the

coordinates of the extremes of the segment composing the polyhedron shown in Figure 2. Given a number of segments h , such variables are denoted with the vectors:

$$p_x = (p_{x,0}, \dots, p_{x,j}, \dots, p_{x,h}), \quad (9)$$

$$p_y = (p_{y,0}, \dots, p_{y,j}, \dots, p_{y,h}), \quad (10)$$

whereas a generic point on the polyhedron is denoted with $P_j(p_{x,j}, p_{y,j})$.

Now, impose that $(p_{x,0}, p_{y,0}) = (q_{\text{mean}}, \dot{q}_{\text{max}})$, where $q_{\text{mean}} = \frac{q_{\text{max}} + q_{\text{min}}}{2}$ and $(p_{x,h}, p_{y,h}) = (q_{\text{max}}, 0)$. To maximize the area covered by the polyhedron, a cost function equivalent to the opposite of such area is defined as follows:

$$\psi = \frac{1}{2} \sum_{j=1}^h (p_{x,j-1} - p_{x,j})(p_{y,j-1} + p_{y,j}). \quad (11)$$

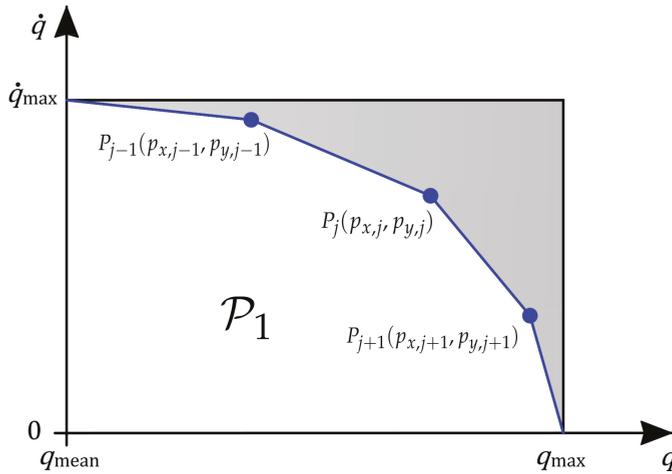


Figure 2. Construction of the polyhedron \mathcal{P}_1 .

The extremes of the segments are required to lay in the first quarter and have to respect the maximum position and velocity bounds. This results in the following box constraints:

$$q_{\text{mean}} \leq p_x \leq q_{\text{max}}, \quad (12)$$

$$0 \leq p_y \leq \dot{q}_{\text{max}}. \quad (13)$$

Moreover, as a first necessary condition to the convexity of the polyhedron, we impose:

$$p_{x,j} - p_{x,j-1} \geq 0 \quad \forall j = 1, \dots, h, \quad (14)$$

$$p_{y,j-1} - p_{y,j} \geq 0 \quad \forall j = 1, \dots, h. \quad (15)$$

One last constraint needs to be imposed to ensure that the resulting polyhedron is viable and convex. To this purpose, we resort to a geometrical reasoning. Consider a generic segment $\overline{P_j P_{j+1}}$ whose extremes are the points $(p_{x,j}, p_{y,j})$ and $(p_{x,j+1}, p_{y,j+1})$ and denote with \vec{n}_j the unitary vector normal to the segment and directed toward the inner region of the polyhedron. It results that:

$$\vec{n}_j = \left(\frac{p_{y,j+1} - p_{y,j}}{\|\overline{P_j P_{j+1}}\|}, \frac{p_{x,j} - p_{x,j+1}}{\|\overline{P_j P_{j+1}}\|} \right). \quad (16)$$

Consider then a generic point $P_{\zeta}(p_{x,\zeta}, p_{y,\zeta})$ on the segment. By applying the maximum deceleration \ddot{q}_{\min} to the system, the states are forced to move on the curve given by:

$$\begin{cases} q = p_{x,\zeta} + p_{y,\zeta}t + \frac{1}{2}\ddot{q}_{\min}t^2 \\ \dot{q} = p_{y,\zeta} + \ddot{q}_{\min}t \end{cases} \quad (17)$$

The tangent vector to such curve in the (q, \dot{q}) -plane is given by:

$$\vec{t} = \left(\frac{dq}{dt}, \frac{d\dot{q}}{dt} \right) = (p_{y,\zeta} + \ddot{q}_{\min}t, \ddot{q}_{\min}). \quad (18)$$

The vector \vec{t} at $t = 0$ is therefore the tangent vector in the point P_{ζ} and results to be:

$$\vec{t}(p_{x,\zeta}, p_{y,\zeta}) = (p_{y,\zeta}, \ddot{q}_{\min}). \quad (19)$$

Note that $\vec{t}(p_{x,\zeta}, p_{y,\zeta})$ represents the direction of the state movement when the maximum realizable deceleration is applied. The state will therefore be able to remain below the considered segment if the following condition holds:

$$\vec{n}_j \cdot \vec{t}(p_{x,\zeta}, p_{y,\zeta}) \geq 0 \quad \forall (p_{x,\zeta}, p_{y,\zeta}) \in \overline{P_j P_{j+1}} \quad (20)$$

where (\cdot) denotes the scalar product between two vectors. As

$$\min_{p_{x,\zeta}, p_{y,\zeta}} (\vec{n}_j \cdot \vec{t}(p_{x,\zeta}, p_{y,\zeta})) = \vec{n}_j \cdot \vec{t}(p_{x,j}, p_{y,j}), \quad (21)$$

condition in Equation (20) can be imposed on the sole extreme point P_j of each segment and becomes:

$$p_{y,j}(p_{y,j+1} - p_{y,j}) + \ddot{q}_{\min}(p_{x,j} - p_{x,j+1}) \geq 0. \quad (22)$$

Finally, the resulting optimization problem can be written as:

$$\begin{aligned} & \underset{p_x, p_y}{\text{minimize}} && \psi \\ & \text{subject to} && q_{\text{mean}} \leq p_x \leq q_{\text{max}} \\ & && 0 \leq p_y \leq \dot{q}_{\text{max}} \\ & && p_{x,j} - p_{x,j-1} \geq 0 \quad \forall j = 1, \dots, h \\ & && p_{y,j-1} - p_{y,j} \geq 0 \quad \forall j = 1, \dots, h \\ & && p_{y,j}(p_{y,j+1} - p_{y,j}) + \ddot{q}_{\min}(p_{x,j} - p_{x,j+1}) \geq 0 \quad \forall j = 1, \dots, h-1 \\ & && p_{x,0} = q_{\text{mean}} \\ & && p_{y,0} = \dot{q}_{\text{max}} \\ & && p_{x,h} = q_{\text{max}} \\ & && p_{y,h} = 0 \end{aligned} \quad (23)$$

Following the same reasoning, an analogous optimization problem can be set up for the lower position bound. Note that, if the joint limits are symmetric, the solution for the lower bounds can be obtained by “mirroring” the solution of Equation (23) into the third quarter of the (q, \dot{q}) -plane. Notice that, as Equation (23) is non-convex, global optimization such as genetic or multi-start algorithms should be adopted to solve the optimization problem (see Section 4).

Denoting with \mathcal{P}_1 the polyhedron determined by the segments obtained as solution of Equation (23), and with \mathcal{P}_2 the analogous polyhedron obtained for the lower bounds, the overall polyhedron for the single joint is given by:

$$\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2 \cup [q_{\min}, q_{\text{mean}}] \times [0, \dot{q}_{\text{max}}] \cup [q_{\text{mean}}, q_{\text{max}}] \times [\dot{q}_{\min}, 0]. \quad (24)$$

Proposition 1. *Polyhedron \mathcal{P} is a viable set with respect to the given joint position, velocity and acceleration limits. Moreover, such polyhedron is convex.*

Proof (Viability). Viability of the set is implicitly ensured by the imposition of Equation (22). □

Proof (Convexity). To prove it by contradiction, denote the solution of Equation (23) with p_x^*, p_y^* and assume that the corresponding polyhedron \mathcal{P}_1^* (i.e., the part of the polyhedron in the first quarter) is non-convex. Consider then a triple of consequent extremes $\{P_{j-1}, P_j, P_{j+1}\}$, which causes a non-convexity. Equations (20) and (21) imply:

$$\vec{n}_{j-1} \cdot \vec{t}(p_{x,j-1}^*, p_{y,j-1}^*) \geq 0, \quad (25)$$

$$\vec{n}_j \cdot \vec{t}(p_{x,j}^*, p_{y,j}^*) \geq 0. \quad (26)$$

Consider now the point P_c given by the projection of P_j onto $\overline{P_{j-1}P_{j+1}}$ and denote with \vec{n}_c the vector normal to $\overline{P_{j-1}P_{j+1}}$. We want to prove that the polyhedron obtained by substituting P_c to P_j in \mathcal{P}_1^* is a feasible and more efficient solution of Equation (23). First, constraints in Equations (12)–(15) are straightforwardly satisfied. Moreover,

$$\vec{n}_c \cdot \vec{t}(p_{x,j-1}^*, p_{y,j-1}^*) \geq \vec{n}_{j-1} \cdot \vec{t}(p_{x,j-1}^*, p_{y,j-1}^*) \geq 0 \quad (27)$$

as $\vec{n}_{j-1} \cdot \vec{t} \leq \vec{n}_c \cdot \vec{t} \leq 0$ and $\vec{n}_c \cdot \vec{j} \leq \vec{n}_{j-1} \cdot \vec{j} \leq 0$ by construction (where \vec{t} and \vec{j} denote the unitary vectors directed as the horizontal and the vertical axis, respectively).

Moreover, as \dot{q}_{\min} in Equation (19) is assumed to be negative, the following inequality holds:

$$\vec{n}_c \cdot \vec{t}(p_{x,c}, p_{y,c}) \geq \vec{n}_c \cdot \vec{t}(p_{x,j-1}^*, p_{y,j-1}^*) \geq 0. \quad (28)$$

Note that Equations (27) and (28) implies that the new candidate solution satisfies Equation (22) and is therefore a feasible solution to Equation (23). Moreover, the candidate solution is more efficient than the assumed one, as

$$\psi(P_0, \dots, P_{j-1}, P_c, P_{j+1}, \dots, P_h) \leq \psi(P_0, \dots, P_{j-1}, P_j, P_{j+1}, \dots, P_h), \quad (29)$$

which means that p_x^*, p_y^* are not the optimal solution of Equation (23), as a feasible and more efficient solution that eliminates such non-convexity exists. Applying this reasoning to any triples of extremes that cause a non-convexity implies that the optimal polyhedron \mathcal{P}_1 must be convex. The same demonstration can be applied to \mathcal{P}_2 leading to the same conclusion. □

4. Numerical Results

Consider the first joint of the robot manipulator Universal Robot UR10. The joint configuration and velocity limits from the datasheet are given by:

$$q_{\max} = -q_{\min} = 3.14 \text{ rad}, \quad \dot{q}_{\max} = -\dot{q}_{\min} = 2.16 \text{ rad/s}. \tag{30}$$

Typically, acceleration limits are not explicitly given in the datasheet. Indeed, the actual dynamic limit of the joint is usually determined by the available joint torque. However, handling viability directly in terms of torque limits is typically avoided due to the complexity of the problem. The common approach consists in estimating a (usually conservative) corresponding acceleration limit for each joint separately. A method for the estimation of such acceleration limit is given, for instance, in [28]. As a first example, assume that the acceleration limits are given by $\ddot{q}_{\max} = -\ddot{q}_{\min} = 6 \text{ rad/s}^2$.

We solve the optimization problem (Equation (23)) for different values of h to evaluate the resulting polyhedrons as the number of edges grows and to compare the results to the quadratic and the linear viability constraints (Equations (7) and (8), respectively). Table 1 shows the values of the inner area of polyhedron \mathcal{P}_1 . The values are normalized with respect to the maximal viable set obtained by means of Equation (7), whose area is therefore equal to one (last column of Table 1). As expected, the smaller area is given by Equation (8) (first column of Table 1), while the extension of the polyhedron obtained by solving Equation (23) grows with the number of edges. In other words, the larger the number of edges, the closer the optimal polyhedron is to the maximal extension (given by Equation (7)). This is clearly shown in Figure 3, which depicts the different viable sets.

An analogous example is performed by using $\ddot{q}_{\max} = -\ddot{q}_{\min} = 3 \text{ rad/s}^2$. The values of the normalized area of \mathcal{P}_1 are shown in Table 2 and the resulting viable sets are depicted in Figure 4. The results lead to conclusions similar to the ones given by the previous example. However, it is clear that the magnitude of the phenomenon grows as the acceleration limits get smaller.

Of course the use of a larger value of h also gives some drawbacks. First, the computational complexity of Equation (23) rapidly grows with h as the number of variables and constraints is linear in h . As an example, the time needed to solve Equation (23) for $h = 1$ and $h = 11$ was in the order of 1 second and 15 s, respectively (the computation was performed in Matlab using a multi-start gradient-descent method on a standard laptop mounting a 2.5 GHz Intel Core i5-2520M processor).

Furthermore, the value of h determines the number of linear inequalities describing the polyhedron. In optimization-based control algorithms (such as linear MPC), this affects the computational complexity of the online optimal control problem, which is typically a critical issue in online methods, as mentioned in Section 1.

Table 1. Inner area of \mathcal{P}_1 for different values of h in case $\ddot{q}_{\max} = -\ddot{q}_{\min} = 6 \text{ rad/s}^2$.

Equation (8)	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 8$	$h = 9$	$h = 10$	$h = 11$	Equation (7)
0.9139	0.9636	0.9773	0.9836	0.9871	0.9895	0.9911	0.9920	0.9923	0.9925	1

Table 2. Inner area of \mathcal{P}_1 for different values of h in case $\ddot{q}_{\max} = -\ddot{q}_{\min} = 3 \text{ rad/s}^2$.

Equation (8)	$h = 3$	$h = 4$	$h = 5$	$h = 6$	$h = 7$	$h = 8$	$h = 9$	$h = 10$	$h = 11$	Equation (7)
0.8200	0.9239	0.9525	0.9656	0.9731	0.9731	0.9813	0.9838	0.9857	0.9857	1

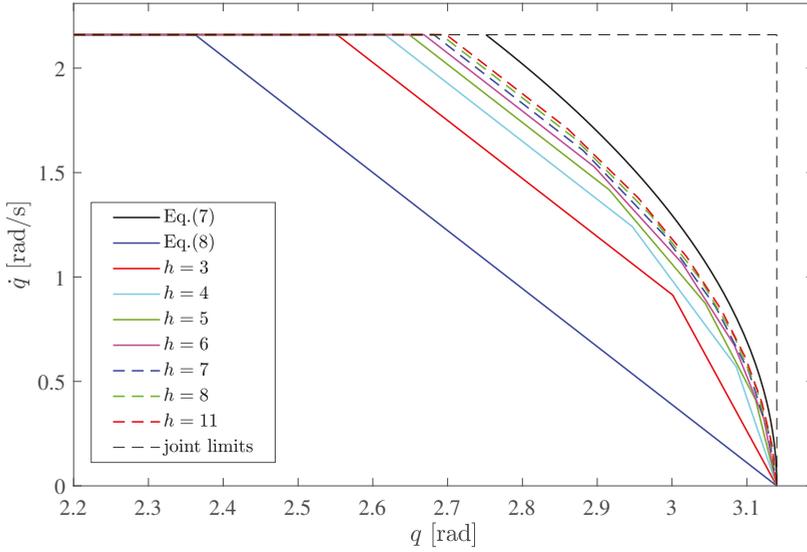


Figure 3. Comparison of the viable polyhedrons for different values of h (portion of interest of the first quarter) in the case $\dot{q}_{\max} = -\dot{q}_{\min} = 6 \text{ rad/s}^2$.

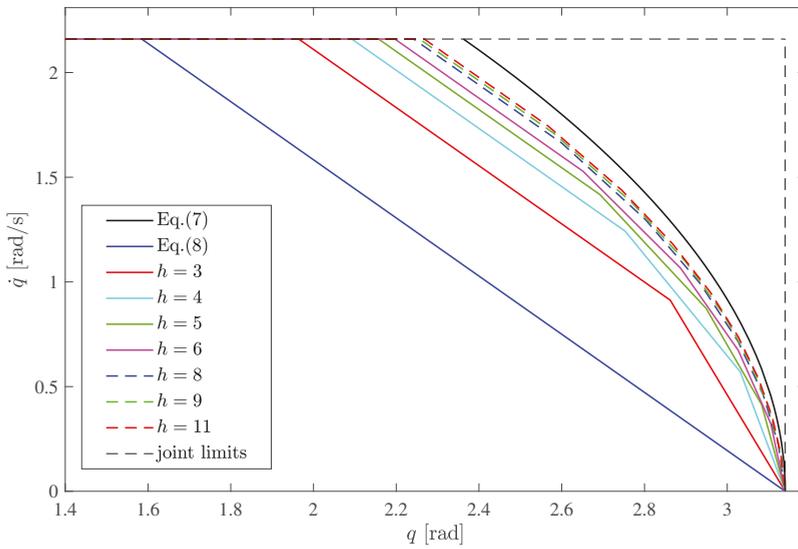


Figure 4. Comparison of the viable polyhedrons for different values of h (portion of interest of the first quarter) in the case $\dot{q}_{\max} = -\dot{q}_{\min} = 3 \text{ rad/s}^2$.

5. Experimental Results

Experimental were also performed on a six-degrees-of-freedom Universal Robot UR10 manipulator to prove the validity of the proposed approach. For the purposes of the paper, the joint position and velocity limits have been set to slightly conservative values with respect to the ones given in the datasheet. In particular, they are set as:

$$Q_{\max} = -Q_{\min} = (3, 3, 3, 3, 3, 3) \text{ rad}, \quad (31)$$

$$\dot{Q}_{\max} = -\dot{Q}_{\min} = (2, 2, 3, 3, 3, 3) \text{ rad/s}. \quad (32)$$

The acceleration limits have been set as $\ddot{Q}_{\max} = -\ddot{Q}_{\min} = (3, 3, 3, 3, 3, 3) \text{ rad/s}^2$. The experimental platform is shown in Figure 5. The objective of these experiments was the evaluation of the performance of an optimization-based control algorithm when different viability constraints are implemented. To this purpose, a simple MPC scheme was applied to a trajectory following problem. The MPC control scheme is in charge of following a position reference signal in the joint space. The model implemented in the MPC consists of a double integrator for each joint, as typical of robot kinematic control [15,30]. The input action is therefore represented by the joint accelerations, which then feed the low-level controller of the robot. Joint position, velocity, and acceleration limits can be implemented as linear constraints in the MPC (see [15] for details) and the online optimal control problem results to be a QP which minimizes the weighted sum of the tracking error and the control effort, as typical of linear MPC [31]. Notice that the choice of such a simple control scheme is due to the will of highlighting the behavior of a linear MPC controller in the case of different viability conditions. However, the proposed method could be straightforwardly applied to more complex MPC techniques such as [15,16,29].



Figure 5. Universal Robot UR10 used as experimental platform.

The MPC algorithm was implemented using a sampling period $T = 8 \text{ ms}$ and by setting a predictive and a control horizon $N = 20$ sampling periods. The robot trajectory is controlled by means of a ROS-based control architecture. Namely, a position controller runs in a ROS Kinetic Ubuntu 16.04. The controller communicates with the robot by means of a TCP connection. The controller takes the MPC position output as reference and receives the actual joint position. The controller output is the sum of a proportional action, with gain equal to seven, and a feedforward term equal to the MPC velocity output.

We considered a simple trajectory given by a straight line in the joint space, parameterized with respect to the normalized longitudinal length along the path, denoted with $r \in [0, 1]$. The trajectory $Q_{des}(t)$ is therefore defined as:

$$Q_{des}(t) = Q_{start} + (Q_{end} - Q_{start}) r(t), \quad (33)$$

where $Q_{start} = (0, -1, 0.5, 0, 0, 0.5)$ rad and $Q_{end} = (2.9, -0.5, -1.5, 1, -0.5, 2.9)$ rad are the initial and the final points of the trajectory, respectively, and $r : [0, t_{end}] \rightarrow [0, 1]$, $t \mapsto r(t)$ is defined as a timing law with trapezoidal velocity profile, where t_{end} represents the total time of the trajectory, as depicted in Figure 6.

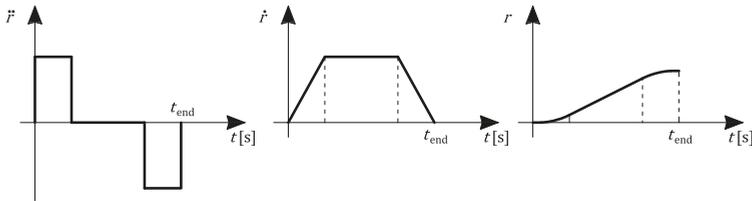


Figure 6. Timing law with trapezoidal velocity profile.

The control problem the MPC controller has to solve at each cycle k therefore results:

$$\begin{aligned} & \underset{\ddot{Q}}{\text{minimize}} && \sum_{i=1}^N \|Q_{des}(k+j) - Q(k+j)\|^2 + \lambda \sum_{i=0}^{N-1} \|\ddot{Q}(k+j)\|^2 \\ & \text{subject to} && \dot{Q}_{min} \leq \dot{Q}(k+j) \leq \dot{Q}_{max} \quad \forall j = 1, \dots, N \\ & && \ddot{Q}_{min} \leq \ddot{Q}(k+j) \leq \ddot{Q}_{max} \quad \forall j = 0, \dots, N-1 \\ & && Q(k+1) = Q(k) + T\dot{Q}(k) + 0.5T^2\ddot{Q}(k) \\ & && \dot{Q}(k+1) = \dot{Q}(k) + T\ddot{Q}(k) \end{aligned} \quad (34)$$

where Q , \dot{Q} , and $\ddot{Q} \in \mathbb{R}^6$ are the joint position, velocity, and acceleration vectors, respectively, and the controller effort weighting factor $\lambda = 1 \times 10^{-6}$ was tuned empirically. Different implementations of the position limits will then be added to the problem in order to evaluate the different behaviors of the system.

As a first example, the total time of the trajectory was chosen as $t_{end} = 2.07$ s and the behavior of the MPC controller was evaluated in three different scenarios:

- No position bounds are implemented.
- Box position constraints are implemented, which means that the following constraints are added to Equation (34):

$$Q_{min} \leq Q(k+j) \leq Q_{max} \quad \forall j = 1, \dots, N \quad (35)$$

- The linear viability condition in Equation (8) is applied, that is, Equation (35), and the following constraint are added to Equation (34):

$$-\frac{\ddot{Q}_{i,min}}{\ddot{Q}_{i,max}} Q_i(k+j) + \dot{Q}_i(k+j) + \frac{\ddot{Q}_{i,min}}{\ddot{Q}_{i,max}} Q_{i,max} \leq 0 \quad \forall j = 1, \dots, N, \quad \forall i = 1, \dots, 6; \quad (36)$$

where the subscript i denotes the i th element of the vector.

The viability constraints are imposed throughout the whole control horizon for the sake of clarity and because the small increment in the computational time (due to the larger number of linear constraints) does not represent a significant issue in the presented experimental tests. However, in the case of linear MPC, imposing such constraints only on the last time instant of the horizon would be enough to ensure feasibility of the control problem and would not give significant differences in terms of control performance. The imposition of the constraint throughout the whole horizon permits to obtain a more reliable prediction of the future trajectory. This can be helpful especially in the case the prediction is utilized in linearized methods such as [15], as a non-reliable prediction might worsen the performance of the method.

Notice that the trajectory is devised in such a way that the desired motions of the first and the sixth joints exceed the maximal viable set, as shown in Figure 7a,b. The figures also show the phase plot of the position/velocity variables computed by the MPC in the three above-mentioned cases. In the case no position bounds are implemented (dashed green line), the joint position obviously does not satisfy the position limit. However, when box position constraints are implemented in the MPC (dash-dotted purple line), infeasibility of the control problem occurs and, since that time, the joint is forced to decelerate with the maximum admissible deceleration, but the position limit cannot be satisfied anyway. This highlights the importance of the viability property. In fact, the implementation of the linear constraints in Equation (8) does ensure the feasibility of the control problem and permits to satisfy the position bound (with a deformation of the original trajectory) (dashed red line). These behaviors are clarified also in Figure 8a,b, where the ideal and the measured joint positions for Joint 1 and Joint 6 are shown.

A second experiment is performed by choosing a total trajectory time $t_{end} = 2.65$ s. In this case, the trajectory drives the first and the third joints in the region between Equations (7) and (8). The behavior of the MPC controller is evaluated in cases where three different viability constraints are implemented:

- The linear viability condition in Equation (8), that is, Equation (36) is added to Equation (34).
- The viable polyhedron obtained by solving Equation (23) with $h = 3$, that is, the following constraints are added to Equation (34):

$$(Q_i(k+j), \dot{Q}_i(k+j)) \in \mathcal{P}_{h=3}^i \quad \forall j = 1, \dots, N, \quad \forall i = 1, \dots, 6; \quad (37)$$

where the superscript i denotes the viable polyhedral of the i th joint.

