UNIVERSITY OF CALIFORNIA,

IRVINE

Dual Quaternion Synthesis of Constrained Robotic Systems

DISSERTATION

submitted in partial satisfaction of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in Mechanical and Aerospace Engineering

by

Maria Alba Perez

Dissertation Committee:

Professor J. Michael McCarthy, Chair

Professor James E. Bobrow

Professor Bahram Ravani

2003

The dissertation of Maria Alba Perez

is approved and is acceptable in quality

and form for publication on microfilm:

_____

_____

_____

Committee Chair

University of California, Irvine

2003

Dedication

To my family.

Table of Contents

List of Figures

List of Tables

Nomenclature

Roman symbols

$\mathbb{E}^n$	The Euclidean space of dimension n.
$\mathbb{P}^n$	The projective n-space.
$\mathcal{C}(V)$	The Clifford algebra of a vector space V
$[M]$	A matrix.
$\mathbf{s}$	A vector; also a three-dimensional point.
S	A dual vector; $\mathsf{S} = \mathbf{s} + \epsilon\mathbf{s}^0$.
$\hat{Q}$	A dual quaternion.
$\hat{q}$	A dual number.
Q	A quaternion.
ϵ	The dual unit such that $\epsilon^2 = 0$.
$\times$	The cross-product of three-dimensional vectors.
$\cdot$	The dot product of vectors.
$\wedge$	The wedge product of the algebra of multi-vectors.
$\rtimes$	The semi-direct product in group theory.
$[Z(\theta, d)]$	A screw displacement about the Z axis, of rotation θ and translation d.

Acknowledgements

Many people have contributed, in different ways, to the development of this research and, unfortunately, I cannot take the space to thank them all.

My family, back in Spain, deserves the first and warmest thanks for their continuous love and support.

The path that led me to the University of California was enabled by the generosity of Pete Balsells, and the dedication of Professor Roger Rangel. I am grateful to the team that created the Balsells fellowship program, formed by the Balsells family, the University of California and the government of Catalonia, for providing not only with the money but also with the guidance to reach this academic institution. I also acknowledge the further financial support provided by the National Science Foundation.

My research here could not have been developed without the guidance, enthusiasm and deep knowledge of my advisor, Professor J. Michael McCarthy. I am really in debt with him for helping me opening a tiny door to a huge new world. Neither can I forget the friendship and guidance of Professor Bruce Bennett, who helped me in key moments of my research and who is now sorely missed. I am also thankful to Dr. Curtis Collins, who offered his help when I needed it.

I want to thank the members of my Ph.D. committee, Professor James Bobrow and Professor Bahram Ravani, for their interest in my research and for their useful remarks. Broader thanks go to the rest of faculty and staff of the Mechanical and Aerospace Engineering department. And for their camaraderie and our shared enthusiasm for engineering, I want to thank my fellows from the Balsells fellowship and my colleagues of the Robotics and Automation laboratory, especially Hai-Jun Su and Juanita Albro.

The last thank you goes to my husband, Gaurav Lachhwani, for his sincere support and his invaluable friendship.

Curriculum Vitae

Maria Alba Perez

Education

09/2003 **Ph.D.** in Mechanical and Aerospace Engineering.
Department of Mechanical and Aerospace Engineering.
University of California, Irvine.
Irvine, CA.
Advisor: J. Michael McCarthy.

03/1999 **M. S.** in Mechanical and Aerospace Engineering.
Department of Mechanical and Aerospace Engineering.
University of California, Irvine.
Irvine, CA.
Advisor: J. Michael McCarthy.

06/1992 **B. S.** in Industrial Engineering.
Department of Mechanical Engineering.
Escola Tècnica Superior d'Enginyers Industrials de Barcelona.
Polytechnic University of Catalonia (UPC).
Barcelona, Spain.

Experience

03/1999 - 09/2003 Robotics and Automation Laboratory, UCI, Ph.D. advisor: Prof. J. M. McCarthy: **Research Assistant** — Design of spatial mechanisms and robotic systems.

04/2001 - 06/2001 and 04/2000 - 06/2000 Dept. of Mechanical Engineering, UCI, Course Professor: J. M. McCarthy: **Teaching Assistant**, MAE 145: Theory of Machines and Mechanisms (Spring 2001, Spring 2000).

09/1999 - 12/1999 Dept. of Mechanical Engineering, UCI, Course Professor: T. S. Hristov: **Teaching Assistant**, MAE 30: Applied Mechanics: Statics (Fall 1999).