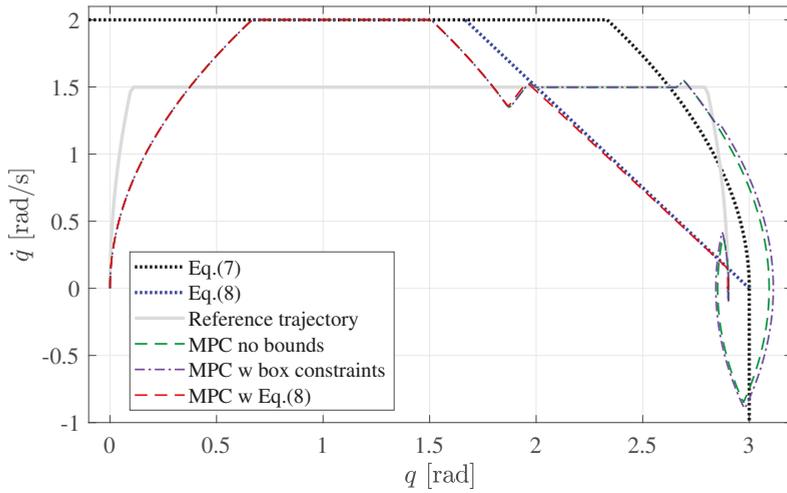
- The viable polyhedron obtained by solving Equation (23) with $h = 4$, that is, the following constraints are added to Equation (34):

$$(Q_i(k+j), \dot{Q}_i(k+j)) \in \mathcal{P}_{h=4}^i \quad \forall j = 1, \dots, N, \quad \forall i = 1, \dots, 6. \quad (38)$$

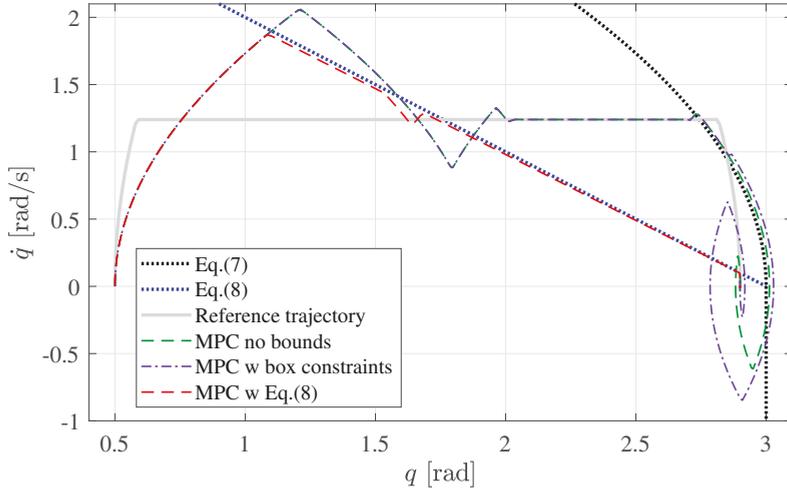
Figure 9a,b shows the state trajectory for the first joint in the three different cases. Notice that the position limit is respected and the control problem remains feasible for all cases. However, the reduction of the admissible region determined by Equation (8) gives rise to a significant modification of the original trajectory, although the desired states are always potentially realizable by the robot (as the trajectory does not exceed Equation (7)). An improvement is obtained when the polyhedron with $h = 3$ is used.

Notice that the small velocity bumps visible in the figures are due to predictive nature of the controller. In fact, as the control scheme is based on the tracking of a position reference along the predictive horizon, the controller slightly increases the velocity when it realizes the viability constraint will be activated and a deformation of the task will arise (as the state will be forced to follow the constraint). By giving a small acceleration before the activation of the constraint, the controller minimizes the future deviation with respect to the given position reference.

Finally, the use of $h = 4$ permits to obtain a viable polyhedron that is large enough to enclose the whole desired trajectory. These behaviors are clarified also in Figure 10a,b, where the ideal and the measured joint positions for Joint 1 and Joint 6 are shown.



(a)



(b)

Figure 7. State trajectory in the (q, \dot{q}) -plane for (a) Joint 1 and (b) Joint 3 for $T_{tot} = 2.07$ s: Reference trajectory (gray solid line); MPC without position bounds (green dashed line); MPC with box position bounds (purple dash-dot line); MPC with linear viability inequality (Equation (8)) (red dashed line); maximal viable set (black dotted line); and linear viability inequality ((Equation (8)) (blue dotted line).

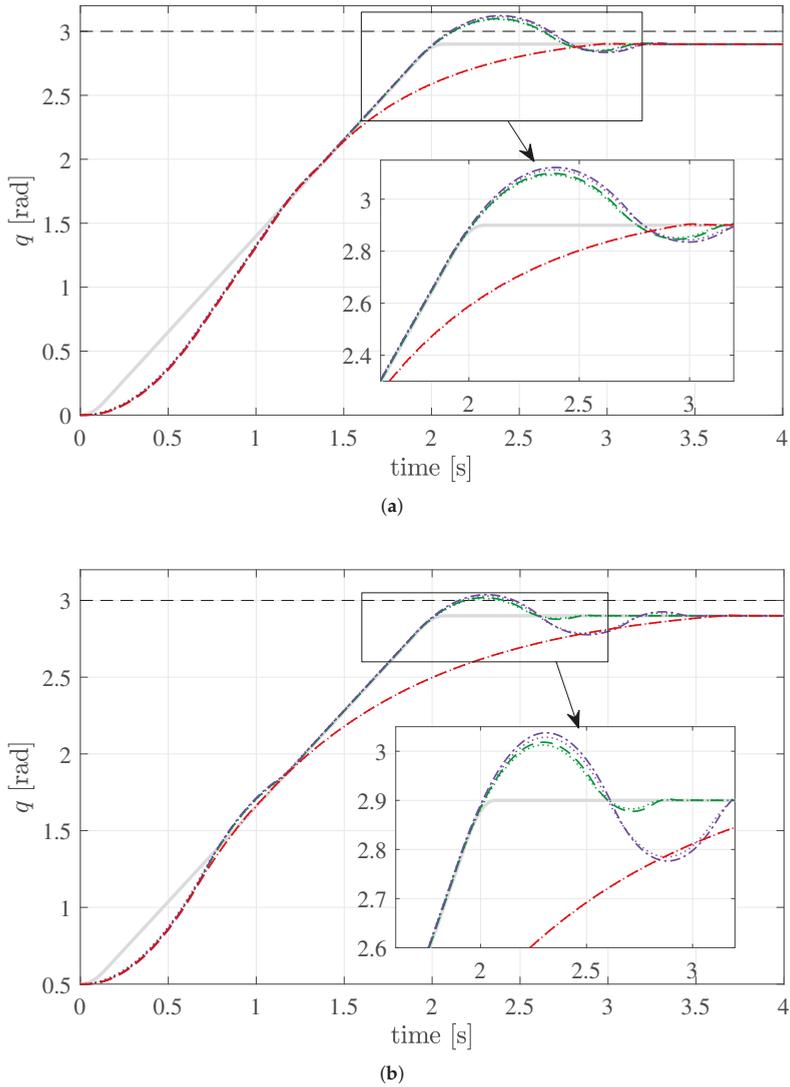


Figure 8. Position of (a) Joint 1; and (b) Joint 3 for $T_{\text{tot}} = 2.07$ s: Reference trajectory (gray solid line). Output of MPC given as reference signal to the low-level controller: MPC without position bounds (green dotted line); MPC with box position bounds (purple dotted line); and MPC with linear viability inequality (Equation (8)) (red dotted line). Measured position: MPC without position bounds (green dashed line); MPC with box position bounds (purple dash-dot line); and MPC with linear viability inequality (Equation (8)) (red dashed line).

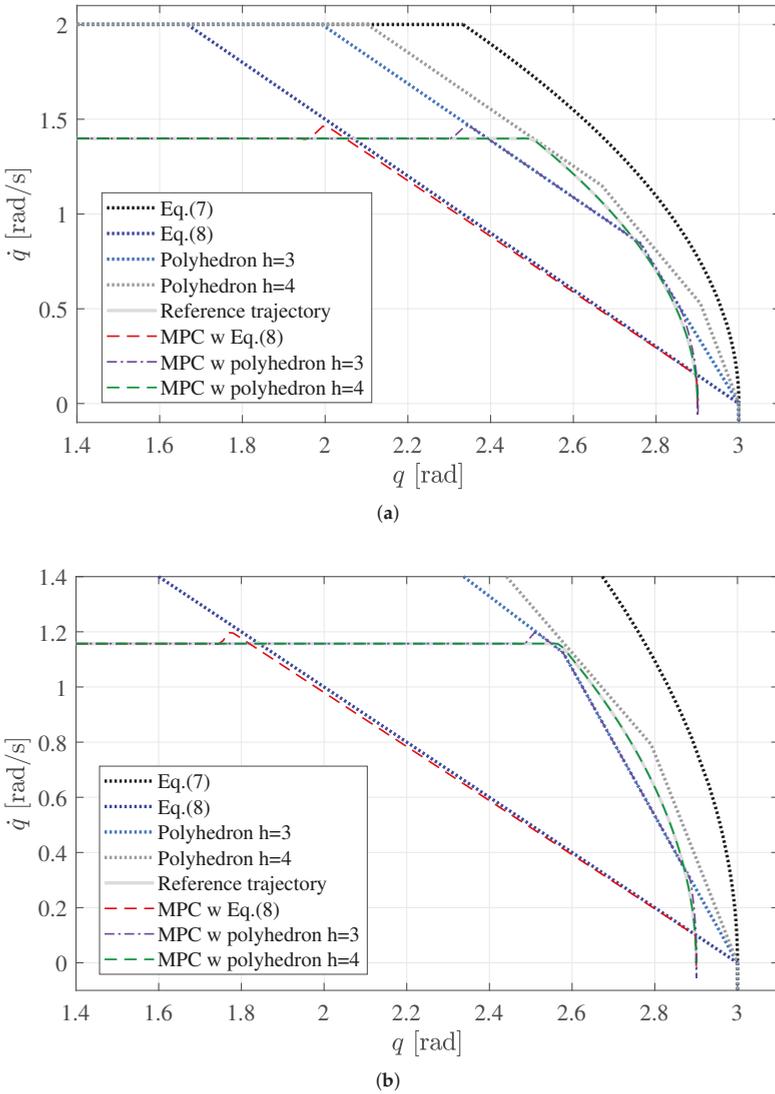


Figure 9. State trajectory in the (q, \dot{q}) -plane for (a) Joint 1 and (b) Joint 3 for $T_{tot} = 2.65$ s: Reference trajectory (gray solid line); MPC with linear viability inequality (Equation (8)) (red dashed line); MPC implementing the viable polyhedron with $h = 3$ (purple dash-dot line); and MPC implementing the viable polyhedron with $h = 4$ (dashed green). Viable sets: Maximal (black dotted line); Linear viability inequality (Equation (8)) (blue dotted line); Polyhedron with $h = 3$ (light blue dotted line); and Polyhedron with $h = 4$ (gray dotted line).

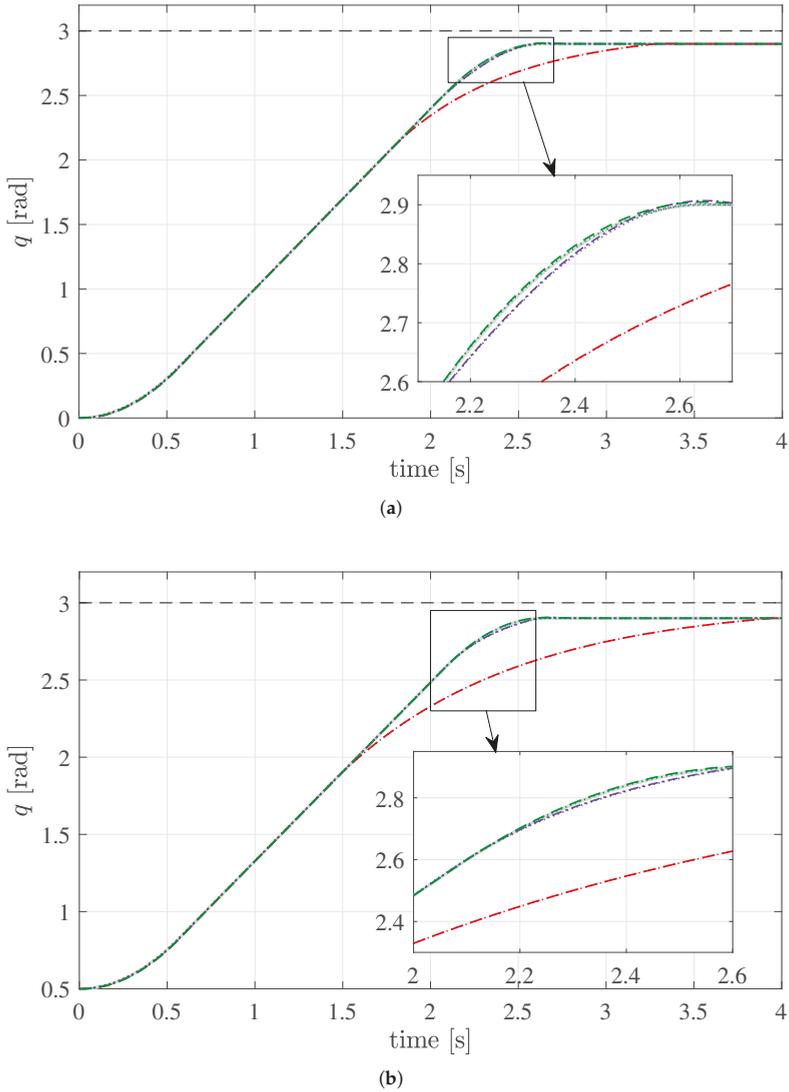


Figure 10. Position of (a) Joint 1 and (b) Joint 3 for $T_{tot} = 2.65$ s. Reference trajectory (gray solid line). Output of MPC given as reference signal to the low-level controller: MPC with linear viability inequality (Equation (8)) (red dotted line); MPC implementing the viable polyhedron with $h = 3$ (purple dotted line); and MPC implementing the viable polyhedron with $h = 4$ (green dotted line). Measured position: MPC with linear viability inequality (Equation (8)) (red dashed line); MPC implementing the viable polyhedron with $h = 3$ (purple dash-dot line); and MPC implementing the viable polyhedron with $h = 4$ (green dashed line).

6. Conclusions

In this paper, we have proposed a method to compute the maximal viable polyhedron for a robot manipulator with bounded joint positions, velocities, and accelerations. In the proposed approach, each joint is considered separately and two optimization problems are devised: one for the upper bounds and one for the lower ones. Given its number of sides, the resulting polyhedron maximizes the area of the admissible position/velocity region. Moreover, the set is proven to be viable and convex. In this way, it can be easily implemented as linear constraints in optimization-based control methods, such as linear MPC. Numerical results demonstrate that the percentage of area covered by the polyhedrons increases as the number of sides grows. This allows the controller to exploit a larger admissible state region and, thus, gives it a broader margin of maneuver in the case the desired motion drives the robot in proximity of the viability boundary. In this case, a better task following can be achieved, as demonstrated by means of experimental results on a six-degrees-of-freedom manipulator.

Author Contributions: M.F. devised the method and wrote the paper; M.B. and N.P. conceived and performed the experiments; and A.V. supervised the whole research work.

Funding: The research leading to these results was partially funded by the European Union H2020-CS2-738039-Eureca: Enhanced Human Robot cooperation in Cabin Assembly tasks.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Verscheure, D.; Demeulenaere, B.; Swevers, J.; Schutter, J.D.; Diehl, M. Time-Optimal Path Tracking for Robots: A Convex Optimization Approach. *IEEE Trans. Autom. Control* **2009**, *54*, 2318–2327. [[CrossRef](#)]
2. Riazi, S.; Wigström, O.; Bengtsson, K.; Lennartson, B. Energy and Peak Power Optimization of Time-Bounded Robot Trajectories. *IEEE Trans. Autom. Sci. Eng.* **2017**, *14*, 646–657. [[CrossRef](#)]
3. Kazerounian, K.; Wang, Z. Global versus local optimization in redundancy resolution of robotic manipulators. *Int. J. Robot. Res.* **1988**, *7*, 3–12. [[CrossRef](#)]
4. Faroni, M.; Beschi, M.; Visioli, A.; Molinari Tosatti, L. A global approach to manipulability optimisation for a dual-arm manipulator. In Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation, Berlin, Germany, 6–9 September 2016; pp. 1–6.
5. Lee, J.; Chang, P.H. Redundancy resolution for dual-arm robots inspired by human asymmetric bimanual action: Formulation and experiments. In Proceedings of the IEEE International Conference on Robotics and Automation, Seattle, WA, USA, 26–30 May 2015; pp. 6058–6065.
6. Liegeois, A. Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms. *IEEE Trans. Syst. Man Cybern.* **1977**, *7*, 868–871.
7. Cheng, F.T.; Chen, T.H.; Sun, Y.Y. Resolving manipulator redundancy under inequality constraints. *IEEE Trans. Robot. Autom.* **1994**, *10*, 65–71. [[CrossRef](#)]
8. Kanoun, O.; Lamiroux, F.; Wieber, P.B. Kinematic control of redundant manipulators: Generalizing the task priority framework to inequality tasks. *IEEE Trans. Robot.* **2011**, *27*, 785–792. [[CrossRef](#)]
9. Escande, A.; Mansard, N.; Wieber, P.B. Hierarchical quadratic programming: Fast online humanoid-robot motion generation. *Int. J. Robot. Res.* **2014**, *33*, 1006–1028. [[CrossRef](#)]
10. Liu, M.; Tan, Y.; Padois, V. Generalized hierarchical control. *Auton. Robots* **2016**, *40*, 17–31. [[CrossRef](#)]
11. Rocchi, A.; Hoffman, E.M.; Caldwell, D.G.; Tsagarakis, N.G. Opensot: A whole-body control library for the compliant humanoid robot Coman. In Proceedings of the IEEE International Conference on Robotics and Automation, Seattle, WA, USA, 26–30 May 2015; pp. 6248–6253.
12. Siciliano, B.; Slotine, J.J.E. A general framework for managing multiple tasks in highly redundant robotic systems. In Proceedings of the International Conference on Advanced Robotics, Pisa, Italy, 19–22 June 1991; pp. 1211–1216.
13. Flacco, F.; De Luca, A. Discrete-time redundancy resolution at the velocity level with acceleration/torque optimization properties. *Robot. Auton. Syst.* **2015**, *70*, 191–201. [[CrossRef](#)]
14. Flacco, F.; De Luca, A.; Khatib, O. Control of Redundant Robots Under Hard Joint Constraints: Saturation in the Null Space. *IEEE Trans. Robot.* **2015**, *31*, 637–654. [[CrossRef](#)]

15. Faroni, M.; Beschi, M.; Visioli, A.; Molinari Tosatti, L. A Predictive Approach to Redundancy Resolution for Robot Manipulators. In Proceedings of the IFAC World Congress, Toulouse, France, 9–14 July 2017; pp. 8975–8980.
16. Faroni, M.; Beschi, M.; Visioli, A. Predictive Inverse Kinematics for Redundant Manipulators: Evaluation in Re-Planning Scenarios. In Proceedings of the IFAC Symposium on Robot Control, Budapest, Hungary, 27–30 August 2018.
17. Buizza Avanzini, G.; Zanchettin, A.M.; Rocco, P. Reactive Constrained Model Predictive Control for Redundant Mobile Manipulators. In *Intelligent Autonomous Systems 13, Advances in Intelligent Systems and Computing*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 1301–1314.
18. Ghazaei Ardakani, M.M.; Olofsson, N.; Robertsson, A.; Johansson, R. Real-Time Trajectory Generation using Model Predictive Control. In Proceedings of the IEEE International Conference on Automation Science and Engineering, Gothenburg, Sweden, 24–28 August 2015; pp. 942–948.
19. Ferrara, A.; Incremona, G.P.; Magni, L. A robust MPC/ISM hierarchical multi-loop control scheme for robot manipulators. In Proceedings of the Annual Conference on Decision and Control, Florence, Italy, 10–13 December 2013; pp. 3560–3565.
20. Blanchini, F. Set invariance in control. *Automatica* **1999**, *35*, 1747–1767. [[CrossRef](#)]
21. Saab, L.; Ramos, O.E.; Keith, F.; Mansard, N.; Soueres, P.; Fourquet, J.Y. Dynamic whole-body motion generation under rigid contacts and other unilateral constraints. *IEEE Trans. Robot.* **2013**, *29*, 346–362. [[CrossRef](#)]
22. Park, K.C.; Chang, P.H.; Kim, S.H. The enhanced compact QP method for redundant manipulators using practical inequality constraints. In Proceedings of the IEEE International Conference on Robotics and Automation, Leuven, Belgium, 20 May 1998; pp. 107–114.
23. Kanehiro, F.; Morisawa, M.; Suleiman, W.; Kaneko, K.; Yoshida, E. Integrating geometric constraints into reactive leg motion generation. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; pp. 4069–4076.
24. Wensing, P.M.; Orin, D.E. Generation of dynamic humanoid behaviors through task-space control with conic optimization. In Proceedings of the IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 3103–3109.
25. Guarino Lo Bianco, C.; Gerelli, O. Online trajectory scaling for manipulators subject to high-order kinematic and dynamic constraints. *IEEE Trans. Robot.* **2011**, *27*, 1144–1152. [[CrossRef](#)]
26. Scheint, M.; Wolff, J.; Buss, M. Invariance control in robotic applications: Trajectory supervision and haptic rendering. In Proceedings of the IEEE American Control Conference, Seattle, WA, USA, 11–13 June 2008; pp. 1436–1442.
27. Zanchettin, A.M.; Rocco, P. Motion planning for robotic manipulators using robust constrained control. *Control Eng. Pract.* **2017**, *59*, 127–136. [[CrossRef](#)]
28. Del Prete, A. Joint Position and Velocity Bounds in Discrete-Time Acceleration/Torque Control of Robot Manipulators. *IEEE Robot. Autom. Lett.* **2018**, *3*, 281–288. [[CrossRef](#)]
29. Faroni, M.; Beschi, M.; Pedrocchi, N.; Visioli, A. Predictive Inverse Kinematics for Redundant Manipulators with Task Scaling and Kinematic Constraints. *IEEE Trans. Robot.* **2018**, submitted.
30. Gerelli, O.; Guarino Lo Bianco, C. Nonlinear variable structure filter for the online trajectory scaling. *IEEE Trans. Ind. Electron.* **2009**, *56*, 3921–3930. [[CrossRef](#)]
31. Wang, L. *Model Predictive Control System Design and Implementation Using MATLAB®*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2009.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Article

Motion Investigation of a Snake Robot with Different Scale Geometry and Coefficient of Friction

Naim Md Lutful Huq, Md Raisuddin Khan *, Amir Akramin Shafie, Md Masum Billah and Syed Masrur Ahmmad

Department of Mechatronics Engineering, International Islamic University Malaysia, Selangor 53100, Malaysia; lutful@gmail.com (N.M.L.H.); aashafie@iium.edu.my (A.A.S.); masum.uia@gmail.com (M.M.B.); bdmasarur91@gmail.com (S.M.A.)

* Correspondence: raisuddin@iium.edu.my; Tel.: +6-03-6196-4575

Received: 1 March 2018; Accepted: 19 April 2018; Published: 28 April 2018

Abstract: Most snakes in nature have scales at their ventral sides. The anisotropic frictional coefficient of the ventral side of the snakes, as well as snake robots, is considered to be responsible for their serpentine kind of locomotion. However, little work has been done on snake scales so far to make any guidelines for designing snake robots. This paper presents an experimental investigation on the effects of artificial scale geometry on the motion of snake robots that move in a serpentine manner. The motion of a snake robot equipped with artificial scales with different geometries was recorded using a Kinect camera under different speeds of the actuating motors attached to the links of the robot. The results of the investigation showed that the portion of the scales along the central line of the robot did not contribute to the locomotion of the robot, rather, it is the parts of the scales along the lateral edges of the robot that contributed to the motion. It was also found that the lower frictional ratio at low slithering speeds made the snake robot motion unpredictable. The scales with ridges along the direction of the snake body gave better and more stable motion. However, to get the peg effect, the scales needed to have a very high lateral to forward friction ratio, otherwise, significant side slipping occurred, resulting in unpredictable motion.

Keywords: snake robot; snake scale; scale geometry; friction ratio; serpentine motion

1. Introduction

Snakes possess a unique feature of locomotion that no other creature has. Although legs and wheels give very effective and efficient motion, this creature has the unique ability to move through terrains that are almost impossible to travel through by limbs or wheels. Thus, the ability to move through almost all kinds of surfaces puts snakes in a superior position in terms of locomotion. Snakes have four basic types of locomotion: concertina, serpentine, sidewinding, and rectilinear. Among these, the serpentine locomotion is a faster and a very efficient one for normal conditions. Thus, this motion can be proven exceptionally beneficial in snake robot applications such as search and rescue [1,2], firefighting [3,4], assisting in surgery [5,6], surveillance [7,8] exploring unknown territory, and so forth.

The mechanics of serpentine locomotion were first discovered and explained by J. Gray [9] in 1946. According to Gray, the serpentine locomotion of snakes is due to the lateral friction of the scales of a snake that work as supporting pegs. After the ground breaking discovery of J. Gray, Hu et al. [10] carried out several investigations on real snakes and established the theory experimentally. Some more works on biological snakes that were not focused on the analysis or investigation of the forces applied by the snakes took place. Jayne [11,12] was more focused on the muscular activities during the different types of locomotion with extensive work on serpentine and sidewinding locomotion. Jayne provided extraordinary information regarding the optimization of the actuation method and the energy for the future developments of robots. Meanwhile Miller [13] simulated the motion dynamics

of snake locomotion. Later on, Jayne and Davis [14] developed a relationship between the tunnel width and the speed, or, in other words, the movable space available and the speed of the snake using the kinematics of the concertina locomotion of snake. Marvi and his group studied the friction augmentation by the snakes in concertina locomotion [15].

The archive of information on snake robots dates back to the year 1993 when S. Hirose developed the idea of the locomotion of a snake robot [16], encouraging researchers in working consistently to improve and optimize their artificial snakes. For instance, Wang, Osborne, and Alben [17] and Jing and Alben [18] optimized locomotion on an inclined surface. Instead of real snakes, Kyriakopoulos, Migadis, and Sarrigeorgidis [19] worked on the kinematics of snake robots and provided a design and motion planning in 1999. Prautsch and Mita developed a theory for the dynamic position control [20] for snake robots. Khan et al. [21] first developed a scale based snake trying to mimic the serpentine motion. Later on, they put forward their previous work by investigating the snake robot moving with the serpentine method of locomotion [22]. Apart from that, Marvi et al. [23] presented a snake inspired robot segment based on concertina locomotion in 2011. It was capable of changing the angle of attack of its scales with the change in the slope angle while climbing uphill. They also worked on sidewinding with minimal slips for both the snake and the snake robot on sandy slopes [24]. Varesis, Diamantopoulos, and Tzes [25] conducted an experiment on robots with serpentine locomotion moving through an inclined surface [23]. However, none of the works on artificial snake scales considered the geometry of the scales to investigate the motion of the snake robots. As a continuation of the above efforts, this paper presents the effects of the scale geometry and the friction factor on the locomotion of a snake robot.

This paper is organized as follows. The experimental setup and data acquisition system are described in Section 2, while the effect of the surface properties on motion and the effect of the surface on the directional stability are presented in Sections 3 and 4, respectively. Section 5 draws the conclusions of the paper.

2. The Experimental SETUP and Data Acquisition

A snake robot made of nine links, eight servo-motors, and a microcontroller, as shown in Figure 1, was used in this research. The microcontroller was programmed for controlling the motion of the servo-motors and propelling the snake in a serpentine motion. The snake's scales were attached at the ventral side of the snake robot, while markers were attached on the dorsal side to keep track of the motion of the robot.



Figure 1. The snake robot with markers on the dorsal side.

Four different types of scales were used to investigate the effect of scale geometry and the coefficient of friction on the motion of the snake robot. Photographs and the geometry of the scales are shown in Figure 2a–d. On an average, the span of the scales was 20 mm, except for the scale in Figure 2d. The scales shown in Figure 2a were fabricated from Acrylonitrile Butadiene Styrene (ABS) using a 3D printer while the scales shown in Figure 2b,c were made of ordinary plastic shells and externally threaded cylinder (Cylinder_1), splitting the cylinders into two halves. In Figure 2d, a scale

made from an externally threaded cylinder with a lower friction factor than Cylinder_1 (we call it Cylinder_2) extending between the left and right edge of a link is shown.

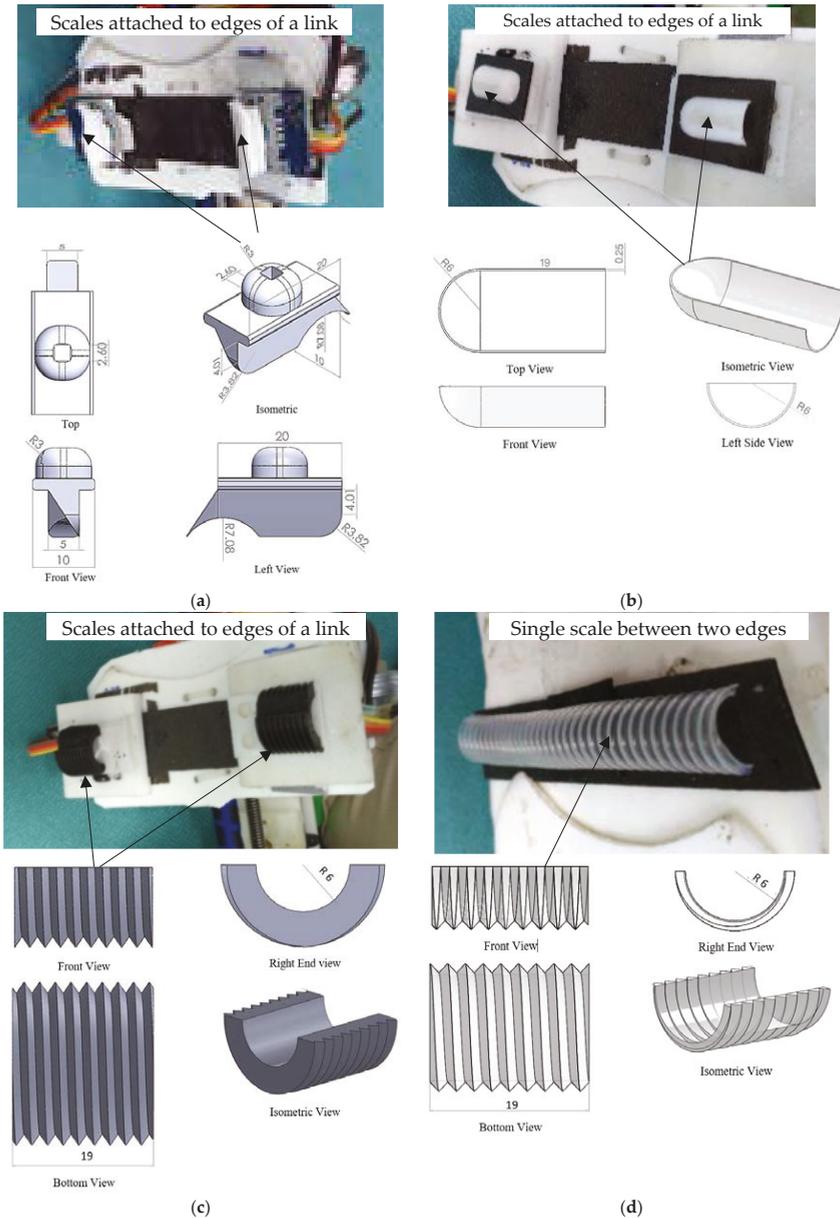


Figure 2. The photographs of the snake scales attached to the links of the snake robot and orthographic views of the scales (a) Designed scale; (b) Unthreaded scale (half-shell); (c) Threaded scale (Cylinder_1); (d) Threaded scale (Cylinder_2).

As the goal of this research is to investigate the effect of scale geometry on the motion of snake robots, the robot was equipped with one type of scale at a time. One pair of scales were attached at the lateral edges of each link of the robot. The snake robot was run on three different floor surfaces, namely, Surface 1 (Cloth), Surface 2 (artificial lather 1), and Surface 3 (artificial leather 2). The static friction coefficients between the snake scales and the surfaces are presented in Table 1. One of the main features of the snake scales was the anisotropic friction coefficients along the tangential and normal directions. Frictional anisotropy were achieved through the design of the snake scales of the snake robot.

Table 1. The static friction coefficients between the scales and different surfaces.

Scale/Surface	Fiction Coefficients								
	Surface 1			Surface 2			Surface 3		
	Tangential	Normal	Friction Ratio	Tangential	Normal	Friction Ratio	Tangential	Normal	Friction Ratio
Designed Scale	-	-	-	0.244	0.303	1.243	0.273	0.318	1.166
Half Shell	0.265	0.293	1.103	0.273	0.318	1.166	0.303	0.382	1.261
Threaded cylinder_1	0.310	0.399	1.286	0.187	0.273	1.460	0.201	0.288	1.433
Threaded cylinder_2	0.270	0.310	1.150	0.201	0.288	1.433	0.303	0.366	1.207

Figure 3 illustrates the robot orientation and the positions of the scales L1 through R4. Where 'L' denotes the left side and 'R' denotes the right side looking from the top of the robot. A Microsoft Kinect XBOX was attached on a frame above the plane of the motion to capture the motion and the shape of the robot during its motion. A photograph of the experimental setup is shown in Figure 4.

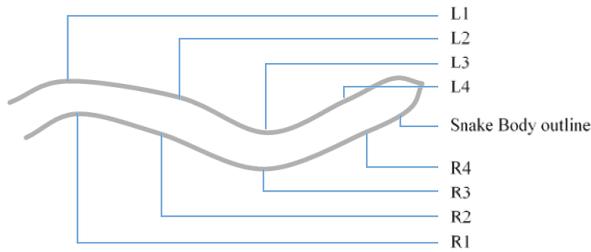


Figure 3. The snake’s shape and scale placement (L and R stands for left and Right respectively).

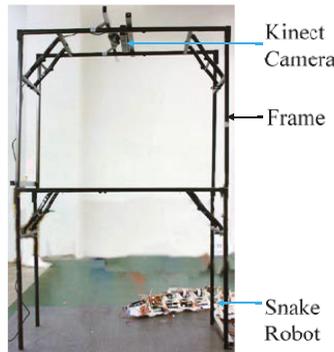


Figure 4. The experimental setup for the snake’s motion capture.

3. The Effect of the Surface Properties on Motion

The otion data of the snake robot are presented in Table 2.

Table 2. The motion characteristics of the snake robot with different scales on different surfaces at different motor speeds.

Scale and Surface	High			Medium			Low					
	Cycle/s	Dis/Cy (cm)	V (cm/s)	θ (°)	Cycle/s	Dis/Cy (cm)	V (cm/s)	θ (°)	Cycle/s	Dis/Cy (cm)	V (cm/s)	θ (°)
Designed Scale on Surface 1	X	3.667	1.770	2.471	X	3.066	1.338	0.620	X	2.521	0.878	X
Designed Scale on Surface 2	0.483	6.127	2.470	5.730	0.424	4.000	1.697	4.158	0.428	3.880	1.662	1.210
Designed Scale on Surface 3	0.403	1.368	0.622	2.174	0.429	1.986	0.853	5.265	0.306	1.997	0.611	6.457
Half Shell on Surface 1	0.455	2.678	1.218	2.381	0.429	2.433	1.043	1.673	0.307	2.163	0.663	2.710
Half Shell on Surface 2	0.455	2.710	1.234	0.762	0.427	2.184	0.932	5.508	0.306	1.087	0.333	1.035
Half Shell on Surface 3	0.457	6.381	2.918	5.638	0.429	5.607	2.403	8.607	0.333	4.455	1.483	0.533
Thread Cylinder_1 on Surface 1	0.479	3.561	1.707	0.634	0.429	3.041	1.306	1.719	0.333	3.300	1.099	5.330
Thread Cylinder_1 on Surface 2	0.484	5.298	2.567	6.437	0.429	4.245	1.819	5.989	0.332	3.366	1.116	2.178
Thread Cylinder_2 on Surface 1	0.483	5.046	2.435	2.411	0.429	3.614	1.549	7.666	0.305	3.394	1.036	4.181
Thread Cylinder_2 on Surface 2	0.483	3.656	1.765	0.816	0.428	3.112	1.333	4.098	0.349	3.079	1.076	5.207
Thread Cylinder_2 on Surface 3	0.482	4.139	1.996	5.600	0.428	2.358	1.008	0.818	0.349	2.400	0.837	3.757
Thread Cylinder_2 at Center on Surface 2	0.484	0.622	0.301	5.398	-	-	-	-	-	-	-	-
Thread Cylinder_2 at Center on Surface 3	-	-	-	-	-	-	-	-	-	-	-	-

where, Cycle/s = the fraction of a complete cycle completed by the robot; Dis/Cy = the distance traveled by the robot in one cycle; V = the average velocity of the robot; θ = the angle between the initial and final orientations of the robot; High, medium, slow = the cycle speed of the robot.

Table 2 compares four motion elements like the number of cycles per second (seven cycles were taken for each observation), the distance traveled by the robot per cycle, the average velocity of the robot, and the deviation from the initial axis or orientation of the robot after each cycle corresponding to three different motor speeds, namely, high speed, medium speed, and low speed. The three types of motions (high, medium, and low) of the robot were the cycle speed of the sine wave those were created by controlling the speed of the 8 actuating motors. The symbol 'X' in the first row of this table means no data, that is, the robot was unable to move without damaging itself on surface 1 with the designed scale as the scale got stuck in the surface. The velocity of the robot should depend on the frequency of the sine wave of the snake robot. This is evident when the velocities of the robot under three types of motion are compared under the three types of motion. The highest velocity (2.918 cm/s) was achieved in the case of the cylinder_1 on surface 1 corresponding to the high speed of the motors. This surface combination also produces the highest velocity corresponding to the medium and low speed of the motors. The angular deviation of the robot θ from its initial orientation, in the case of the high and medium speeds, are 5.638° and 8.607° , respectively, which are quite high compared to the 0.533° that happened in the case of low speed. The friction ratio for this surface combination is lower than that of the cylinder_1 on surface 2, however, the coefficient of friction along the normal direction for cylinder_1 on surface 1 is the highest ($\mu = 0.399$). Thus, it may be concluded that the higher value of the coefficient of friction along the normal direction plays a significant role in increasing the speed of the robot.

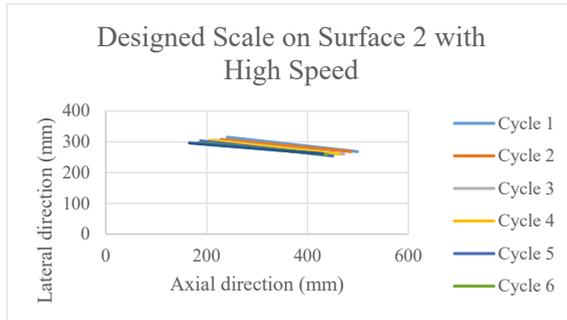
Among the four types of scales and the three types of surfaces, the friction coefficient of surface 2 gives the maximum possible friction ratios between the tangential and normal friction coefficient (Table 1), while the threaded cylinder_1 on surface 2 gives the highest friction ratio among them all. The comparison of the robot velocities of the designed scale on surface 2 and that of cylinder_1 on surface 2 showed very similar results. The robot velocities achieved in these cases are almost half that of cylinder_1 on surface 1 for all types of motor speed. However, the angular deviations are significantly low except for the low speed of the motors. The half shell on surface 3 has the highest movement with better stability (lowest angular deviation) than others, while the smooth half shell gives the minimum distance per cycle. It is noticed that the half shell had the highest friction ratio on surface 3. The last piece of data is a very important one, this data is for the threaded cylinder_2 on surface 2 (the one with the highest friction ratio) where the scales are placed at the mid surface (centerline) of the links instead of at the edges of the links. It shows that it does not move even a centimeter, yet, the same scale on the same surface gave the best output among all other combinations. Thus, it can be concluded that snake scales need to be placed at the edges of the links for the better motion of the robot, and it is unnecessary to place the snake scales along the centerline of the links.

4. The Effect of the Surface on Directional Stability

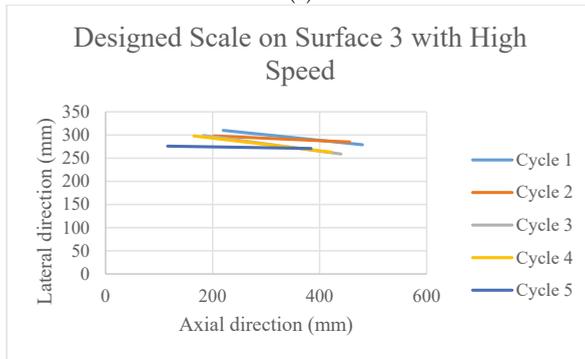
The poses of the robot after each cycle of motion are presented in Figure 5. Each line in these figures is the connecting line between the head and tail of the robot after each cycle of actuation. Thus, these lines are presenting the change in position as well as change in the orientation with respect to the x - y axis. The initial orientation of the robot was parallel to the x axis.

It is observed in Figure 5a–l that the surfaces and the scales have profound effects on the directional stability and motion for the movement of the robot.

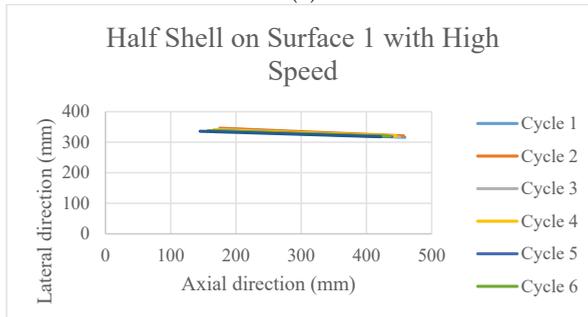
Due to practical reason it was not possible to run the robot on surface 1 using the designed scale, thus, the data is absent. However, some important phenomena can be observed from the rest of the figures. All the scales that have threads on it (other than Figure 5c–e,l) are superior in terms of stability in the motion direction, as well as the distance covered. On top of that, when they move on the surface 1 (cloth) (Figure 5f,i) which has an anisotropic friction property within itself, cover higher distance. On the other hand, the half shell (Figure 5c–e) did not have much of a promising outcome in terms of the directional stability and velocity. On surface 2, the motion direction is better maintained than that on surface 3.



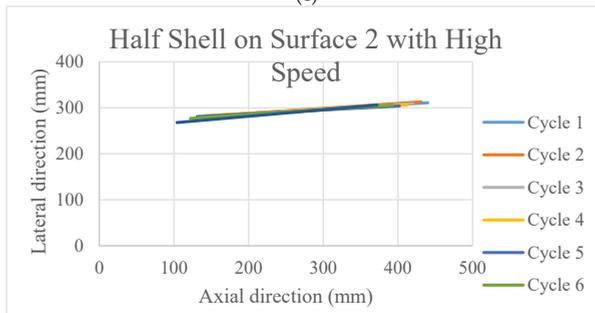
(a)



(b)

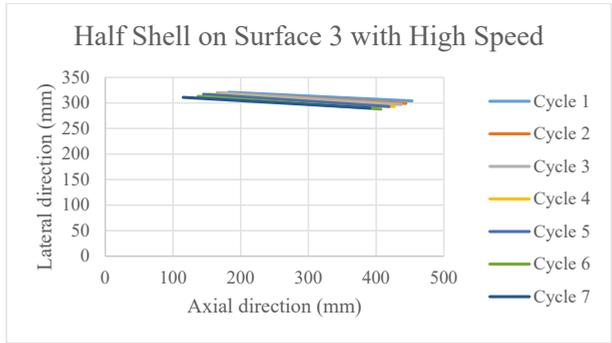


(c)

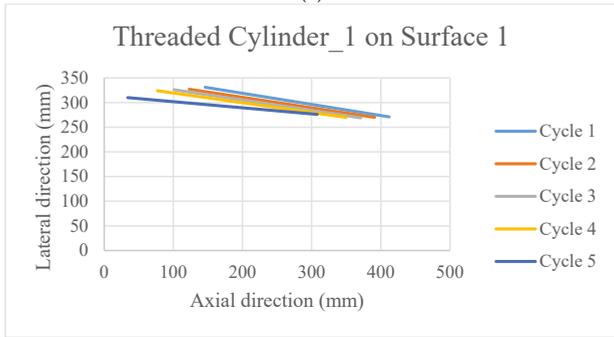


(d)

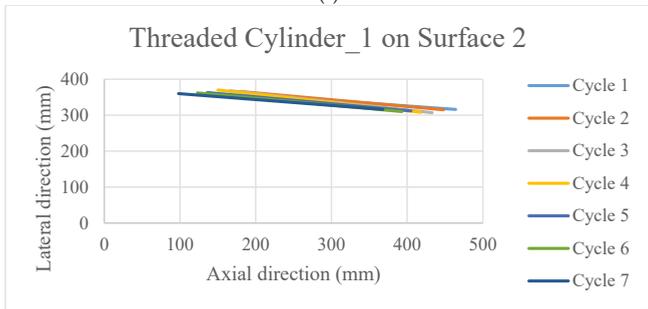
Figure 5. Cont.



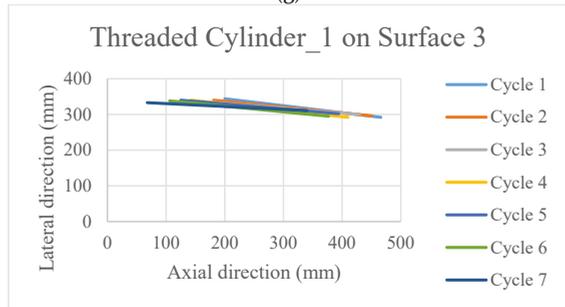
(e)



(f)



(g)



(h)

Figure 5. Cont.

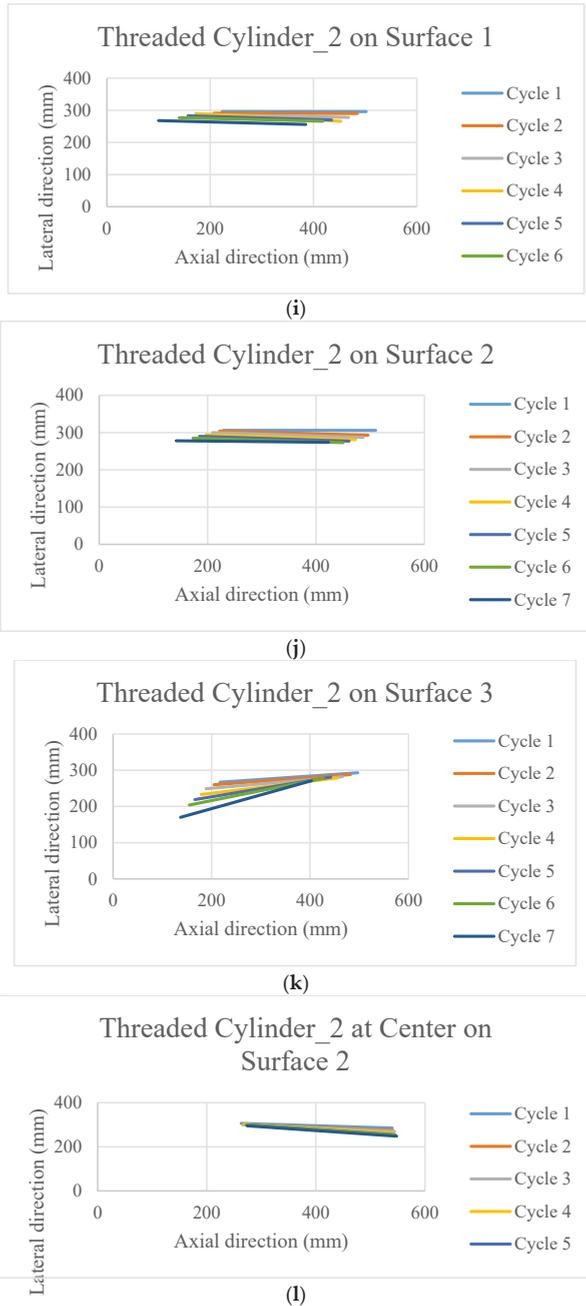


Figure 5. The poses of the snake robot on the different surfaces at the end of each cycle of actuation.

All the orientations of the snake robot after the last cycle of actuation found in Figure 5a–l are presented in Figure 6. A very important finding that is observed in the last set of data in Figure 5l is shown circled in red in Figure 6. This is the situation when the scales were placed in the middle of the

snake body leaving the edges of the link free and floating. There is no significant movement as well as no control of direction in this case. In fact, the robot moved in the backward direction instead of moving forward, however, the same threaded cylinder 2 with the scale at the outer edges of the links gave better robot motion.

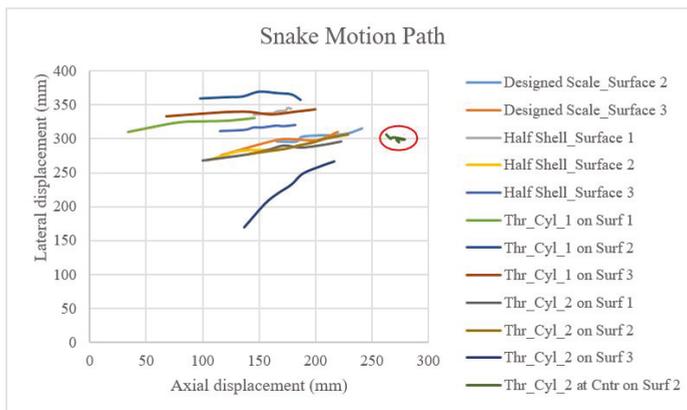


Figure 6. The comparison of the motion paths of the snake head on different surfaces.

5. Conclusions

This work makes an effort to discover the relationship between the different variables of the snake scales, like friction ratio, scale geometry, and the span of the scales beneath a snake robot that undergoes serpentine motion. The comparison of the motion shows that scales with a higher friction ratio set at the outer edge of the snake robot helps to generate better and more stable motion. On the other hand, scales only at the center of the belly, irrespective of the friction ratio, does not help the robot to generate any good motion or any forward motion at all. Scales have a ridge like texture on it along the direction of the snake body, which, when placed at the outer edge of the belly, helps generate stable and faster motion. It is predictable from the above experiments that the higher friction ratio will help achieve higher and better motion. Motions with the actuating motors at lower cycle speeds with a scale-surface pair having a lower friction ratio are very unpredictable in nature.

Author Contributions: Writing –Original Draft Preparation, Naim Md. Lutful Huq; Supervision, Funding Acquisition & Project administration, Md. Raisuddin Khan and Amir Akramin Shafie; Software, Md. Masum Billah; Methodology, Syed Masrur Ahmmad.

Funding: This research was funded by Ministry of Higher Education (MOHE) grant number FRGS13-072-0313.

Acknowledgments: The authors profoundly acknowledge the Ministry of Higher Education (MOE), Malaysia for funding this research through the Fundamental Research Grant Scheme (FRGS).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Casper, J.; Murphy, R.R. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **2003**, *33*, 367–385. [[CrossRef](#)] [[PubMed](#)]
2. Wolf, A.; Brown, H.B.; Casciola, R.; Costa, A.; Schwerin, M.; Shamas, E.; Choset, H. A mobile hyper redundant mechanism for search and rescue tasks. In Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, NV, USA, 27–31 October 2003; Volume 3, pp. 2889–2895.
3. Liljeback, P.; Stavdahl, O.; Beitnes, A. SnakeFighter-development of a water hydraulic firefighting snake robot. In Proceedings of the 9th International Conference on Control, Automation, Robotics and Vision, ICARCV'06, Singapore, 5–8 December 2006; pp. 1–6.

4. Cai, L.; Zhang, R. Design and Research of Intelligent Fire-Fighting Robot. *Adv. Mater. Res.* **2013**, *823*, 358–362. [[CrossRef](#)]
5. Ho, C.; Chen, M.; Lien, C. Machine Vision-Based Intelligent Fire Fighting Robot. *Key Eng. Mater.* **2010**, *450*, 312–315. [[CrossRef](#)]
6. Degani, A.; Choset, H.; Wolf, A.; Zenati, M.A. Highly articulated robotic probe for minimally invasive surgery. In Proceedings of the 2006 IEEE International Conference on Robotics and Automation, ICRA 2006, Orlando, FL, USA, 15–19 May 2006; pp. 4167–4172.
7. Tully, S.; Kantor, G.; Zenati, M.A.; Choset, H. Shape estimation for image-guided surgery with a highly articulated snake robot. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), San Francisco, CA, USA, 25–30 September 2011; pp. 1353–1358.
8. Granosik, G.; Hansen, M.G.; Borenstein, J. The OmniTread serpentine robot for industrial inspection and surveillance. *Ind. Robot* **2005**, *32*, 139–148. [[CrossRef](#)]
9. Granosik, G.; Borenstein, J.; Hansen, M.G. Serpentine robots for industrial inspection and surveillance. In *Industrial Robotics: Programming, Simulation and Applications*; I-Tech Publ.: London, UK, 2007; pp. 633–662.
10. Gray, J. The mechanism of locomotion in snakes. *J. Exp. Biol.* **1946**, *23*, 101–120. [[PubMed](#)]
11. Hu, D.L.; Nirody, J.; Scott, T.; Shelley, M.J. The mechanics of slithering locomotion. *Proc. Natl. Acad. Sci. USA* **2009**, *106*, 10081–10085. [[CrossRef](#)] [[PubMed](#)]
12. Jayne, B.C. Muscular mechanisms of snake locomotion: An electromyographic study of the sidewinding and concertina modes of *Crotalus cerastes*, *Nerodia fasciata* and *Elapheobsoleta*. *J. Exp. Biol.* **1988**, *140*, 1–33. [[PubMed](#)]
13. Jayne, B.C. Muscular mechanisms of snake locomotion: An electromyographic study of lateral undulation of the Florida banded water snake (*Nerodia fasciata*) and the yellow rat snake (*Elapheobsoleta*). *J. Morphol.* **1988**, *197*, 159–181. [[CrossRef](#)] [[PubMed](#)]
14. Miller, G.S. The motion dynamics of snakes and worms. In Proceedings of the ACM Siggraph Computer Graphics and Interactive Techniques, Atlanta, GA, USA, 1–5 August 1988; Volume 22, pp. 169–173.
15. Jayne, B.C.; Davis, J.D. Kinematics and performance capacity for the concertina locomotion of a snake (*Coluber constrictor*). *J. Exp. Biol.* **1991**, *156*, 539–556.
16. Marvi, H.; Hu, D.L. Friction enhancement in concertina locomotion of snakes. *J. R. Soc. Interface* **2012**, *9*, 3067–3080. [[CrossRef](#)] [[PubMed](#)]
17. Hirose, S. *Biologically Inspired Robots*; Oxford University Press: Oxford, UK, 1993.
18. Wang, X.; Osborne, M.T.; Alben, S. Optimizing snake locomotion on an inclined plane. *Phys. Rev. E* **2014**, *89*. [[CrossRef](#)] [[PubMed](#)]
19. Jing, F.; Alben, S. Optimization of two-and three-link snakelike locomotion. *Phys. Rev. E* **2013**, *87*. [[CrossRef](#)] [[PubMed](#)]
20. Kyriakopoulos, K.J.; Migadis, G.; Sarrigeorgidis, K. The NTUA snake: Design, planar kinematics, and motion planning. *J. Robot. Syst.* **1999**, *16*, 37–72. [[CrossRef](#)]
21. Prautsch, P.; Mita, T. Control and analysis of the gait of snake robots. In Proceedings of the 1999 IEEE International Conference on Control Applications, Kohala Coast, HI, USA, 22–27 August 1999; Volume 1, pp. 502–507.
22. Khan, R.; Watanabe, M.; Shafie, A.A. Kinematics model of snake robot considering snake scale. *Am. J. Appl. Sci.* **2010**, *7*, 669. [[CrossRef](#)]
23. Marvi, H.; Gong, C.; Gravish, N.; Astley, H.; Travers, M.; Hatton, R.L.; Mendelson, J.R., III; Choset, H.; Hu, D.L.; Goldman, D.I. Sidewinding with minimal slip: Snake and robot ascent of sandy slopes. *Science* **2014**, *346*, 224–229. [[CrossRef](#)] [[PubMed](#)]
24. Khan, R.; Billah, M.; Huq, N.; Helmi, M.; Ahmmed, S. Investigation on Data Extraction Trends for Snake Robot. *Procedia Comput. Sci.* **2014**, *42*, 145–152. [[CrossRef](#)]
25. Varesis, O.; Diamantopoulos, C.; Tzes, A. Experimental studies of serpentine motion control of snake robots at inclined planes. In Proceedings of the 2016 24th Mediterranean Conference on Control and Automation (MED), Athens, Greece, 21–24 June 2016; pp. 737–742.



Article

Fully Mechatronical Design of an HIL System for Floating Devices

Hermes Giberti ^{1,*}, Francesco La Mura ^{2,*}, Gabriele Resmini ² and Marco Parmeggiani ²

¹ Dipartimento di Ingegneria Industriale e dell'Informazione, Università degli Studi di Pavia, Via A. Ferrata 5, 27100 Pavia, Italy

² Department of Mechanical Engineering, Politecnico di Milano, 20156 Milano, Italy; gabriele.resmini@mail.polimi.it (G.R.); marco1.parmeggiani@mail.polimi.it (M.P.)

* Correspondence: hermes.giberti@unipv.it (H.G.); francesco.lamura@polimi.it (F.L.M.); Tel.: +39-0382-98-5255 (H.G.)

Received: 22 May 2018; Accepted: 11 July 2018; Published: 20 July 2018

Abstract: Recent simulation developments in Computational Fluid Dynamics (CFD) have widely increased the knowledge of fluid–structure interaction. This has been particularly effective in the research field of floating bodies such as offshore wind turbines and sailboats, where air and sea are involved. Nevertheless, the models used in the CFD analysis require several experimental parameters in order to be completely calibrated and capable of accurately predicting the physical behaviour of the simulated system. To make up for the lack of experimental data, usually wind tunnel and ocean basin tests are carried out. This paper presents a fully mechatronical design of an Hardware In the Loop (HIL) system capable of simulating the effects of the sea on a physical scaled model positioned in a wind tunnel. This system allows one to obtain all the required information to characterize a model subject, and at the same time to assess the effects of the interaction between wind and sea waves. The focus of this work is on a complete overview of the procedural steps to be followed in order to reach a predefined performance.

Keywords: parallel kinematic machine; kinematic optimisation; hardware in the loop; mechatronic design

1. Introduction

Recent simulation developments in Computational Fluid Dynamics (CFD) have widely increased the knowledge of fluid–structure interaction. This has been particularly effective in the research field of floating bodies such as offshore wind turbines and sailboats, where air and sea are involved. Scale model testing of floating structures is a common practice in this research field and help in the development of new concepts and technological solutions, driving the design choices and rapidly answering the scientific questions. Moreover, this study approach allows one to evaluate related problems such as the definition of the control algorithm structures and quantification of costs and benefits.

Nevertheless, when the effects of wind and wave loads become comparable, the validation of the test models is affected by a scaling issue called Froude–Reynolds conflict, related to the ability to reproduce the effects of wind and wave at the same time in a correct manner. For this reason, hybrid tests both in ocean basins [1] and wind tunnels [2] are the most effective approach used to overcome scaling constraints and exploit separately wind and wave generators.

In this paper, the hardware/software setup developed for hybrid tests in a wind tunnel is presented and fully described. The aim of this work is to provide a complete overview of the design procedures and methodologies required to develop a hardware in the loop device capable of simulating the effects of the sea on a physical scaled model positioned in a wind tunnel. In order to achieve the required performance and in compliance with the dimensional constraints, a fully mechatronical design approach has been used. Moreover, this system is particularly challenging in terms of dynamic

response. For these reasons, it can be taken as an example of integrated design not only for this specific application, but also for a generic simulator. In fact, the design scheme followed and the numeric tools to develop the system sizing and select the commercially available components are of general interest.

Figure 1 summarises the design procedure of the HIL device presented in this paper and called HexaFloat. The latter consists of a six-degree-of-freedom (DOF) parallel manipulator custom designed to move a scaled wind turbine (e.g., 1/75 of the 10 MW DTU reference wind turbine [3]) in a wind tunnel. The manipulator end effector is equipped with a six-degrees of freedom weight scale used to measure the constraint reactions between the robot and the scaled wind turbine (i.e., aerodynamic load effects). A real-time mathematical model generates the reference commands to be followed by the robot according to the dynamic of the floating substructure simulated by the system and the measures acquired from the weight scale (i.e., real-time hydro/structure computations). The design procedure, as shown in Figure 1, is made up of four main steps that for descriptive simplicity are presented in sequence in this paper but should be considered as a cyclic procedure that ends asymptotically towards the result. The first step relates to the kinematic topological definition and its optimisation as a function of the workspace dimension and the dimensional constraints. The second and the third steps are related to the sizing procedures necessary to ensure the static and dynamic performance of the device taking into account both the actuation system and structural components. The last step deals with the definition of the hardware and software control issues.

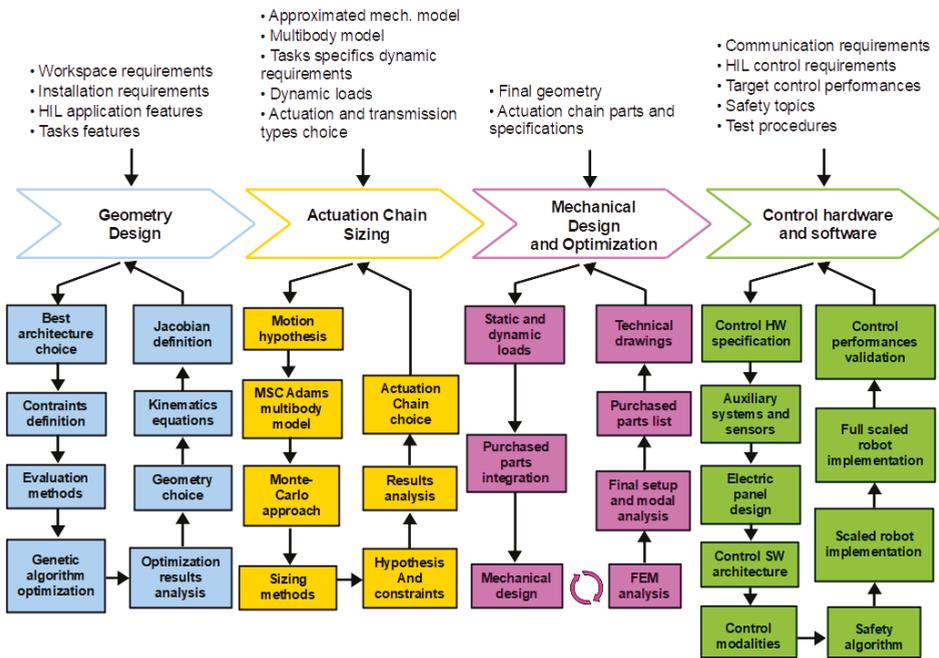


Figure 1. General scheme of the HexaFloat design process.

2. Geometric and Kinematic Design

The starting point of every project is the definition of the operating conditions. Figure 2 summarises the dimensional constraints and shows the position of the desired working volume within the wind tunnel. This position has been chosen in order to optimize the layout of the aerodynamic tests using a scaled turbine with respect to the wind tunnel performance. No commercial solutions suit the

requirements due to the restricted space available to install the device so therefore a custom solution must be realized out of necessity.

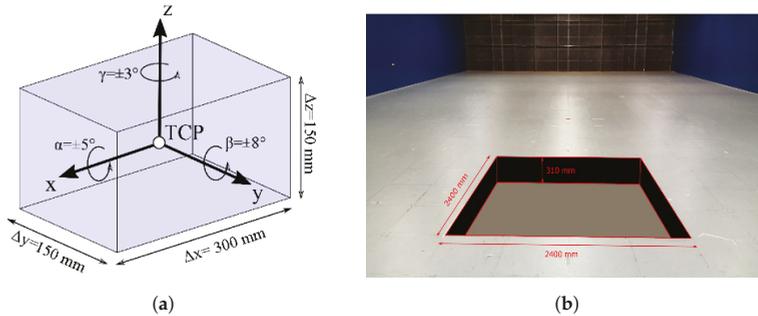


Figure 2. Dimensional constraints: (a) Workspace constraints; (b) Wind tunnel frontal view and dimensions of the space dedicated to the machine.

Among the possible six-DoF manipulator architectures, a parallel kinematic machine (PKM) has been preferred rather than the serial one, due to its high stiffness and high dynamic performance even if this entails an accurate design of the joints in order to have an adequate working space and the absence of backlash. In the first part of the design procedure, the main focus is on the choice of the kinematic topology and the geometric dimensions [2,4]. The most common and widespread six-DOF PKM architectures among which to choose the right topology for this application can be summed up in three families:

- 6-UPS:** manipulators with kinematic chain characterised by a sequence of a universal joint at the base (U), an actuated prismatic joint (P) that changes the length of the link as well as a spherical joint (S) connected to the mobile platform;
- 6-RUS:** manipulators with fixed length links, moved by actuated revolute joints (R) located at the base. The other joints in the kinematic chain are: an intermediate universal joint and a spherical joint connected to the mobile platform;
- 6-PUS:** manipulators with fixed length links. The actuated prismatic joint is generally composed of a slider moving along a rectilinear guide. The link is connected to the slider by means of a universal joint and to the mobile platform by means of a spherical joint.

The most promising solution within the three families is the six-PUS configuration because the actuation system lays on the floor grounded; moreover, the vertical bulk is limited as opposed to the horizontal one. This topology has been selected in order to guarantee high performance in terms of dynamic response and structural properties with the view to reduce the size vertically. Within this category, the two manipulators shown in Figure 3 have been chosen and evaluated [5]. The first one is called Hexaglide, having parallel linear guides and therefore is characterized by symmetry with respect to the longitudinal median plane. The second is called Hexaslide and is characterized by a radial symmetry with respect to the vertical axis due to the radial distribution of the linear guides in the plane. This characteristic leads to a better isotropy of the workspace.

2.1. Kinestostatic Optimization

An optimization process [5] is required in order to synthesize the geometric parameters of the two chosen robot architectures and evaluate which is the best solution. To properly set up the process, it is first necessary to identify the geometric parameters characterizing the two architectures and thereafter a function that mathematically describes the goal to be achieved. Finally, a set of physical constraints affecting the system have to be identified and defined in mathematical form in

order to create the numeric boundaries for the solution. When choosing the number of parameters to be optimized, a trade-off has to be found: if the number of parameters used to characterize a machine is increased, the whole process of optimization would benefit, thus allowing a higher freedom. However, the complexity of the machine increases dramatically. A trade-off needs to be found between global performance and structural modularity, which is critical in order to simplify design and optimization steps.

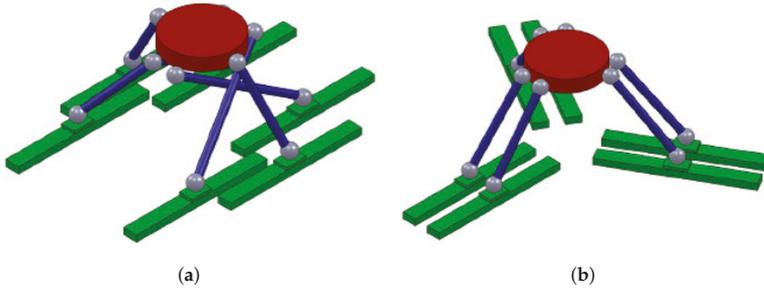


Figure 3. Hexaglide robot (a) and Hexaslide robot (b) simplified geometry scheme.

The Hexaslide architecture can be analysed taking into consideration the following parameters (Figure 4) for the optimization process:

- s : semi-distance between two parallel guides;
- l_i : length of the links;
- R_p : radius of the circumference on which the platform joint centres are located;
- θ_p : semi-angular aperture between the two segments connecting the origin of the Tool Center Point (TCP) reference frame and a couple of platform joints;
- z_{wsd} : height of the centre of the desired workspace.

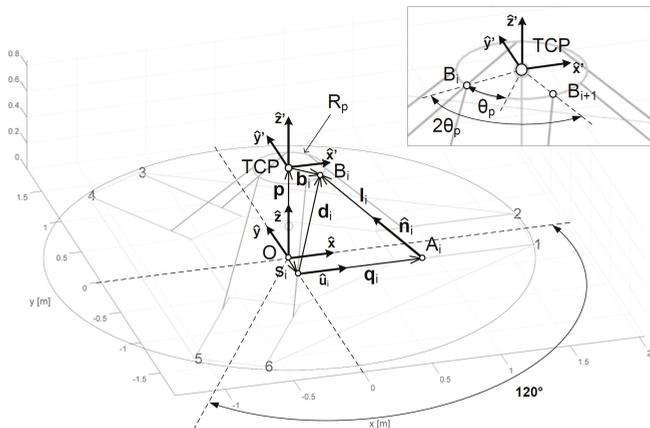


Figure 4. Hexaslide geometric parameters, the Cartesian space is represented in meters.

For this architecture, the same kinematic chain is replicated identically for six times.

The Hexaglide architecture, shown in Figure 5, can be analysed considering a higher number of parameters because three different kinematic chains can be identified in the robot configuration. The parameters to be optimized are:

- $R_{p,1}$: radius of the circumference on which the platform joints 1 and 2 are located;
- $R_{p,2}$: radius of the circumference on which the platform joints 3 and 4 are located;
- $R_{p,3}$: radius of the circumference on which the platform joints 5 and 6 are located;
- $\theta_{p,1}$: semi-angular aperture between the two segments connecting the origin of the TCP reference frame and platform joints 1 and 2;
- $\theta_{p,2}$: semi-angular aperture between the two segments connecting the origin of the TCP reference frame and platform joints 3 and 4;
- $\theta_{p,3}$: semi-angular aperture between the two segments connecting the origin of the TCP reference frame and platform joints 5 and 6;
- t_{01} : vertical distance between TCP and the plane where platform joints 1 and 2 lie;
- t_{02} : vertical distance between TCP and the plane where platform joints 3 and 4 lie;
- t_{03} : vertical distance between TCP and the plane where platform joints 5 and 6 lie;
- l_{01} : length of links 1 and 2;
- l_{02} : length of links 3 and 4;
- l_{03} : length of links 5 and 6;
- s_{01} : semi-distance between parallel guides 1 and 2;
- s_{02} : semi-distance between parallel guides 3 and 4;
- s_{03} : semi-distance between parallel guides 5 and 6;
- s_{h01}, s_{h02} : vertical distance between ground and parallel guides;
- z_{wsd} : height of the center of the desired workspace.

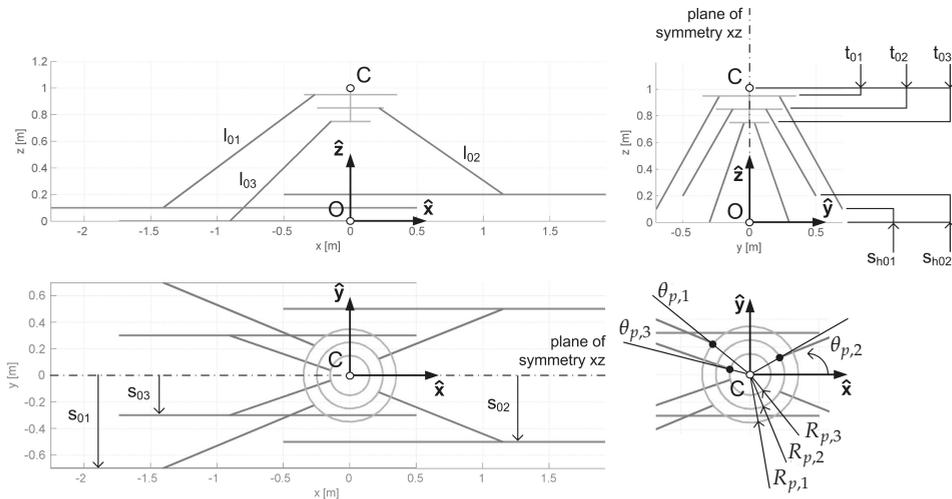


Figure 5. Hexaglide geometric parameters, the Cartesian space is represented in meters.

For this architecture, each kinematic chain is replicated twice, one on both sides of the longitudinal plain of symmetry.

The goal is to optimize robot kinematic performance while ensuring the desired workspace boundaries. In order to define a proper objective function, the desired workspace is discretized in elementary volumes: for each elementary volume, the specific set of parameters under analysis must be allowed to reach its centre point. Moreover, for each centre point analysed, the resulting machine must be able to orientate the TCP with any possible combination of pitch, roll and yaw angles varying inside a predefined range. If that is not the case, the elementary volume is added to the total volume that the current set of parameters is not able to cover. In order to make the computational cost affordable and

restrict the checks to a finite amount of angles combinations, the three angular ranges are discretized. The evaluation of the objective-function is performed firstly by fixing the mobile-platform orientation $\Theta_m = [\alpha_m; \beta_m; \gamma_m]$, and thereafter computing the part of volume that end-effector is unable to reach for these specific orientations, such as:

$$v_{nc}(\Theta_m) = \sum_i \sum_j \sum_k c_{i,j,k}(\Theta_m) \Delta v \quad \begin{cases} i = 1, \dots, N_x, \\ j = 1, \dots, N_y, \\ k = 1, \dots, N_z, \end{cases} \quad (1)$$

where the combination of parameters i, j, k unequivocally identify a check-point of the workspace in which N_x, N_y, N_z represent the number of discretization points, respectively, along the x, y, z axes. The variable $c_{i,j,k}(\Theta_m)$ is equal to 1 if the specific parameters prevent the end-effector from reaching the pose defined by i, j, k and Θ_m , and 0, otherwise. The term Δv is the elementary volume resulting from the discretization of the workspace.

This procedure is repeated for all combinations of pitch, roll and yaw angles and the resulting value for the objective-function V_{nc} is computed as:

$$V_{nc} = \sum_m v_{nc}(\Theta_m). \quad (2)$$

Even if a volume portion is discarded for only one specific orientation, that portion of workspace is excluded for the corresponding geometrical configuration. Using this approach, in the ideal case in which a set of parameters allows the robot to reach any point in the workspace regardless of its orientation, the objective-function would be equal to 0. On the other hand, if the set of parameters does not allow the reaching of a particular point of the workspace, the objective function would assume a value equal to the dimension of the single discrete volume times the number of orientations for which that volume has been discarded.

The kinematic capability of reaching each point with every possible orientation is not in itself enough, but additional constraint definitions are required. The kinematic constraints are defined as follows:

- Distance between the i -th platform joint and the corresponding base joint should not exceed the length of the link for geometric congruence;
- Each actuated joint coordinate has to be comprised within a range defined by the dimension of the machine, since actuators' stroke range have a direct impact on the major bulk direction of the machine, longitudinal for the Hexaglide, and radial for the Hexaslide;
- Each passive joint, both platform and base ones, should respect their respective mobility ranges.

The kinetostatic constraints are defined by the transmission ratio between forces and moments acting on the end-effector and the actuation forces. This transmission ratio for each actuator is computed and the maximum should be lower than a prescribed limit value. As is common for PKMs, this transmission ratio varies considerably within the nominal workspace due to nonlinear kinematics, especially in proximity of the singular configurations. The geometric constraints enforce the respect of the minimum distance between two links and between a link and the mobile platform in order to avoid the problem of self-collision of the component for particular poses.

The solution related to the minimisation of the objective function has been obtained through a single objective genetic algorithm approach [6]. The use of a semi-stochastic search and the evaluation of the performance of different individuals at each iteration makes the process of finding the global minimum of the objective-function to be minimized, easier. The steps characterizing this kind of algorithm are:

- Choice of a sufficiently high number of individuals representing a generation in order to have a significant statistical sample;

- Evaluation of the performance for each individual of the current generation depending on the values assumed by its genes;
- Choice of the group with the better performance that will constitute the elite and will pass unchanged to the next generation;
- Creation of a new generation based on elitary choice, crossover and mutation;
- Computation of the performance of the individuals of the new generation in comparison to the goal desired.

The algorithm would keep modifying parameters trying to cover the desired workspace as much as possible and minimizing the defined objective function. In Tables 1 and 2, the lower and upper bounds imposed at the beginning of the optimization and the optimal values obtained are reported.

Table 1. Limits and results of the Hexaslide optimization.

Symbol	Lower Bound	Upper Bound	Optimal Value	MU
z_{WSd}	400	500	463.6	mm
s	200	300	203.1	mm
l	400	700	686.6	mm
R_p	200	250	238.7	mm
θ_p	0	60	38.5	°

Table 2. Limits and results of the Hexaglide optimization.

Symbol	Lower Bound	Upper Bound	Optimal Value	MU
z_{WSd}	500	700	593.9	mm
s_1	100	980	1050.8	mm
s_2	100	980	277.7	mm
s_3	100	980	330.2	mm
l_1	600	1600	1207.7	mm
l_2	600	1600	873.9	mm
l_3	600	1600	825.7	mm
$R_{p,1}$	200	400	306.2	mm
$R_{p,2}$	200	400	326.7	mm
$R_{p,3}$	200	400	275.9	mm
t_1	-200	0	-183.0	mm
t_2	-200	0	-72.4	mm
t_3	-200	0	22.9	mm
θ_{p1}	10	170	77.3	°
θ_{p2}	10	170	37.9	°
θ_{p3}	10	170	100.6	°

Minimum distance between the links, minimum distance between links and platform, transmission ratio and maximum and minimum strokes of the actuated joint coordinates are mapped throughout the workspace to guide one in the choice of the best architecture. The following conclusions hold respectively for the Hexaslide and Hexaglide.

Hexaslide:

- Topology: both the height and the in plane bulkiness of the manipulator are quite limited. When the robot is in the home position, the links are arranged in such a way that a good compromise is achieved between the capacity to generate velocity in all directions and to bear external forces without too much effort required by the actuators.
- Link-to-link and link-to-platform minimum distances: The link-to-platform minimum distance recorded is above 270 mm throughout the workspace, thus avoiding any risk of collision between the legs.

- Force transmission ratio: the worst case obtained by the computation is closed to the limit value of 5 but restricted to only a few small lower regions of the workspace.
- Actuated joints maximum and minimum stroke: as the joints coordinate distance with respect to the global reference frame is always positive, it is sufficient to check the maximum value in order to evaluate the bulkiness of the robot. This joint coordinate excursion is about 1 m.

Hexaglide

- Topology: The in plane bulkiness of this robot is much higher with respect to the Hexaslide. Moreover, the centre of the desired workspace in the optimized configuration is placed in a higher position compared to the Hexaslide configuration.
- Link-to-link and link-to-platform minimum distances: no risk of self-collision between the elements has been detected.
- Force transmission ratio: the transmission ratio remains limited above 2.5.
- Maximum and minimum stroke: the bulkiness of this solution is higher compared to the Hexaglide one.

It can thus be concluded that Hexaslide architecture is more compact compared to Hexaglide but reaches higher values of force transmission ratio. The Hexaslide architecture has been chosen due to its lower vertical bulkiness, more compact in plane dimensions, better workspace isotropy and lower position of the workspace centre. All of these features make it possible to install the machine under the wind tunnel floor level, reducing the robot influence on the air flow quality and keeping the turbine farther from the wind tunnel ceiling. Moreover, Hexaslide offers two additional advantages: all the elements that make up the links are the same for all six of the kinematic chains and the radial symmetry simplifies the design process. Hereafter, this machine will be referred to as Hexafloat, while the term Hexaslide will refer to the specific architecture of the robot.

2.2. Kinematics Analysis

In order to develop the optimisation problem and design the control algorithm of the robot, it is necessary to solve the inverse and forward kinematics. In this section, for the sake of brevity, only the solution of the inverse and forward kinematic problem of the robot architecture chosen is presented.

2.2.1. Inverse Kinematics (IK)

In order to study *Inverse Kinematics*, two different reference frames have been considered, the global one and the local one with origin in the TCP and built into the robot platform. With reference to Figure 6, two different vector closures for each kinematic chain can be set up in order to compute the joint coordinates vector q . The first vector closure allows one to determine the absolute position vector \mathbf{d}_i of the platform joint B_i with respect to a point that is the intersection of the \mathbf{q}_i direction with its orthogonal plane passing through the global origin. Each \mathbf{s}_i has a fixed length with different orientations in the X_0 - Y_0 plane:

$$\mathbf{d}_i = \mathbf{p} + R_{TCP}\mathbf{b}_i - \mathbf{s}_i, \quad (3)$$

where R_{TCP} is the rotation matrix defining the platform orientation. This matrix is used to transform the expression of \mathbf{b}_i , constant into a rotating local reference frame, in its equivalent with respect to a zero orientation local frame translating with the platform. Position vector \mathbf{p} and the offset \mathbf{s}_i complete the transformation from the zero orientation local frame to the global one. The second vector closure allows one to determine the position of the i -th slider on the guide:

$$\mathbf{l}_i = \mathbf{d}_i - \mathbf{q}_i \hat{\mathbf{u}}_i. \quad (4)$$

The magnitude of \mathbf{l}_i vector corresponds to the length of the robot leg while $\hat{\mathbf{u}}$ represents the guide direction unitary vector. This procedure is identical, except for the orientation of \mathbf{b}_i , $\hat{\mathbf{u}}_i$ and \mathbf{s}_i , for all six links of the robot.

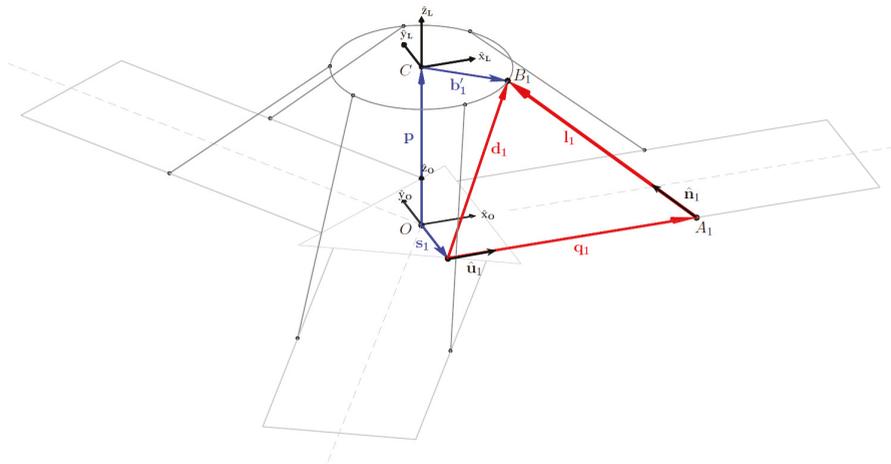


Figure 6. Vector closures used to solve the inverse kinematics (IK) problem: in blue, the 1st closure while in red the 2nd one.

2.2.2. Forward Kinematics (FK)

The Hexafloat robot will not be equipped with sensors able to directly measure the platform position and orientation, therefore a good quality FK could work as a virtual sensor for the pose of the robot. A high frequency estimation of the actual pose of the TCP would be a valuable feedback tool for the other controller in the HIL setup, but a short calculation time and overall stability, in the case of numerical algorithms, is crucial. Extensive research has been conducted to ascertain analytical methods to solve the FK of PKMs, especially for Gough–Stewart configuration, but the pose of the robot has not been expressed in explicit form so far.

Considering numerical methods, the Newton–Raphson (NR) algorithm has its numerical stability highly dependent on the accuracy of the initial approximation of the solution vector so a monotonic descent operator can be added obtaining the so-called Modified-Global-Newton–Raphson (MGNR) algorithm, able to estimate the FK solution of six-DoF parallel robots for any initial approximation in the non-singular workspace without divergence. The algorithm requires the definition of a system of six nonlinear equations and the evaluation of a matrix of partial derivatives:

$$f_i(\underline{X}) = 0, \tag{5}$$

$$P_{ij} = \frac{\partial f_i}{\partial x_j}. \tag{6}$$

The evaluation of partial derivatives matrix can be simplified by implementing Jacobian-Free-Monotonic-Descendent (JFMD) algorithm. A first-order Taylor expansion can be used to approximate the partial derivatives matrix in a numerical way. The JFMD method is implemented via the following steps:

- proper initial approximation of x_0 for the solution is chosen and the corresponding $f_i(\underline{X}) = 0$ is calculated;
- the $(k + 1)$ th solution attempt is calculated according to the following formula:

$$x_{k+1} = x_k - \rho_k [J(x_k)]^{-1} f(x_k), \tag{7}$$

where ρ_k ($0 \leq \rho_k \leq 1$) is the monotonic descent factor. It starts from 1 and during each iteration is calculated as $\rho_k = 2^{-m}$, where m is the number of rechecking times in the corresponding iteration,

necessary to obtain the monotonic trend. The approximated Jacobian matrix is evaluated from the first-order Taylor expansion of its partial derivatives and considering a perturbation parameter η of 1×10^{-16} . The i-th row and j-th column of the approximated Jacobian matrix is evaluated as:

$$J_{ij} = \frac{f_i(x + \eta_j e_j) - f_i(x)}{\eta_j}, \tag{8}$$

- convergence criterion is defined imposing the error to satisfy the following inequality:

$$\|x_k - \rho_k [J(x_k)]^{-1} f(x_k)\|_2 \leq \|f(x_k)\|_2, \tag{9}$$

- the algorithm stops if convergence is achieved or the maximum number of iterations is reached:

$$\|f(x_{k+1})\|_2 \leq \delta = 1 * 10^{-10} \quad \text{or} \quad k \geq k_{max} = 20, \tag{10}$$

where δ is the required computation tolerance and k_{max} is the given maximum number of iterations.

The first initial approximation must be taken close to the Home Position (463.56 mm from the ground reference frame), where the robot is supposed to be when switched on. For the following times, the initial approximation is chosen as the estimated pose at the previous cycle. The logic of the JFMD algorithm is clarified in the flow chart in Figure 7. Thanks to C language implementation, this routine is able to converge to a solution in relative little iteration, with a mean calculation time of around 50 μ s in the actual hardware chosen to control the robot. This performance allows one to use FK as a virtual real-time sensor with sufficient accuracy.

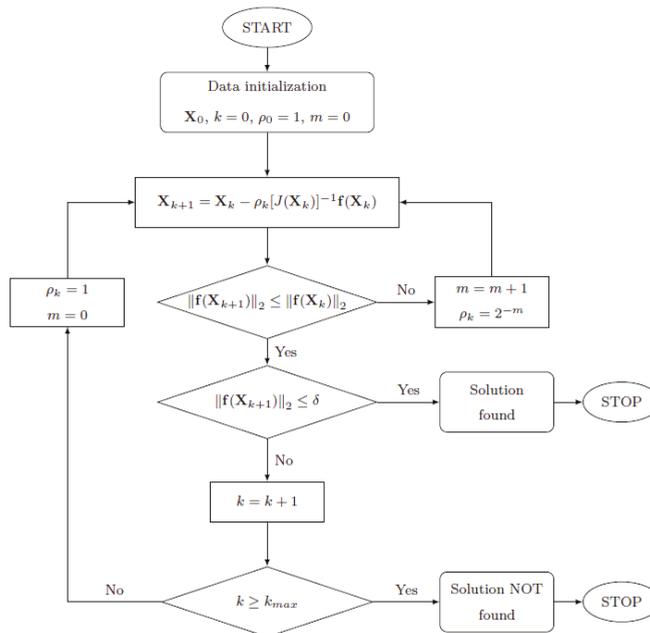


Figure 7. Flow chart of the Jacobian Free Monotonic Descendent (JFMD) algorithm.

2.3. Velocity Analysis and Kinetostatic

In order to determine the relation between the velocity of the TCP and the ones of the joint coordinates, it is necessary to calculate the Jacobian matrix. If all six links are considered together, a compact matrix form coupling the joint velocity vector \dot{q} and the workspace velocity vector w can be defined:

$$\begin{bmatrix} \hat{n}_1^T \hat{u}_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \hat{n}_6^T \hat{u}_6 \end{bmatrix} \dot{q} - \begin{bmatrix} \hat{n}_1^T & \dots & (b_1 \times \hat{n}_1)^T \\ \vdots & \ddots & \vdots \\ \hat{n}_6^T & \dots & (b_6 \times \hat{n}_6)^T \end{bmatrix} w = 0,$$

$$[J_q] \dot{q} - [J_{gs}]^{-1} w = 0,$$

and with some algebraic steps:

$$\dot{q} = [J]^{-1} w. \tag{11}$$

The solution of the kinetostatic analysis provides the actuation forces τ_a required to bear the external forces f_{ec} applied to the TCP. Due to the virtual work principle and knowing that the virtual variation of the workspace coordinates is related to the virtual variation of the joint coordinates through the Jacobian matrix $[J]$, the actuation forces can be computed as:

$$\tau_a = -[J]^T f_{ec}. \tag{12}$$

The representation of the way in which forces applied to the robot platform are transmitted to the actuators is obtained considering the unitary hypersphere of forces in the workspace [7]. This unitary hypersphere is transformed into a hyper-ellipsoid in the space of actuation forces. This is a common description used in the robotic field in order to characterise the behaviour of the robot in every point of the working space and it states that, if one of the forces applied to the TCP reaches the maximum value of 1, the other components must be nil. An alternative representation is based on the use of a hyper-cube of unitary semi-side that is transformed into a hyper-polyhedron through the Jacobian matrix, as shown in Figure 8. Moreover, it should be noted that the inverse Jacobian matrix can be split into a translational and a rotational part as presented in [8]. Using this approach, it is possible to note that the rotational components of the inverse Jacobian matrix are dimensional and in fact they correspond to a length. In order to let all the elements of the inverse Jacobian matrix be dimensionless, it is useful to divide the rotational components by a scale factor defined as characteristic length L_c . In this way, a normalized inverse Jacobian matrix is obtained and is used to obtain the maximum actuation force.

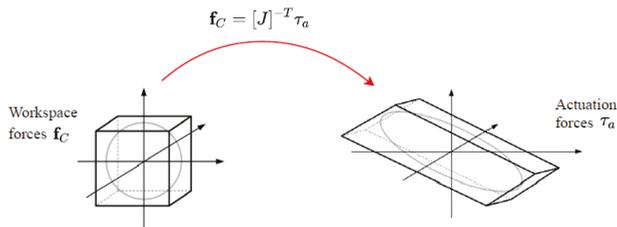


Figure 8. Transformation of the workspace forces unitary hypercube into the actuation forces hyperpolyhedron.

3. Actuation Chain Sizing

A proper sizing of mechanical components and actuating systems requires the knowledge of the most power demanding motion that the machine is required to execute, considering the influence of robot dynamics and carried loads. For the application taken into account, the movement of the end

effector is unknown in advance because it depends on the wind effect over the wind turbine and the mathematical model used to define the behaviour of the floating structure. To obtain a scenario of the possible movement of the end effector, it is necessary to consider an approximation of the wave motion of the sea expressed by a combination of six cosinusoidal functions [9,10]. Analysing any combination of a parameter set, which is used to define the sea behaviour, one can obtain a possible end effector motion. The time history of the sea level ε is expressed as:

$$\varepsilon(t) = \rho(t) \cos(\omega t + \varphi(t)), \quad (13)$$

where ρ is the amplitude, ω the pulsation, and φ the phase shift. Both amplitude and phase shift change very slowly in time so it is reasonable to assume they are constant for the whole duration of the simulation and the time history of each DoF is described as:

$$j(t) = A_{0,j} + A_j \cos(2\pi f_j t + \varphi_j) \quad \text{with} \quad j = x, y, z, \alpha, \beta, \gamma, \quad (14)$$

where $A_{0,j}$ is an offset parameter taking into account the initial pose of the robot.

The worst operating conditions may be represented in the case where all frequencies f_j and amplitudes A_j assume their maximum value. However, the intrinsically nonlinear kinematic of the robot may invalidate this assumption. In effect, it is not guaranteed that this is the most demanding case in terms of internal loads, motor torques, velocities and accelerations. Furthermore, a simple co-sinusoidal motion of the end-effector is translated into a periodic motion of the joint coordinates where higher order harmonics with respect to the f_j appear. It is difficult to predict if the energy associated with the higher order harmonics of a specific f_j is bigger than the energy of the harmonics corresponding to a lower f_j .

The four characteristic parameters are thus defined in the following ranges:

- Initial pose $A_{0,j}$: may vary in the range $\pm L_{WSd,j}$;
- Frequency f_j : in the range between f_{min} and f_{max} ;
- Phase shift φ_j : in the range of $[0, 2\pi]$;
- Amplitude A_j : its maximum range is $[-L_{WSd,j}, L_{WSd,j}]$ and the relation $|A_{0,j} \pm A_j| \leq L_{WSd,j}$ has to be verified to guarantee that the end-effector remains within the boundaries of the desired workspace. The effective range of A_j is obtained by combining the maximum range with the expressed relation, eventually modifying it in the event that $A_{0,i}$ is different from zero.

In order to test any pose in the working volume, a procedure has been designed [11]. Firstly, the workspace has been divided into a finite number of portions. For each of these, an admissible range of motion parameters for each degree of freedom has been chosen. In particular, these ranges have to be discretized in order to obtain a finite number of combinations. Therefore, a set of time histories for the six-DoF end-effector is generated and thereafter the inverse kinematic and dynamic problem solved in order to calculate the motor torques required for each set.

3.1. Multibody Model

The procedure described above needs a mathematical model to be implemented. In particular, a multibody model of the robot has been developed using the commercial software MSC ADAMS 2016 in order to compute the dynamic and kinematic quantities that allow one to properly size mechanical components as well as the actuating system [12]. At the preliminary stage of the project, the inertial properties of the components, that make up the robot, are unknown and must be estimated. The results obtained by the use of the multibody model and the developed procedure are used to refine these values and update the model to subsequently enhance the results accuracy and converge towards the final result.

The developed multibody model is based on a parametric approach in order to be easily integrated into the motor sizing procedure. The first step for the model formulation is the definition of a set

of reference frames located in positions that allow one to easily define the inertial properties, joints, applied forces and measurements points. The reference frames created are: a TCP reference frame; six reference frames called B_i in correspondence to the centre of the end-effector joints; six reference frames called A_i in correspondence to the centre of the base universal joints; six reference frames called P_i located at the origin of the guide axes, as shown in Figure 9.

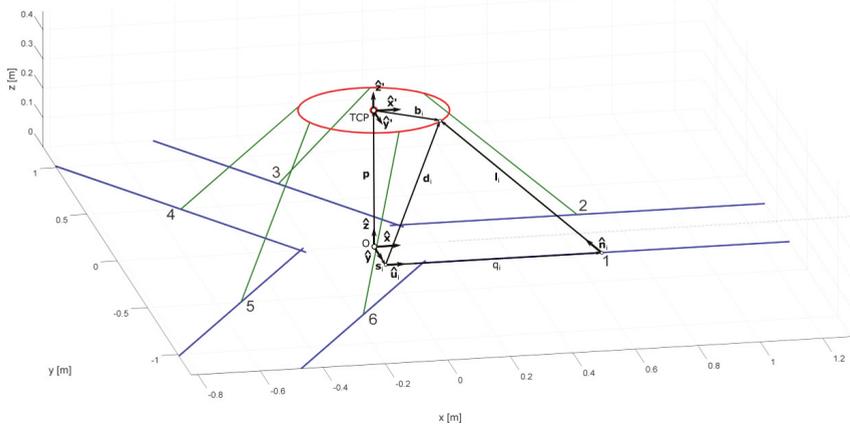


Figure 9. Reference frames configuration, the Cartesian space is represented in meters.

The three main components for each kinematic chain are: platform, links and sliders. The element called platform takes into account the mobile platform, the six spherical joints attached to it, the six axes load cell used to measure the forces and moments exerted by the scaled model of the turbine on the robot. Due to the symmetry, the centre of mass is expected to be located on the z-axis of the TCP reference frame and the principal axes of inertia are expected to be parallel to the axes of the TCP. Links can be represented as cylinders whose centre of mass is located in the middle of the link and whose principal axes of inertia are aligned with those of the A_i reference frame. The elements called sliders take into account the universal joints, joint supports and sliders of the transmission unit. Since these bodies will be subjected to a purely translational motion during simulations, it is sufficient to characterise them with their total mass. This schematization with the equivalent mass and inertial properties of the assembly is possible because the model described in this section is supposed to be rigid. The mechanical and geometrical limits of each joint are represented by locking one or more relative degrees of freedom and giving a limited range of displacement/rotation among the possible ones.

3.2. Monte Carlo Method

Due to the impossibility in the definition of the most demanding task for this application, a statistical approach has been developed, based on the Monte-Carlo Method. This novel approach is implemented as follows:

- Choice of a sufficiently high number M of simulations, each of which has a specific time history for every DoF.
- Definition of the Probability Density Functions (PDF) of the input parameters that define every motion task: it is assumed that, at the initial instant, the TCP may be located with the same probability in any point of the workspace. The PDFs chosen for the parameters describing the motion are respectively: for the initial pose a uniform distribution, for the frequency a uniform distribution, for phase shift a uniform distribution and for the amplitude a Rayleigh distribution.

- Repeated sampling of the chosen PDFs: each parameter is allocated a random value for each simulation. The result is a set of M vectors fully defining a six-DoF motion task to be assigned to the TCP.
- Solution of the IK problem to find the joint coordinates time histories $q(t)$ to be used as inputs for the simulations involving the multibody model.
- Solution of the inverse dynamics problem for each of the M simulations.
- Post-processing: evaluation of a distribution among the M simulations of the parameters of interest.

The results collected are used to size the mechanical components and the actuators. In particular, it is possible to obtain the slider velocity and acceleration, forces exerted by the motors on the sliders, axial forces along the guide and the so-called “Load factor” for the actuator sizing procedure. A summary set of values is required in order to easily compare these quantities with the corresponding limits specified by the manufacturers. These sets are computed through the extraction of the maximum values for each simulation stored in M array; among these, the highest value is extracted. Therefore, the interval from 0 to the highest extracted value is divided into subranges, and, for each of these, the number of occurrences of the M values is assessed. A set of discrete PDFs is obtained and this represents the probability that the maximum value obtained during a simulation is comprised within a certain interval.

3.3. Actuating System Sizing Procedure

The selection of an actuating system requires one to choose both the electric motor and the gearbox unit. In scientific literature, several procedures to size the motor reducer group are available. In this work, the approach proposed in [11,13] is applied. Independently from the procedure used, the motor reducer sizing is based on the checking of the following three relationships:

- Limit on maximum torque:

$$\max |C_m(t)| < C_{m,max} \quad \text{and} \quad d \max |C_m(t)| < C_{t,max},$$

- Limits on nominal torque:

$$C_{m,rms}^* < C_{m,nom},$$

- Limit on maximum speed:

$$\max |\omega_r(t)| < \tau \omega_{m,max} \quad \text{and} \quad \max |\omega_r(t)| < \omega_{t,max}.$$

C_m is the motor torque, $C_{m,max}$ the maximum torque the motor is able to generate, and $C_{t,max}$ is the maximum torque that the transmission unit is able to bear. In addition, ω_r represents the resistant speed computed on the load side, $\omega_{m,max}$ is the maximum speed the motor can reach without damaging its mechanical components, $\omega_{t,max}$ is the maximum speed the slider can achieve without damaging mechanical components. Following the [13] approach, from the power balance and the thermal check inequality, it is possible to define the transmission ratio as follows:

$$\alpha > \beta + [C_{r,rms}^* (\frac{\tau_{rid}}{\sqrt{J_m}}) - \dot{\omega}_{r,rms} (\frac{\sqrt{J_m}}{\tau_{rid}})]^2, \tag{15}$$

where the acceleration factor α and the load factor β are defined as:

$$\alpha = \frac{C_{m,nom}^2}{J_m}; \quad \beta = 2 [\dot{\omega}_{r,rms} C_{r,rms}^* + (\dot{\omega}_r C_r^*)_{mean}]. \tag{16}$$

The RMS values of torque and acceleration are defined as:

$$C_{r,rms}^* = \sqrt{\frac{1}{t_c} \int_0^{t_c} [C_r^*(t)]^2 dt} \quad \text{and} \quad \dot{\omega}_{r,rms} = \sqrt{\frac{1}{t_c} \int_0^{t_c} [\dot{\omega}_r(t)]^2 dt}.$$

The acceleration factor α is a motor characteristic that can be evaluated directly from the datasheets, whereas the load factor β depends on the task performed by the actuations system, and, for the case under study, can be calculated using the multibody model. Following the sizing procedure, firstly, the optimal transmission ratio value τ_{opt} is evaluated. Then, this value is compared to the transmission ratio purchasable in the market and the closest is chosen taking into consideration that it is within the available range. In particular, greater attention must be paid to evaluate the maximum velocity required during the task and the relative limit transmission value using the formula:

$$\tau_{lim} = \frac{\max|\omega_r(t)|}{\omega_{m,max}} \max(\tau_{lim}, \tau_{min}) < \tau < \tau_{max}. \tag{17}$$

All the results obtained from the multibody simulations refer to the kinematic and dynamic quantities of the manipulator sliders. In order to transform force into torque and linear velocity or acceleration into angular ones, the following equations have to be used:

$$\dot{\omega}_{r,i} = \frac{\ddot{q}_i(t)}{\tau_{TU}} \quad C_{robot,i}(t) = \tau_{TU} F_{robot,X_i}(t). \tag{18}$$

It is worth noticing that to increase the transmission stiffness and achieve the maximum velocity required, every linear axis of the robot is made up of only a single reduction stage. In particular, the motor is directly connected to the screw by means of a rigid joint and the transmission ratio is the screw lead. Due to the high dynamic performance, stiffness and precision required, recirculating ball screws have been chosen. The double slider configuration for each guide has been selected in order to guarantee a uniform load distribution. The manufacturer’s instructions to correctly size these devices have been followed, in particular to the total length of the screw, maximum allowed velocities, forces that the guide can bear and the like. All of this is summarized in Table 3. As concerns the electric motor OMRON (Tokyo, Japan), as well as the heat dissipation check, the maximum torque achievable has been evaluated.

Table 3. Resume of checks done on guides.

Parameter	Value		Limit Value	UM
Length of the guide	1100	>	650	mm
Max velocity	1.67	<	2.66	m/s
Max acceleration	28.5	<	50	m/s ²
F_x	1385	<	12,250	N
F_y	1385	<	69,600	N
F_z	1385	<	69,600	N
M_x	174	<	3028	Nm
M_y	174	<	2290	Nm
M_z	174	<	2290	Nm

Figures 10 and 11 show the maximum torque distribution and maximum rotation speed obtained during simulations. Once fixed a 99% coverage threshold, the corresponding values are extracted and they are used as guidelines in the components’ choice. The most restricting limit is imposed by transmission units: the Rollon ballscrew drive (Vimercate (MB), Italy) chosen satisfies the maximum torque test in 94.7% of the cases with Rayleigh approach and in 89.4% with the other one. Even if the coverage is not 99%, the choice has been made to find a suitable trade-off between performances and size. As for maximum torque analysis, the most restricting limit for rotation speed is imposed by the linear transmission unit, but, since the selected transmission satisfies more than 99% of the cases, this test is passed.

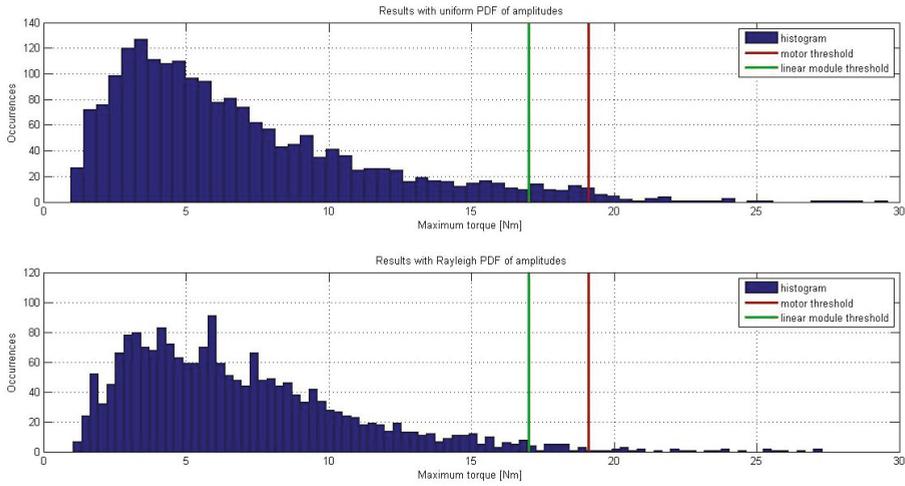


Figure 10. Maximum torque distribution along all the simulations.

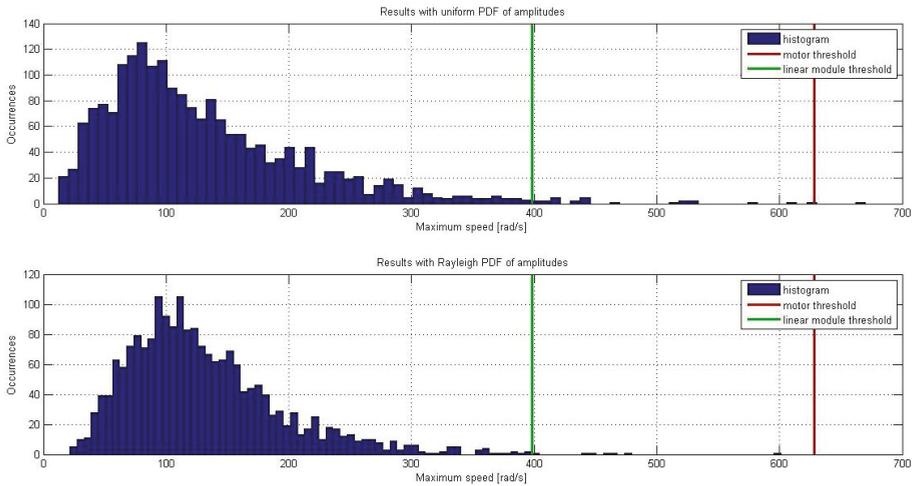


Figure 11. Maximum angular velocity distribution along all the simulations.

4. Mechanical Design and Component Sizing

A design process is always iterative: an initial model, which satisfies the preliminary parameters required, is firstly created, thereafter each part of this model is refined and improved in order to enhance its dynamic characteristics and reduce mass and costs of the whole system. In this process, the multibody model plays an important role: it is used many times in an iterative way to analyse the dynamic behaviour of the system after every single modification in order to check the mechanical components and actuation system. For the development of a simulator, it is necessary to realize two kinds of structural analyses: the first one is a static analysis for which the focus is to determine the stress and strain of any mechanical components. The second analysis is a dynamic one performed to obtain the modal behaviour of the system.

4.1. Robot Description and Overview

In order to clarify the results obtained with static and dynamic analysis, a complete overview of the “as built” project is necessary, Figure 12a. The entire system can be divided into two main parts: the Hexafloat manipulator and the auxiliary systems. The first part is made up of “Grounded elements” (all the fixed supporting structure and the power and actuation units), “Joints”, “Links” and “Platform”, whereas the auxiliary systems include: “Lifting system” (a tool used to manage the rest task of the system), “Energy chain” (device that houses cables and the like) and “External Sensors” (sensors for monitoring the system during the operating tasks).

Hereafter is a more detailed description of every component and technical solution adopted.

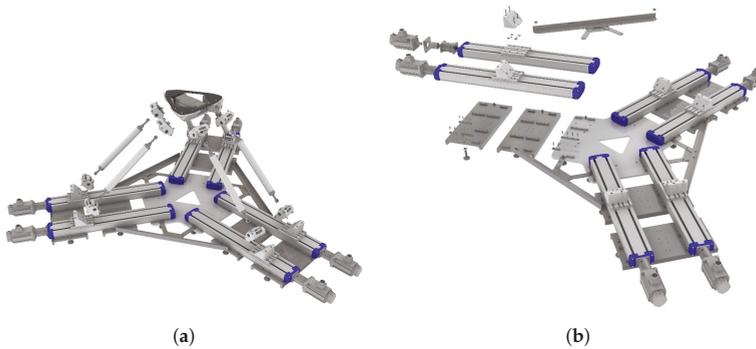


Figure 12. Components. (a) Hexafloat exploded; (b) parts on the ground partially exploded view.

4.1.1. Grounded Components

This is the framework and gives stability to all the moving parts, Figure 12b. The linear actuation systems are fixed onto it. Therefore, this assembly should be stiff enough to bear static and dynamic loads. The *Central plate*, made of aluminium, is used to provide the correct orientation and position of the linear guides through calibrated machining. The linear actuators are grouped into two by two and each couple is oriented with an angle of 120° with respect to the other. This leads to a radial symmetry of the machine. Each couple of linear actuators is fixed to the *Central plate*. To provide further stiffness to the system, avoid undesired tilting and relative displacement between the two couples of guides, three “K” links constrain the *Central plate* with the “C” shaped supports over which the actuation system is mounted. *Aluminium joint holders* are used to couple the *Joints* with the linear actuators. Through the optimization process and the solving of the kinematic closure equations, when the robot is in the *Home Position*, the direction of each *Link* is defined and the *Joints holder* shape and inclination designed. The whole framework will be placed in the wind tunnel using 22 levelling elements to distribute the weight uniformly and maintain the guides on the same plane, without misalignments.

4.1.2. Joints

The PUS kinematic chain on the basis of which the manipulator is realized is made up of an actuated prismatic joint followed by a double revolute joint (universal joint) and a spherical one mounted on the *Platform*. Placing a universal joint and a revolute one so that the rotation axis of the latter passes through the intersection of the universal joint rotation axes, it is possible to obtain the same numbers of DoFs that the spherical joint has. Therefore, the same universal joint used in the lower part of the kinematic chain is mounted on the *Platform*, and the rotational degree of freedom is directly realized within the *Link*. In order to obtain the stiffness, precision and mobility required, a custom joint solution has been realized and they are shown in Figure 13a,b. The main feature of such

Joint is the possibility to have a cone shaped motion range with a semi-angular aperture of 45° around the normal direction, designed and assembled to have zero backlash.

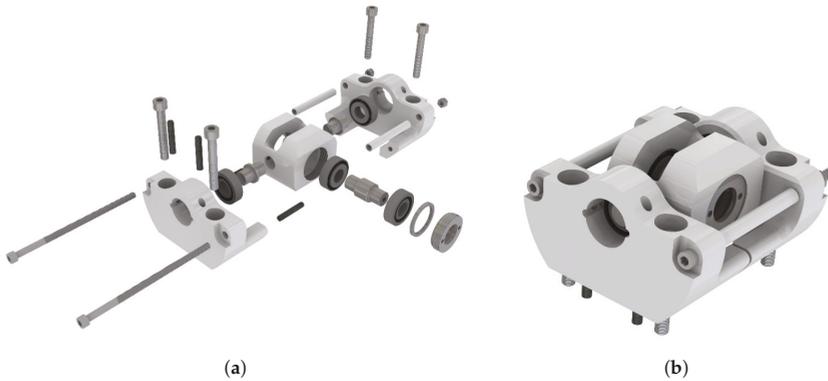


Figure 13. Custom joint. (a) joint exploded view; (b) joint assembly.

Each joint is made up of a couple of *Half shells* connected to an *Inner block* through two roller bearings and two *Support shafts*. These components are tightened by two screws and aligned by two calibrated dowel pins. In this way, the *Inner block* has a relative movement with respect to the shell, the further DoF is provided by another two roller bearings that support the *Inner shaft* pinned to the link. All these inner components are packed through a *Distance ring* and a *Closing ring*. *Half shells* and *Inner block* are made of aluminium, while other components are made of steel. The *Joint* is fixed to the *Joints holder* through four screws and aligned with two calibrated dowel pins.

4.1.3. Links

The *Links*, Figure 14, connect the *mobile platform* to the fixed one. These bear the pay load made up of the mobile platform, RUAG (*RUAG Aviation - Aerodynamic, Schiltwald, Switzerland*) load cell, scaled wind turbine and sensors. The RUAG six-DOF weight scale, model W192-6I, is dimensioned for the following forces and moments: $F_x = \pm 1500$ N, $F_y = \pm 1000$ N, $F_z = \pm 5000$ N, $M_x = \pm 500$ Nm, $M_y = \pm 1000$ Nm and $M_z = \pm 600$ Nm. Due to the complexity of this device, it is necessary to use a calibration matrix capable of taking into account not only the weight scale deformation but also the working temperature. This calibration matrix, provided by the manufacturer, follows quality procedures in use at the wind tunnel and it is available at the internet address of the Life50+ project. The *Link* is composed of:

- Lower rod, made of steel, at one side, directly connected to the *Inner shaft* of the *Joint* assembled on the *grounded components*, the other side is used to pack up the bearing group between a mechanical stop and nuts;
- Bearing case, made of steel, which houses the bearings that provide the rotational DoF along the *Link* axis, and is connected to the *Leg pipe* by six screws;
- Leg pipe, a hollow cylinder made of aluminium;
- Distance washer, made of steel, it allows the regulation of the total *Link* length through a threaded connection with the *Upper rod*. It is also connected to the *Leg pipe* by six screws;
- Upper rod, made of steel, the final component connected to the *Inner shaft* of the *Joint* assembled on the *Platform*;



Figure 14. Link exploded view.

4.1.4. Platform

The *Platform*, Figure 15a, is realised in order to define the right position of the joints and the six-DOF weight scale. It is made up of the following elements:

- Bottom plate, made of aluminium, necessary to sustain and distribute the load and it is the frame of the *Platform*;
- Three Angular joints holder, made of aluminium, needed to guarantee the correct orientation angle of the *Joints*;
- Top plate, made of carbon fibres, to give added rigidity to the structure;
- RUAG 6-axis load cell, to measure loads exchanged between the machine and the wind turbine.

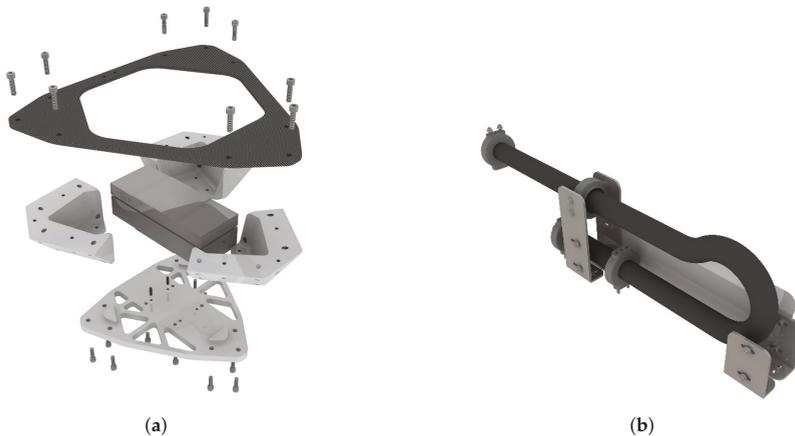


Figure 15. Platform and energy chain. (a) platform exploded view; (b) energy chain assembly.

4.1.5. Auxiliary System

Two auxiliary systems have been designed: an *Energy chain* (Figure 15b) and a *Lifting system* (Figure 16b). The Hexafloat machine is designed to stay below the floor of the wind tunnel when not in operation. In this configuration, the machine has to overcome a singularity to reach Home Position and a *Lifting system* helps the robot to pass this critical point in the rise and return phases. This system can be schematically modelled as in Figure 16a as an isosceles three-hinged arc, in which hinge A is placed on ground, B on the *Moving carriage* and C at the central part of the *Three points beam*, thus allowing a vertical movement of the *Three points beam* top end (point D). Furthermore, a four-link mechanism is coupled to this system in order to keep it parallel to the ground along the whole stroke of the *Coupling platform*, mounted on top of the lifting mechanism. The *Moving carriage* is driven by the motor through a trapezoidal screw, which converts motion from angular to linear.

Measuring and actuation devices will be mounted on the *Platform* and on-board the scaled wind turbine, and these instruments need to be fed by electric energy and have to be connected to a controller in order to guarantee a real-time data exchange. Therefore, a cable housing system is needed in order to protect cables, maintain a good flexibility while following the *Platform* movements, without interfering with these. The system is made up of the following elements: four “L” plates supports, made of steel;

two lateral connecting plates, made of aluminium; two mounting brackets with strain relief mounted, respectively, on the *Platform* and on *Parts on ground*; an intermediate mounting bracket; a mounting sliding bracket and a flexible chain.



Figure 16. Lifting system. (a) lifting system scheme; (b) lifting system assembly.

4.2. Static Analysis

Through the results obtained from the multibody analysis, one can determine the most critical load acting on the system. In particular, the interest is on the force that must be borne by the links because, through this, the remaining parts of the system are loaded. For safety reasons, an overestimated value of 2500 N is assumed both in traction and compression for every link. The static analysis has been performed using a commercial FEM solution and several models have been studied in order to take into consideration both single elements and their subassembly. Taking into account the final configuration, one can state:

- **Grounded components:** The symmetry of the assembly allows one to take into account only one third of the frame. Moreover, only the aluminium profile of the linear actuator is taken into account because it bears the vertical loads. A “worst” vertical load (i.e., 2500 N) is applied on each guide and positioned with different combinations of the total actual stroke of the sliders (i.e., 0%, 25%, 50%, 75%, 100%). Local stresses and strains never exceed the limit values, Figure 17a.
- **Joint:** bearings have been substituted with rigid components to simplify the assessment. Different configurations of angle are tested both as regards compression and traction with an applied load of 2500 N, Figure 18a,b. Only the rod shows visible deformations and stresses. In order to better investigate the behaviour of internal components, other analyses have been carried out and no problems have been reported by the results:
 - **Inner Block:** a load of 2500 N is split into two equal loads, each one acting on a set of bearings. Maximum stress registered is 85 MPa, which is far below the admissible stress of 250 MPa, Figure 19a.
 - **Support shaft:** a load of 1250 N is applied for simulating the presence of two Support blocks per Joint. The test is effected by loading one end of the shaft and maintaining the other one fixed, Figure 19b.
 - **Inner shaft:** a load of 2500 N is applied in the midspan and both ends are pinned in order to simulate the presence of the two bearings, Figure 20a.
- **Link:** an axial load of 2500 N is applied and only traction and compression have been tested. Bearings are substituted with rigid parts, Figure 20b. Considerable stress has been noted on the upper and lower rods, but their values is well below the critical one, Figure 21a,b.
- **Platform:** two horizontal forces are applied at different heights on the TCP. The first of 200 N located at 1.0 m and the second of 100 N at 1.5 m corresponding, respectively, to equivalent

inertial load and aerodynamic thrust due to the pay load. These loads are borne very well by the structure both in terms of stress and displacement, Figure 17b.

- **Lifting system:** this device has to generate a total lift of 150 mm in a time span of about 15 s with a cycloidal motion curve and must also bear the weight of all the moving parts of the robot. This system has been developed in order to help the robot to overcome the singularity configuration during the start and stop procedures. The analysis reveals maximum values occurrence at the initial phase of the rise and FEM analysis ensures that stress remains below the critical values.

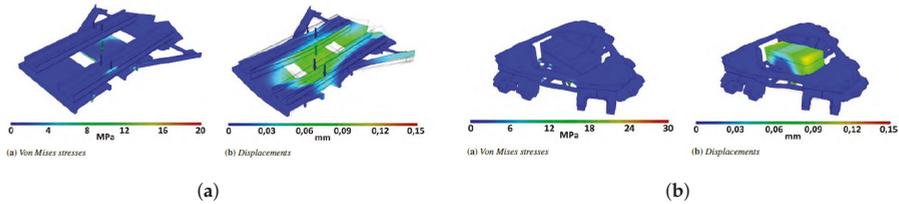


Figure 17. Movable and fixed platform analysis. (a) parts on ground static analysis; (b) platform static analysis.

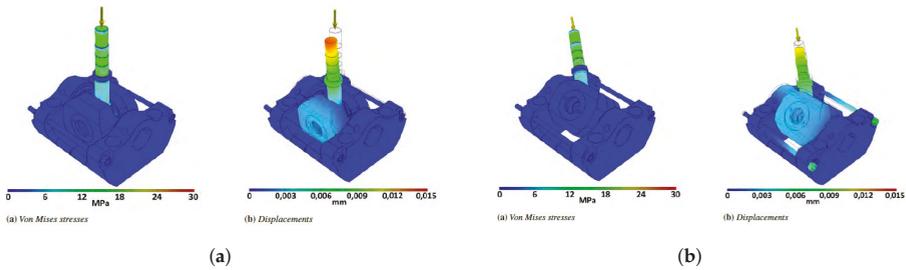


Figure 18. Joint analysis. (a) joint static analysis; (b) joint static analysis, inclined configuration.

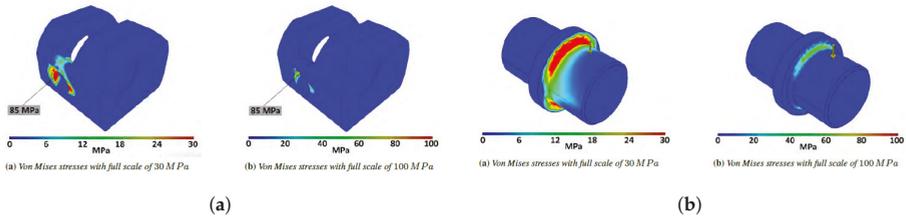


Figure 19. Inner block and shaft analysis. (a) inner block static analysis; (b) support shaft static analysis.

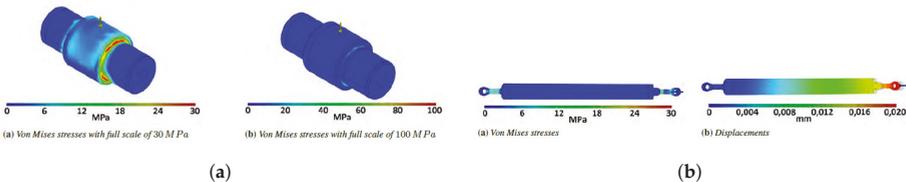


Figure 20. Inner shaft and link analysis. (a) inner shaft static analysis; (b) link static analysis.

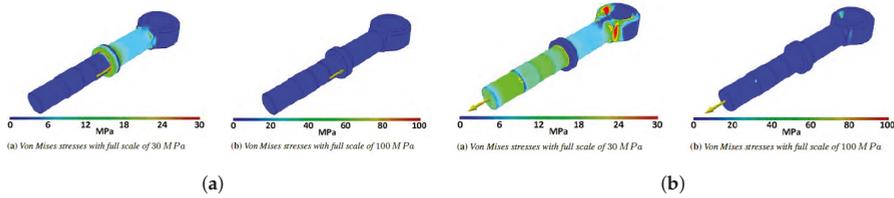


Figure 21. Lower rod analysis. (a) lower rod compression analysis; (b) lower rod traction analysis.

4.3. Purchased Components’ Sizing

Final choices regarding motors, linear actuators and the energy chain are listed.

4.3.1. Motors

The selected model is the OMRON R88M-K2K030F-BS2, whose main characteristics are collected in Table 4(a). The *Lifting system* motor, required to bear the torque increment given by the low efficiency of the trapezoidal screw adopted, is the OMRON R88M-K20030T-BS2, the main characteristics of which are reported in Table 4(b). Both motor models have an auxiliary brake for safety reasons and an encoder to allow one to have a control of position and velocity. For the 2 KW motor of the robot, the encoder is a quadrature incremental encoder with a maximum resolution of 4,194,304 cnt/rev.

Table 4. Motors’ characteristics.

(a) OMRON R88M-K2K030F-BS2 Main Characteristics		
Characteristic	Value	MU
Tension	400	V
Nominal Power	2000	W
Nominal Torque	6.37	Nm
Maximum Torque	19.1	Nm
Nominal velocity	3000	rpm
Maximum velocity	5000	rpm
(b) OMRON R88M-K20030T-BS2 Main Characteristics		
Characteristic	Value	MU
Tension	230	V
Nominal Power	200	W
Nominal Torque	0.64	Nm
Maximum Torque	1.91	Nm
Nominal velocity	3000	rpm
Maximum velocity	6000	rpm

The encoder signal is processed in order to obtain a lower resolution of 131,072 cnt/rev. The resulting linear resolution of 3,276,800 cnt/m is obtained considering a lead of 0.04 m/rev for the ball-screw linear axis. For the lifting system motor, the maximum encoder resolution is 131,072 cnt/rev. The modified used resolution is of 13,107.2 cnt/rev, thus obtaining the same linear resolution of the robots axis due to a smaller lead of 0.004 m/rev.

4.3.2. Linear Actuator

The linear actuator chosen is the model TH145 SP4, produced by ROLLON and whose characteristics are reported in Table 5(a). The following customizations are required: screw lead of 40 mm per revolute, to allow the respecting of required performances; two calibrated centring holes on its lower side to correctly assemble the central plate; two calibrated centring holes on the external carriage to allow the correct positioning of the Joint holder.

Table 5. Other components' characteristics.

(a) ROLLON TH145 SP4 Main Characteristics		
Characteristic	Value	MU
Total length	1100	mm
Width	145	mm
Thickness	85	mm
Screw diameter	20	mm
Number of charts	2	-
Total system mass	30	kg
(b) Igus Triflex TRE.40 Main Characteristics		
Characteristic	Value	MU
Total length	1600	mm
Number of chain links	116	-
External diameter	43	mm
Curvature radius	58	mm
Maximum single cable diameter	13	mm
Maximum relative rotation	± 10	$^{\circ}$

4.3.3. Other Components

The selected energy chain is Igus (Colonia, Germany) Riflex TRE.40 whose main characteristics are reported in Table 5(b). Tapered roller bearings chosen both for the links and joints are the INA FAG 30202-A. These support both axial and radial loads, thus avoiding the usage of more than one bearing. KTR-TOOLFLEX20M motor coupling is chosen. Due to the reduced space available for the *Lifting system*, a more compact model of bearing INA FAG 3000-B-2RS-TVH has been chosen. It is a double row angular contact ball bearing, which can bear both axial and radial loads. KTR Rotex 19/92Sh-A/2.1- ϕ 11/2.0- ϕ 8 motor coupling has been chosen for the *Lifting system* and it provides the required performance and encumbrance constraints.

4.4. Modal Analysis

In this section, the approach followed to assess the correctness of the overall design, in terms of dynamic behaviour, is reported. More specifically, the goal is to verify that the first normal modes of the coupled structure coincide with those of the turbine only, in order to minimize the dynamic effects of the robot. Nevertheless, the modal behaviour of a robot strongly depends on the specific pose of the end effector, since the mass and stiffness distribution change according to the pose.

The normal modes and their associated frequencies are computed in order to verify that the frequencies in the first normal mode were high enough. In particular, they must be higher than the well defined frequency range for all the robot poses in to the working volume. The frequency range depends on the dynamic behaviour of the scaled turbine.

To analyse the dynamic response of the manipulator, the entire workspace was discretized, and the trend of the frequency corresponding to the first normal mode over the entire workspace was mapped on specific planes that intersect the robot workspace. The procedure is listed below:

- Identification of planes that intersect the workspace.
- Identification of a grid of equally spaced points on each plane.
- Discretization of pitch, roll and yaw angles describing the end-effector orientation:
 - Three roll angles: -5° , 0° , $+5^{\circ}$,
 - Five pitch angles: -8° , -4° , 0° , $+4^{\circ}$, $+8^{\circ}$,
 - Three yaw angles: -3° , 0° , $+3^{\circ}$.

- Modification of the pose of the robot in order to have the end-effector placed in correspondence to each point of the grid and exploring all the possible orientations.
- In correspondence to each pose, linearization of the flexible virtual model and computation of the frequency associated to the first normal mode.
- For each point of the grid, recording the lowest value of frequency among the ones obtained by changing the orientation angles.

The final result is a set of maps, one for each intersecting plane, which show the trend of the lowest frequencies regardless of the orientation of the robot, as shown in Figure 22. To perform these sets of simulations, a simplified design of the robot was created. Lower simplified joints are tied to the ground applying spherical pins and the simulation provides the first three eigenmode frequencies. Links can be regarded as the main cause for a possible deterioration of the behaviour of the coupled system. The main deformations are concentrated on links and an advanced multibody model has been developed in the “Adams” environment, Figure 23, whose links are made of two extremity elements and an intermediate one characterised by aluminium properties. The results given by the two numerical environments are reported in Table 6.

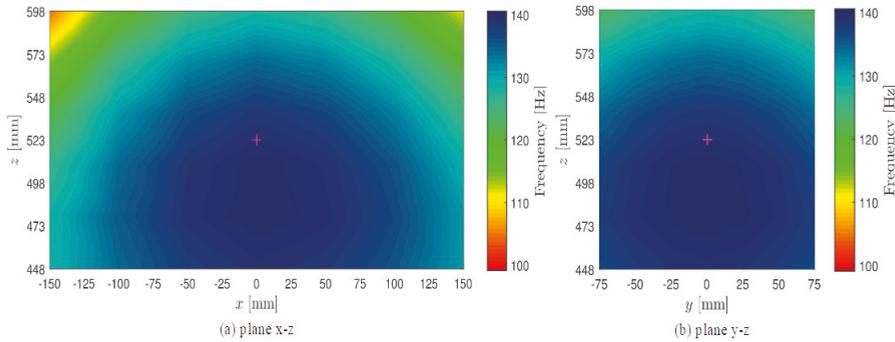


Figure 22. Modal frequencies maps.

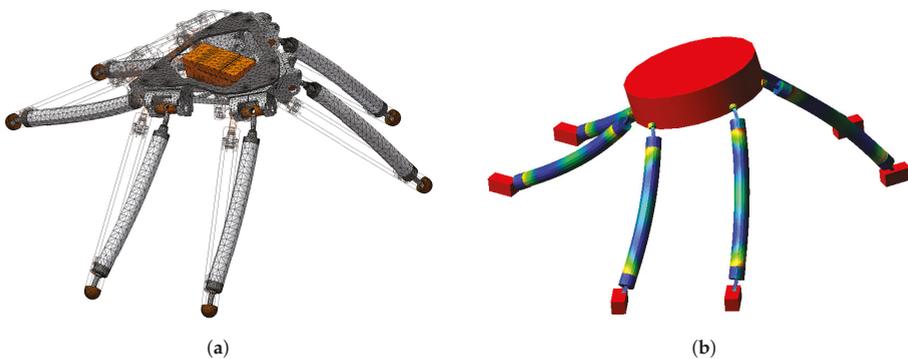


Figure 23. Numerical modal analysis. (a) Inventor 1st mode of vibration; (b) Adams 1st mode of vibration.

Table 6. Eigenmodes frequencies in Home Position.

(a) Eigenmodes Frequencies Calculated through Inventor	
Eigenmode index	Frequency
I	151 Hz
II	154 Hz
III	210 Hz
(b) Eigenmodes Frequencies Calculated through Simplified Adams Model	
Eigenmode index	Frequency
I	169 Hz
II	171 Hz
III	222 Hz

A frequency response analysis has been performed on the wind turbine and on the coupled system made by the wind turbine and the Hexaslide. The aim of this simulation is to check the Hexaslide is not influencing the frequencies of the turbine’s mode of vibrating, thus making the presence of Hexaslide negligible at least from a mechanical point of view.

A swept sine load of 100 N amplitude between 0.1 Hz and 250 Hz has been applied to the rotor centre in wind (x) direction and the displacements and rotation have been measured. This Force is able to well excite the first and most relevant eigenmodes of the turbine.

The results are reported in Figure 24. The first two wind turbine eigenfrequencies are preserved even in the coupled system, while two peaks appear between 75 Hz and 100 Hz.

It can be concluded that, in the range of 0–60 Hz, a robot’s dynamics does not influence the turbine eigenfrequencies.

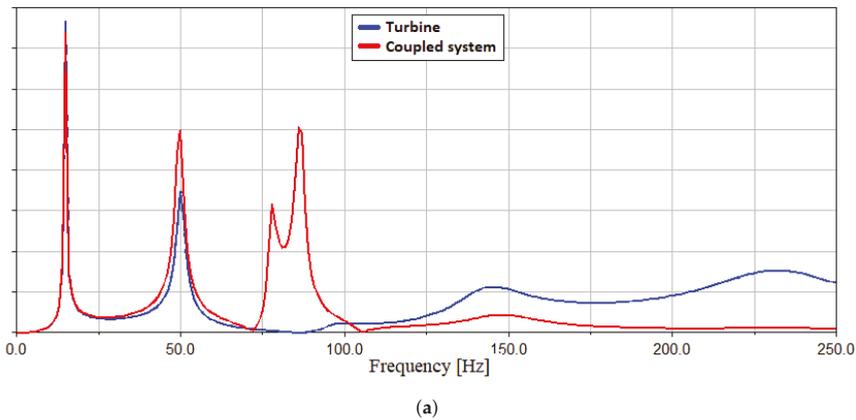


Figure 24. Cont.

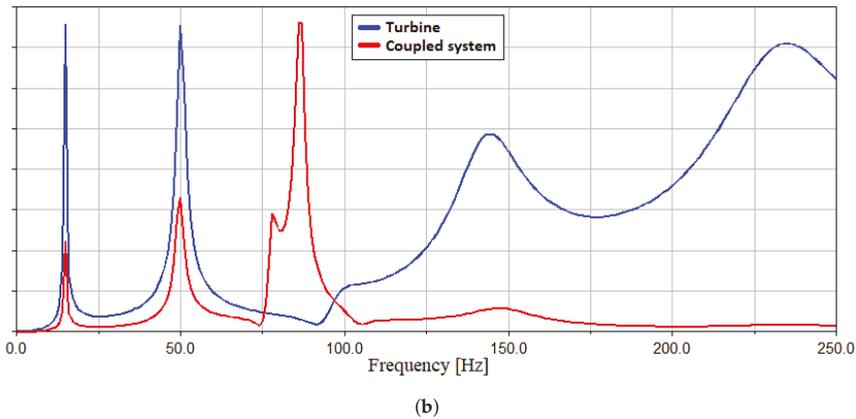


Figure 24. Frequency Response analysis of Turbine and Hexaslide with turbine. (a) displacement along \hat{x}_O direction; (b) rotation around the \hat{y}_O axis.

5. Control Architecture and Electronics

A simplified scheme of the electric panel configuration is reported in Figure 25. The core of the electrical panel is represented by Power PMAC, the property of Delta Tau Data Systems (Chatsworth, CA, USA). There are essentially two zones: one with AC voltage, represented by solid black and blue lines, powered by a 380 V AC line and brings it to the Power PMAC fed at 230 V and to the power circuits of the servo amplifiers fed at 380 V, and the other one with 24 V DC voltage, downstream of the 24 V power supply, which is represented by the solid and dashed green lines. This circuit provides power to the safety relay, limit switches and proximity sensors and to the Beckhoff modules for Ether-CAT communications. Red lines represent the transmission of data between different components.

- **EtherCAT modules:** the Beckhoff EtherCAT module EK1100 is connected to two EL1008 modules, each of them providing eight digital inputs, and to one EL2008 module that makes available eight digital outputs.
- **Servo amplifiers:** their main function is to power the motors according to the signals coming from the motion controller. They also process and gather the feedback signals of the motors' encoders to bring them to Power PMAC. Both signals are transported to and from the servo amplifier into a single cable that is then split into an actual terminal board to be brought to the correct connectors of Power PMAC. Each servo amplifier has the main task of closing the current and phase commutation feedback loop for the motor, starting from the torque reference provided by the Power PMAC motion controller.
- **Power PMAC:** this is the core of the electrical architecture, it is a general-purpose embedded computer with a built-in motion and machine-control application. It also provides a wide variety of hardware machine interface circuits that allow the connection to common servo and stepper drives, feedback sensors, and analogue and digital I/O points.

The modular rack is the most flexible configuration, since it permits the user to choose which CPU card, digital or analogue I/O card, axis interface cards and the like to use in the system. Power PMAC can handle all of the tasks required for machine control, constantly switching back and forth between the different tasks thousands of times per second. On this powerful controller, the main control software of the robot has been designed and implemented. Figure 26 shows the general overview of the software architecture. In particular, it is structured so that a primary and a secondary states machine manages the principal functionality of the system, among which the management of the logic

state (start, stop, homing, jog, and the like) and the functioning of the machine (managing exceptions, motion programs, debug towards the user). The Position-Based-Admittance-Control (PBAC) control scheme used to close the position loop is highlighted in light blue. Further details in this regard are set out in [14].

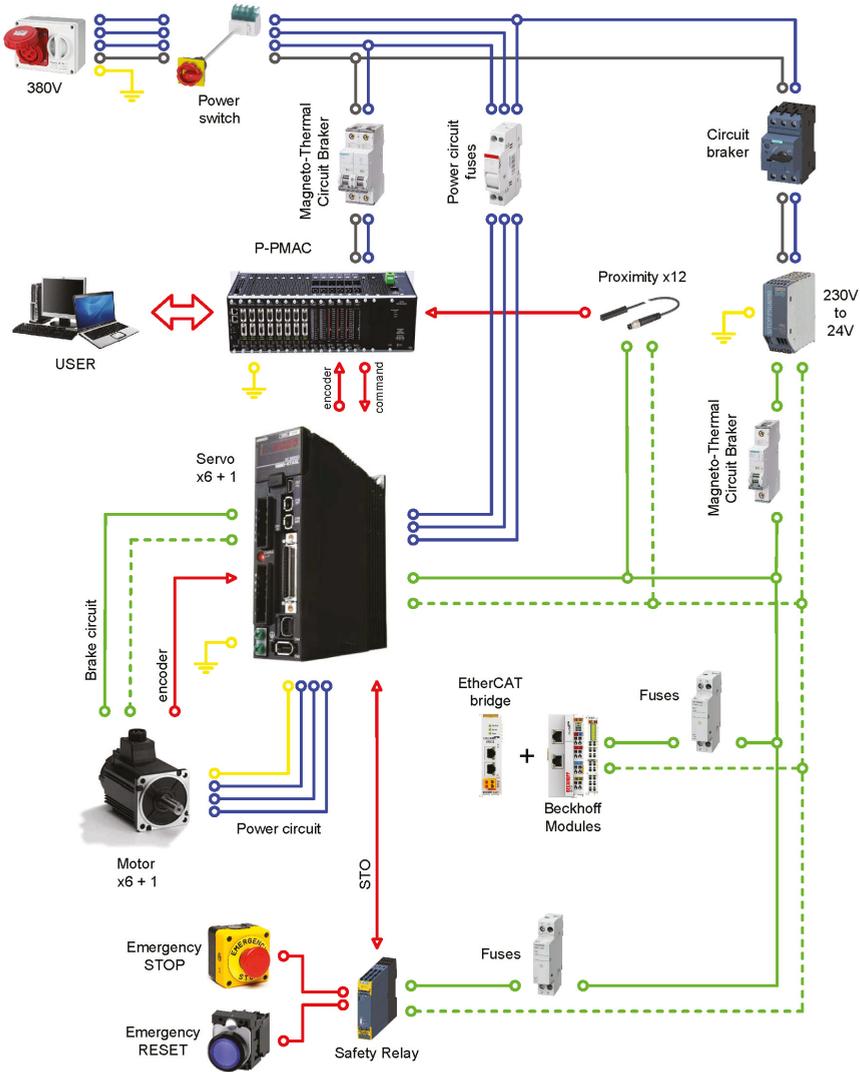


Figure 25. Simplified scheme of an electrical layout.

The lowest level of the control is constituted by the position and velocity servo loops, giving the analogic torque output reference for the seven servo actuator. Each high level control modality developed has been designed to respect real-time performances desired as well as safety, using advanced tools such as position based admittance control, buffering with time-based control, motion look-ahead for smooth blending, fast C written nonlinear FK and demanding algebra, workspace boundaries check

with controlled dynamics on the limits [15], and acceleration saturation with workspace reference tracking check. For more details about control architecture and algorithms, refer to [14,15].

The Human–Machine Interface (HMI) designed allows full control by the user of every significant functioning parameter, simplifying state change and enhancing safety. The HMI communicates with Power PMAC by means of Telnet communication protocol and is completely written in C# language.

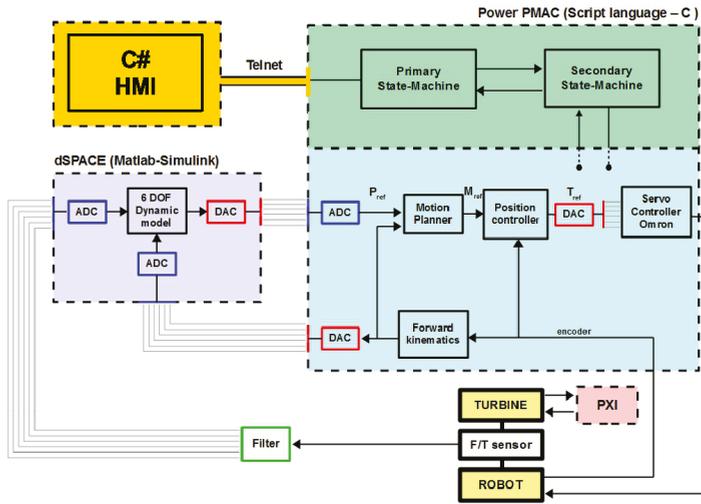


Figure 26. Complete control scheme.

6. Conclusions

The experimental verification of the dynamic response of the robot is currently being finalized. The main control features have been tested on a scaled model of the manipulator with optimum results [14,15]. The full scale is now fully operative and its behaviour is under test. For the sake of completeness in Figure 27, the full scale system equipped with accelerometers during the modal analysis campaign is reported. From the first broad results, it appears confirmed that the robot will not interfere, from a dynamic point of view, with the wind turbine scale model [3].

Further experimental modal analysis is staged for the next months, to fully explore the performances not only of the complete mechanical system, but of the whole machine under control influence. First verifications of control performances have already given satisfying results, but further refinement is done every day to an optimized controlled system behaviour in every required operative condition.

This paper shows the design methodology of the HexaFloat system, a six-DoF robot for wind tunnel hybrid testing of floating offshore wind turbines. This setup consists of a parallel kinematic robot, “HexaFloat”, designed and developed to test the dynamics of floating offshore wind turbine concepts, selected within LIFES50+ project, at the Politecnico di Milano wind tunnel, through a hybrid methodology which combines HIL, in real-time measurements (i.e., aerodynamic forces on the wind turbine scale model) and computations (i.e., hydrodynamic forces on platform). This represents the complementary test approach, with respect to the one developed at the Sintef Ocean basin. The final test campaign for LIFES50+ project is staged for July 2018. The complete HIL setup is currently under testing for performance verification, disturbance influence analysis, control of final refinements and tuning and controllers’ communication optimization.

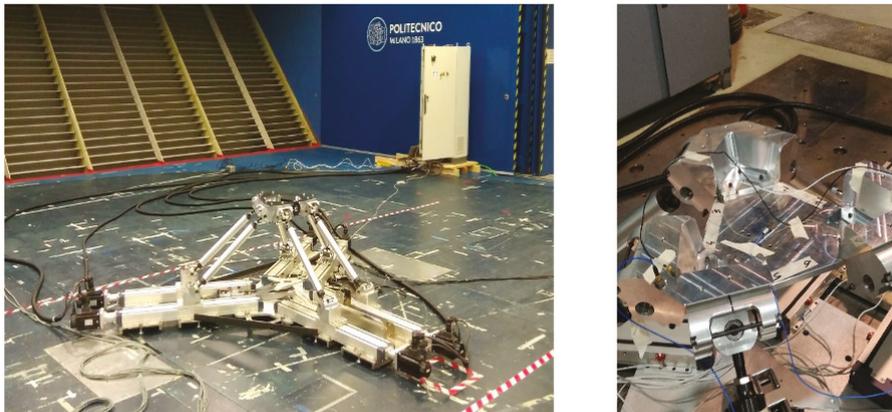


Figure 27. Hexaslide experimental setup.

Author Contributions: H.G. and F.L.M. conceived the devices and the sizing tools used; G.R. and M.P. designed the system; F.L.M. developed the control software of the devices; F.L.M., tested the system. H.G. and F.L.M. wrote the paper.

Funding: This project has received funding from the European Union’s Horizon 2020 research and innovation programme under Grant agreement No. 640741.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Sauder, T.; Chabaud, V.; Thys, M.; Bachynski, E.E.; Sæther, L.O. Real-time hybrid model testing of a braceless semi-submersible wind turbine: Part I—The hybrid approach. In Proceedings of the 2016 35th International Conference on Ocean, Offshore and Arctic Engineering, Madrid, Spain, 17–22 June 2016; American Society of Mechanical Engineers: New York, NY, USA, 2016; pp. V006T09A039.
2. Bayati, I.; Belloli, M.; Ferrari, D.; Fossati, F.; Giberti, H. Design of a 6-DoF robotic platform for wind tunnel tests of floating wind turbines. *Energy Procedia* **2014**, *53*, 313–323. [[CrossRef](#)]
3. Bayati, I.; Belloli, M.; Bernini, L.; Giberti, H.; Zasso, A. Scale model technology for floating offshore wind turbines. *IET Renew. Power Gener.* **2017**, *11*, 1120–1126. [[CrossRef](#)]
4. Giberti, H.; Ferrari, D. A novel hardware-in-the-loop device for floating offshore wind turbines and sailing boats. *Mech. Mach. Theory* **2015**, *85*, 82–105. [[CrossRef](#)]
5. Fiore, E.; Giberti, H. Optimization and comparison between two 6-DoF parallel kinematic machines for HIL simulations in wind tunnel. *MATEC Web Conf.* **2016**, *45*, 04012. [[CrossRef](#)]
6. Ferrari, D.; Giberti, H. A genetic algorithm approach to the kinematic synthesis of a 6-DoF parallel manipulator. In Proceedings of the 2014 IEEE Conference on Control Applications (CCA), Juan Les Antibes, France, 8–10 October 2014; pp. 222–227.
7. Merlet, J. Jacobian, manipulability, condition number, and accuracy of parallel robots. *J. Mech. Des.* **2006**, *128*, 199–206, doi:10.1115/1.2121740. [[CrossRef](#)]
8. Legnani, G.; Tosi, D.; Fassi, I.; Giberti, H.; Cinquemani, S. The “point of isotropy” and other properties of serial and parallel manipulators. *Mech. Mach. Theory* **2010**, *45*, 1407–1423, doi:10.1016/j.mechmachtheory.2010.05.007. [[CrossRef](#)]
9. Vinje, T. The statistical distribution of wave heights in a random seaway. *Appl. Ocean Res.* **1989**, *11*, 143–152, doi:10.1016/0141-1187(89)90024-2. [[CrossRef](#)]
10. Xu, D.; Li, X.; Zhang, L.; Xu, N.; Lu, H. On the distributions of wave periods, wavelengths, and amplitudes in a random wave field. *J. Geophys. Res. C* **2004**, *109*.10.1029/2003JC002073. [[CrossRef](#)]

11. Fiore, E.; Giberti, H.; Bonomi, G. An innovative method for sizing actuating systems of manipulators with generic tasks. *Mech. Mach. Sci.* **2017**, *47*, 297–305, doi:10.1007/978-3-319-48375-7-32. [[CrossRef](#)]
12. Fiore, E.; Giberti, H.; Ferrari, D. Dynamics modeling and accuracy evaluation of a 6-DoF Hexaslide robot. In *Nonlinear Dynamics*; Springer: Berlin, Germany, 2016; Volume 1, pp. 473–479.
13. Giberti, H.; Cinquemani, S.; Legnani, G. A practical approach to the selection of the motor-reducer unit in electric drive systems. *Mech. Based Des. Struct. Mach.* **2011**, *39*, 303–319, doi:10.1080/15397734.2011.543048. [[CrossRef](#)]
14. La Mura, F.; Todeschini, G.; Giberti, H. High Performance Motion-Planner Architecture for Hardware-In-the-Loop System Based on Position-Based-Admittance-Control. *Robotics* **2018**, *7*, 8. [[CrossRef](#)]
15. La Mura, F.; Romanó, P.; Fiore, E.; Giberti, H. Workspace Limiting Strategy for 6 DOF Force Controlled PKMs Manipulating High Inertia Objects. *Robotics* **2018**, *7*, 10. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Technical Note

Nominal Stiffness of GT-2 Rubber-Fiberglass Timing Belts for Dynamic System Modeling and Design

Bozun Wang¹, Yefei Si², Charul Chadha³, James T. Allison¹ and Albert E. Patterson^{1,*}

¹ Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champaign, 117 Transportation Building, 104 South Mathews Avenue, Urbana, IL 61801, USA; bozunw2@illinois.edu (B.W.); jtalliso@illinois.edu (J.T.A.)

² Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, 144 Mechanical Engineering Building, 1206 West Green Street, Urbana, IL 61801, USA; yefeisi2@illinois.edu

³ Department of Aerospace Engineering, University of Illinois at Urbana-Champaign, 306 Talbot Laboratory, 104 South Wright Street, Urbana, IL 61801, USA; charulc2@illinois.edu

* Correspondence: ptttrsv2@illinois.edu; Tel.: +1-217-333-2731

Received: 20 October 2018; Accepted: 19 November 2018; Published: 21 November 2018

Abstract: GT-style rubber-fiberglass (RF) timing belts are designed to effectively transfer rotational motion from pulleys to linear motion in robots, small machines, and other important mechatronic systems. One of the characteristics of belts under this type of loading condition is that the length between load and pulleys changes during operation, thereby changing their effective stiffness. It has been shown that the effective stiffness of such a belt is a function of a “nominal stiffness” and the real-time belt section lengths. However, this nominal stiffness is not necessarily constant; it is common to assume linear proportional stiffness, but this often results in system modeling error. This technical note describes a brief study where the nominal stiffness of two lengths (400 mm and 760 mm) of GT-2 RF timing belt was tested up to breaking point; regression analysis was performed on the results to best model the observed stiffness. The experiments were performed three times, providing a total of six stiffness curves. It was found that cubic regression models ($R^2 > 0.999$) were the best fit, but that quadratic and linear models still provided acceptable representations of the whole dataset with R^2 values above 0.940.

Keywords: timing belt; belt stiffness; dynamic system modeling; mechatronic systems; 3D printers; robotics

1. Introduction

Timing belts are a common means of motion transfer between rotating motors/shafts in a machine or mechatronic system. Many small-to-medium sized mechatronic systems such as 3D printers [1], robots [2,3], desktop computer numerical control (CNC) machines [4], and positioners [5] use such belts, typically in the GT-style [6,7]. GT-style belts are specifically designed to effectively translate rotating motion from pulleys into linear motion with minimal deformation, slippage, and backlash. One of the fundamental characteristics of such a motion transfer system is that the length of the belts changes with time, causing time-variant stiffnesses in the belts which must be considered in dynamic system modeling and design. Note that the “stiffness” in the belt is considered only in the tension direction of the belt for this work, resulting in a stiffness that can be described as a single value or function instead of the full stiffness matrix [8,9].

When analyzing and designing any robotic and other mechatronic systems, it is vital that a good dynamic model of the system be developed and used. Since such systems often use some kind of flexible belts for motion transfer, the belt stiffness is a very important parameter in a system model. In cases where the length of the belt is constant (e.g., running between two fixed pulleys), the stiffness

k of the belt can be modeled as a spring where $f(x) = kx$; therefore, the stiffness of the belt is a function of its deflection x under load. In effect, this constant-length stiffness of the belt is its “nominal” design stiffness. However, in cases where the belt changes length during use, the effective length of these belt is a function of time and, therefore, its stiffness is also time-variant; this time-variant stiffness is the “effective” or apparent stiffness of the belt at some time t . It has been shown that the effective stiffness of the length-changing belts can be directly calculated as a function of the nominal stiffness value, the belt width, and the real-time length of the belt [10] such that:

$$k_i(t) = C_{sp} \frac{b}{L_i(t)} \tag{1}$$

where k_i is the effective stiffness as a function of time, C_{sp} is the nominal stiffness, b is the belt width, and $L_i(t)$ is the length of the belt section at time t . For any case where the length remains constant, the effective and nominal stiffnesses are equal since the value of $L_i(t)$ is a constant. Note that the value of C_{sp} may be a constant or function of material properties for different belt materials; it cannot be considered a function of time the way that the length of the belt is. The most commonly-used GT-style belt is the GT-2; Figure 1 shows the fundamental geometry and specifications for this type of belt.

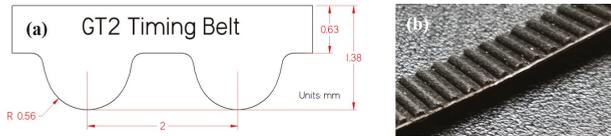


Figure 1. GT-2 belt (a) specifications and (b) basic geometry.

Figure 2 shows a common application, where a GT-style belt is used to transfer motion from a stepper motor to drive a linear positioning system. Also shown is a 2D dynamic model representation of such a system (Figure 2b), where the differences in effective stiffness, based on belt length, in the belt sections are clearly evident. The sections L_1 and L_2 change in effective stiffness as a function of time, while section L_3 stays constant during use [11,12] so the effective and nominal stiffnesses are equal.

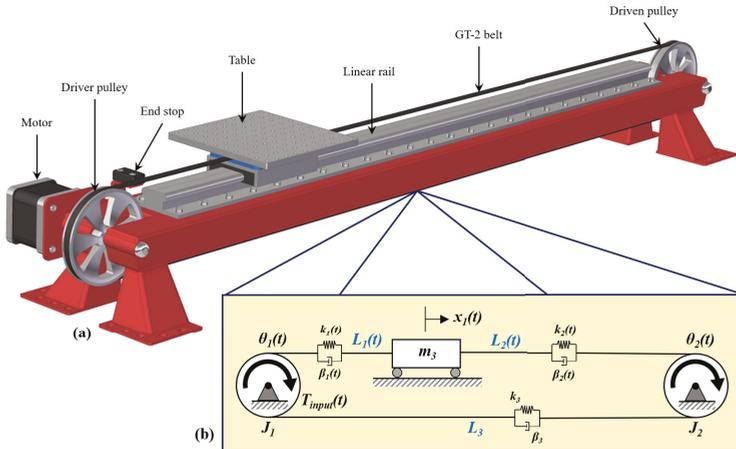


Figure 2. (a) simple positioning system that utilizes a GT-type belt to drive the table and (b) its representative dynamic model.

The work described in this note explored the nominal stiffness C_{sp} and the best way to model it in dynamic systems where belt length is not constant. Several previous studies have assumed that rubber-based timing belts have a linear nominal stiffness [5,11–18]. However, it is vital for designers and engineers working with dynamic systems which use belts for energy transfer to understand the true effects of the belt stiffness [19,20]. Therefore, experimental data was collected and used to derive conclusions on the true stiffness behavior of the GT-2 belts during use. The collected data was subjected to regression analysis to see which type of model best fit, allowing the comparison of models for the same dataset. The information in this study will prove useful, both in choosing k stiffness values for dynamic models and for judging expected model error if linear stiffness assumptions are used.

2. Procedure and Results

Two lengths of new GT-2 belts, 400 mm and 760 mm, were subjected to a simple tensile test until they ruptured. The test apparatus was a custom-built, screw-driven manual desktop test stand set up for tensile testing with 3000 N capability and a travel rate of 1/16 in (1.6 mm) per screw revolution. The screw drive was rotated at a constant rate of 0.5 revolutions per second (0.8 mm/s), a reading being taken every revolution of the screw or every 1.6 mm. Since the length measurement was based on a count of the threads during travel, the uncertainty in length was too small to quantify; the digital readout for the unit used a load cell with a given uncertainty of 100 gram-force or 0.89 N. It was necessary to use this kind of manual tensile testing machine as none of the available standard machines were sensitive enough to measure the force-deflection behavior of these kinds of belts [21]. In addition, the discrete time measurement ensured a reasonably-sized dataset for curve-fitting. This was replicated twice to obtain a set of six different curves, three from each length. The ruptured belts were observed to fail suddenly and to show tearing of the glass fibers inside, as shown in Figure 3. The GT-2 belts used were a composite of neoprene (synthetic rubber) [22] and glass fibers, where the fibers appeared to drive the failure point of the belts.



Figure 3. Belt break interface, showing broken fibers.

The collected data, in terms of force-deflection curves, are shown in Figure 4a, while the equivalent stress-strain curves for the tests are shown in Figure 4b. The length of the belts clearly had an effect on the force-deflection curves, but this largely disappeared when the length was accounted for in the stress-strain curves. Note that most of the curves show hyper-elastic behavior, i.e., there is no region in the curve where the stiffness is constant.

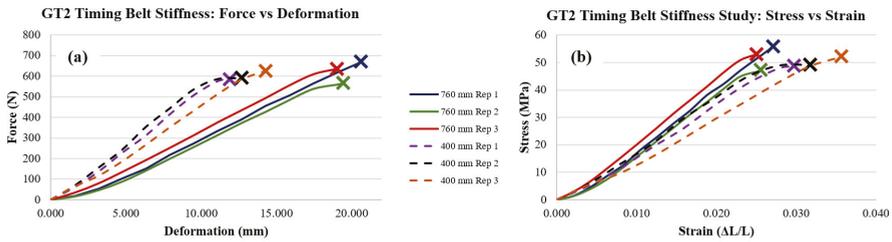


Figure 4. Belt stiffness curves: (a) force-deflection curve and (b) stress–strain curve.

As the nominal compliance of the belts was clearly found to be nonlinear, a regression analysis was performed to model the curves and find the level of unexplained variance in these curves. One of the most common polynomial regression models [23,24] used for hyper-elastic materials is the cubic polynomial. The basic model used for this study began with the following polynomial model:

$$\sigma_{belt} = A\epsilon_{belt}^3 + B\epsilon_{belt}^2 + C\epsilon_{belt} + D \tag{2}$$

where a cubic model includes all of the variables, a quadratic model can be generated by setting $A = 0$, and a linear model can be used with $A = B = 0$. These curve fits, completed using Microsoft Excel™ (Microsoft Corp, Redmond, WA, USA) are shown in Figure 5a, and the fits for each of the variable and the resulting R^2 values are shown in the first six cases in Table 1.

After fitting the cubic models to each of the six sets of experimental data, the cubic model, a quadratic model, and a linear model were then fit to the entire set at once, as shown in Figure 5b. A significant drop in the R^2 value was noted for all of the models fit to the dataset, but differences between the cubic, quadratic, and linear models were observed to be small, as shown in Table 1.

It was observed that the low-strain region of the dataset (Figure 5b,c) conforms better to a linear model when the entire dataset is used. In actual use, it is most likely that the belts will not reach more than 20–30% of the belt breaking strength during normal use [14,25,26], so this is a valid assumption for many systems; this will, of course, need to be determined by the modeler or designer before using a linear belt model. If the low-strain assumption can be used, then the data fit a linear model with a slightly greater R^2 value than a quadratic model for the entire dataset and is certainly superior to a linear model for the entire dataset. The linear model for this case is shown in the last row of Table 1.

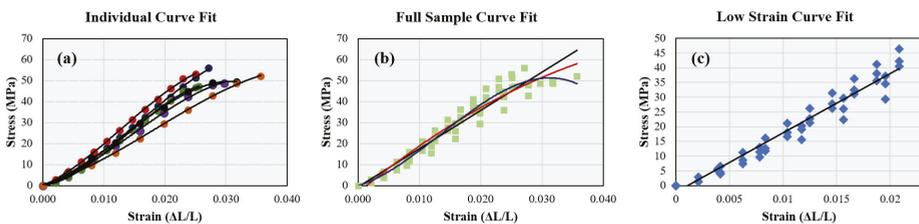


Figure 5. Curve fits for (a) individual belts (cubic model); (b) full sample curve fit (cubic, quadratic, and linear models); and (c) low-strain linear curve fit.

Table 1. Belt stiffness model curve fit data.

Case	Plot Reference	A	B	C	D	R ²
760 mm (cubic model) - R1	Figure 5a	-2.00×10^6	94,969	958.80	-0.5658	0.9996
760 mm (cubic model) - R2	Figure 5a	-3.00×10^6	144,204	445.86	-0.1163	0.9996
760 mm (cubic model) - R3	Figure 5a	-2.00×10^6	95,693	1281.60	-0.0723	0.9997
400 mm (cubic model) - R1	Figure 5a	-2.00×10^6	81,219	849.99	0.2296	0.9993
400 mm (cubic model) - R2	Figure 5a	-2.00×10^6	95,332	935.37	0.2840	0.9995
400 mm (cubic model) - R3	Figure 5a	-922,283	50,993	810.95	0.4965	0.9994
Full dataset (cubic model)	Figure 5b	-2.00×10^6	98,091	937.22	-0.1758	0.9672
Full dataset (quadratic model)	Figure 5b	-	-19,340	2408.9	-3.3656	0.9552
Full dataset (linear model)	Figure 5b	-	-	1821.1	-0.6140	0.9431
Low strain (linear model)	Figure 5c	-	-	2013.8	-2.3275	0.9573

3. Recommendations for Use and Applications

In cases where a time-variant belt length is used in a dynamic system model, the time-dependent stiffness of the belt must be considered, even when a mix of time-variant and time-invariant belt lengths are used. In practice, it is recommended that the modeler follow a three-step procedure:

1. Identify the nominal stiffness C_{sp} of each belt type used in the system (e.g., if two thicknesses of belts are used, two different nominal stiffnesses will be present). This information may be collected from manufacturer datasheets or from tests on each belt type, similar to the tests done in this technical report.
2. Decide if a linear or nonlinear nominal stiffness C_{sp} model will be used for each belt type. The primary driving force for this decision will be the computational cost for analyzing the system; for a simple system, it may be practical to use a nonlinear nominal stiffness model, but a linear model would be more feasible in a system with several elements. However, the importance of the model accuracy is a serious consideration and may justify a high computational cost if high accuracy is required.
3. Based on the configuration of the system and the decisions made in the first two steps, the effective stiffness k can take one of four forms:

- (a) If the belt length is constant and a linear model is used for C_{sp} , the effective stiffness in the equations of motion will be constant and described by

$$k_i = C_{sp} \frac{b}{L} \tag{3}$$

- (b) If the belt length is constant and a nonlinear model is used to find C_{sp} , the nominal stiffness will be a function derived from a force-deflection curve. The effective stiffness in that belt section will be described by

$$k_i = C_{sp}(x) \frac{b}{L} \tag{4}$$

where $C_{sp}(x)$ is a continuous function of x .

- (c) If the belt length is time-variant and a linear model is used for C_{sp} , the effective stiffness in the equations of motion will be time-variant and described by

$$k_i = C_{sp} \frac{b}{L(t)} \tag{5}$$

- (d) If the belt length is time-variant and a nonlinear model is used to find C_{sp} , the nominal stiffness will be a function derived from a force-deflection curve. In this case, the effective belt section stiffness will be described by

$$k_i = C_{sp}(x) \frac{b}{L(t)} \quad (6)$$

where $C_{sp}(x)$ is a continuous function of x and the belt length is a function of time. Therefore, the effective stiffness will be dependent on both the length of the belt and the amount of force placed on the belt.

When modeling these dynamic systems, it is recommended that the simplest model of the belt stiffness which gives acceptable accuracy be used in order to balance computational cost with extreme accuracy in the model. In most cases, the uncertainty in the material properties of the belt and the common use of linearization in dynamic models would erase any advantage to using an extremely high-fidelity belt model.

4. Conclusions

This short technical note presents the results of a brief exploratory study on modeling the nominal stiffness of GT-2 timing belts; this information can be used to more accurately model the true, time-variant, stiffness behavior of common GT-2 belts when the effective length of belt sections changes with time. It was observed that these belts do not behave in a linear way, as expected for belts with a hyper-elastic base material, but that a linear model can provide a reasonable approximation of the behavior under some conditions, particularly low-strain conditions. When possible, the cubic stiffness model should be used, but this would often be impractical for dynamic systems with many components, as it can cause a simple model to become nonlinear in more than one variable. When practical and necessary for problem tractability, a linear model may be used with a reasonable degree of accuracy. The modeler or designer should keep in mind that some uncertainty will exist with any belt model and should choose the model that best balances accuracy with computational cost.

Author Contributions: B.W. and A.E.P. conceived and designed the study. All authors helped to set up the experiments, collect data, perform the regression analyses, and write the report.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest. No external funding was used to perform the work described in this study. Opinions and conclusions presented in this work are solely those of the authors.

Nomenclature

b = Belt width (m)

β_i = Belt section i damping coefficient

C_{sp} = Nominal belt stiffness (N/m)

k_i = Effective (true) belt section i stiffness (N/m)

L_i = Belt section i length (m)

m_i = Mass of block i (kg)

θ_i = Pulley i angle (degrees)

References

1. Laureto, J.; Pearce, J. Open Source Multi-Head 3D Printer for Polymer-Metal Composite Component Manufacturing. *Technologies* **2017**, *5*, 36. [[CrossRef](#)]
2. Krahn, J.; Liu, Y.; Sadeghi, A.; Menon, C. A tailless timing belt climbing platform utilizing dry adhesives with mushroom caps. *Smart Mater. Struct.* **2011**, *20*, 115021. [[CrossRef](#)]

3. Parietti, F.; Chan, K.; Asada, H.H. Bracing the human body with supernumerary Robotic Limbs for physical assistance and load reduction. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014. [CrossRef]
4. Choudhary, R.; Sambhav; Titus, S.D.; Akshaya, P.; Mathew, J.A.; Balaji, N. CNC PCB milling and wood engraving machine. In Proceedings of the International Conference On Smart Technologies For Smart Nation (SmartTechCon), Bangalore, India, 17–19 August 2017. [CrossRef]
5. Sollmann, K.; Jouaneh, M.; Lavender, D. Dynamic Modeling of a Two-Axis, Parallel, H-Frame-Type XY Positioning System. *IEEE/ASME Trans. Mechatron.* **2010**, *15*, 280–290. [CrossRef]
6. York Industries. York Timing Belt Catalog: 2mm GT2 Pitch (pp. 16). Available online: http://www.york-ind.com/print_cat/york_2mmGT2.pdf (accessed on 13 June 2018).
7. SDP/SI. Handbook of Timing Belts, Pulleys, Chains, and Sprockets. Available online: www.sdp-si.com/PDFS/Technical-Section-Timing.pdf (accessed on 13 June 2018).
8. Huang, J.L.; Clement, R.; Sun, Z.H.; Wang, J.Z.; Zhang, W.J. Global stiffness and natural frequency analysis of distributed compliant mechanisms with embedded actuators with a general-purpose finite element system. *Int. J. Adv. Manuf. Technol.* **2012**, *65*, 1111–1124. [CrossRef]
9. Barker, C.R.; Oliver, L.R.; Breig, W.F. *Dynamic Analysis of Belt Drive Tension Forces During Rapid Engine Acceleration*; SAE Technical Paper Series; SAE International: Warrendale, PA, USA, 1991. [CrossRef]
10. Gates-Mectrol. Technical Manual: Timing Belt Theory. Available online: http://www.gatesmectrol.com/mectrol/downloads/download_common.cfm?file=Belt_Theory06sm.pdf&folder=brochure (accessed on 13 June 2018).
11. Hace, A.; Jezernik, K.; Sabanovic, A. SMC With Disturbance Observer for a Linear Belt Drive. *IEEE Trans. Ind. Electron.* **2007**, *54*, 3402–3412. [CrossRef]
12. Johannesson, T.; Distner, M. Dynamic Loading of Synchronous Belts. *J. Mech. Des.* **2002**, *124*, 79. [CrossRef]
13. Childs, T.H.C.; Dalgarno, K.W.; Hojjati, M.H.; Tutt, M.J.; Day, A.J. The meshing of timing belt teeth in pulley grooves. *Proc. Inst. Mech. Eng. D* **1997**, *211*, 205–218. [CrossRef]
14. Callegari, M.; Cannella, F.; Ferri, G. Multi-body modelling of timing belt dynamics. *Proc. Inst. Mech. Eng. K* **2003**, *217*, 63–75. [CrossRef]
15. Leamy, M.J.; Wasfy, T.M. Time-accurate finite element modelling of the transient, steady-state, and frequency responses of serpentine and timing belt-drives. *Int. J. Veh. Des.* **2005**, *39*, 272. [CrossRef]
16. Feng, X.; Shangguan, W.B.; Deng, J.; Jing, X.; Ahmed, W. Modelling of the rotational vibrations of the engine front-end accessory drive system: a generic method. *Proc. Inst. Mech. Eng. D* **2017**, *231*, 1780–1795. [CrossRef]
17. Rodriguez, J.; Keribar, R.; Wang, J. *A Comprehensive and Efficient Model of Belt-Drive Systems*; SAE Technical Paper Series; SAE International: Warrendale, PA, USA, 2010. [CrossRef]
18. Cepon, G.; Boltezar, M. *An Advanced Numerical Model for Dynamic Simulations of Automotive Belt-Drives*; SAE Technical Paper Series; SAE International: Warrendale, PA, USA, 2010. [CrossRef]
19. Tai, H.M.; Sung, C.K. Effects of Belt Flexural Rigidity on the Transmission Error of a Carriage-driving System. *J. Mech. Des.* **2000**, *122*, 213. [CrossRef]
20. Zhang, L.; Zu, J.W.; Hou, Z. Complex Modal Analysis of Non-Self-Adjoint Hybrid Serpentine Belt Drive Systems. *J. Vib. Acoust.* **2001**, *123*, 150. [CrossRef]
21. *Materials Testing Guide*; ADMET: Norwood, MA, USA, 2013.
22. Kumar, D.; Sarangi, S. Data on the viscoelastic behavior of neoprene rubber. *Data Brief.* **2018**, *21*, 943–947. [CrossRef] [PubMed]
23. Mansouri, M.; Darijani, H. Constitutive modeling of isotropic hyperelastic materials in an exponential framework using a self-contained approach. *Int. J. Solids Struct.* **2014**, *51*, 4316–4326. [CrossRef]
24. Shahzad, M.; Kamran, A.; Siddiqui, M.Z.; Farhan, M. Mechanical Characterization and FE Modelling of a Hyperelastic Material. *Mater. Res.* **2015**, *18*, 918–924. [CrossRef]

25. Tokoro, H. Analysis of transverse vibration in engine timing belt. *JSAE Rev.* **1997**, *18*, 33–38. [[CrossRef](#)]
26. Gerbert, G.; Jnsson, H.; Persson, U.; Stensson, G. Load Distribution in Timing Belts. *J. Mech. Des.* **1978**, *100*, 208. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Author Index

A

Ahmmad, Syed Masrur 173
Allison, James T. 215

B

Betancor Bosch, Javier 83
Boschetti, Giovanni 127
Beschi, Manuel 153
Billah, Md Masum 173

C

Ciupitu, Liviu 71
Carbonari, Luca 103
Callegari, Massimo 103
Cattai, Mattia 111
Chadha, Charul 215

D

Del Vescovo, Dionisio 57
Diaz De Cerio Sanchez, Alejandro 83
Di Gregorio, Raffaele 111

F

Fassi, Irene 43
Faroni, Marco 153

G

Giorgio, Ivan 57
Giberti, Hermes 185

H

Huq, Naim Md Lutful 173

K

Khan, Md Raisuddin 173

L

Legnani, Giovanni 43
Liu, Guojun 143
La Mura, Francesco 185

N

Nawratil, Georg 25

P

Palpacelli, Matteo 103
Palmieri, Giacomo 103
Pedrocchi, Nicola 153
Parmeggiani, Marco 185
Patterson, Albert E. 215

R

Rigi, Amin 83
Resmini, Gabriele 185

S

Simas, Henrique 111
Shafie, Amir Akramin 173
Si, Yefei 215

T

Tsiotras, Panagiotis 1
Trevisani, Alberto 127

V

Valverde, Alfredo 1
Vitzilaios, Nikolaos 83
Visioli, Antonio 153

W

Wang, Bozun 215

Z

Zweiri, Yahya 83

MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland
Tel. +41 61 683 77 34
Fax +41 61 302 89 18
www.mdpi.com

Robotics Editorial Office
E-mail: robotics@mdpi.com
www.mdpi.com/journal/robotics



MDPI
St. Alban-Anlage 66
4052 Basel
Switzerland

Tel: +41 61 683 77 34
Fax: +41 61 302 89 18

www.mdpi.com



ISBN 978-3-0365-1019-4