04/1999 - 06/1999 Dept. of Mechanical Engineering, UCI, Course Professor: J. M. McCarthy: **Teaching Assistant**, MAE 183: Computer Aided Mechanism Design (Spring 1999).

03/1997 - 08/1997 MAI, United Technologies Automotive: **Design Engineer**. Design of housings for electrical automotive components. Address: Ctra. Igualada, km 1,5 Pol. Industrial - P.O. Box 106, 43800 Valls (Spain).

12/1996 - 03/1997 Bitron Industrie Espanya, S.A.: **Design Engineer** . Design of temperature and pressure sensors for appliances and automotive industry. Address: Bitron Industrie Espanya, C/ Ifni, 24 30, 08930 Sant Adria del Besos, Barcelona (Spain).

01/1994 - 06/1996 Waste Agency, Departament de Medi Ambient, Catalonia, Spain: **Planning engineer**— Development of the regional planning for the minimization, recycling and disposal of the industrial waste. Address: C/ Doctor Roux, 80, 08017 Barcelona (Spain).

09/1992 - 06/1996 Cybernetics Institute, ETSEIB, Polytechnic University of Catalonia, Advisor: Dr. Luis Basanez Villaluenga: **Research Assistant** — Mechanical design of a robotics platform system.

09/1990 - 03/1992 Material Science Department, ETSEIB, Polytechnic University of Catalonia, Advisor: Dr. Antoni Martinez: **Research Assistant** — Characterization of mechanical properties for various plastic materials.

09/1991 - 12/1991 and 09/1990 - 12/1990 Material Science Department, ETSEIB, Polytechnic University of Catalonia: **Teaching Assistant**, – Lectures on the use of machinery for casting and machining plastic materials.

Awards

. Balsells Fellowship, two-year fellowship from the University of California, the government of Catalonia and the Balsells family, for pursuing graduate studies at the University of California, Irvine.

. NSF Travel Grant to present paper: "Dual Quaternion Synthesis of a Parallel 2-TPR Robot", Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators, October 3-4, 2002, Québec City, Québec, CA.

Publications

1. A. Perez and J.M. McCarthy, "Dimensional Synthesis of Bennett Linkages", *ASME Journal of Mechanical Design*, **125**(1):98-104, March 2003.

2. A. Perez and J.M. McCarthy, "Dual Quaternion Synthesis of Constrained Robotic Systems", accepted at the *Journal of Mechanical Design*, submitted February 2003.

3. A. Perez and J.M. McCarthy, "Dimensional Synthesis of CRR Serial Chains", *ASME Design Engineering Technical Conferences*, Chicago, IL, September 2003.

4. A. Perez and J.M. McCarthy, "Dimensional Synthesis of RPC Serial Robots", *International Conference on Advanced Robotics, ICAR 2003*, Coimbra, Portugal, June 2003.

5. A. Perez and J.M. McCarthy, "Bennett's Linkage and the Cylindroid", *Mechanism and Machine Theory*, **37**(11):1245-1260, November 2002.

6. A. Perez and J.M. McCarthy, "Dual Quaternion Synthesis of a Parallel 2-TPR Robot", *Proc. of the Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators*, October 3-4, 2002, Quebec City, Quebec, CA.

7. A. Perez and J.M. McCarthy, "Dual Quaternion Synthesis of Constrained Robots", *Advances in Robot Kinematics, (J. Lenarcic and F. Thomas,eds.)*, pp. 443-452, Kluwer Academic Publ.,Netherlands, 2002.

8. C.L. Collins, J.M. McCarthy, A. Perez, and H. Su, "The Structure of an Extensible Java Applet for Spatial Linkage Synthesis", *ASME Journal of Computing and Information Science and Engineering*, **2**(1):45-49, 2002.

9. A. Perez and J. M. McCarthy, "Dimensional Synthesis of Bennett Linkages", *Proceedings of the ASME Design Engineering Technical Conferences*, Baltimore, MD, Sept. 10-13, 2000.

10. A. Perez and J. M. McCarthy, "Dimensional Synthesis of Spatial RR Robots", *Advances in Robot Kinematics, (J. Lenarcic and M.M. Stanisic, eds.)*, pp. 93-102, Kluwer Academic Publ., Netherlands, June 2000.

Abstract of the Dissertation

Dual Quaternion Synthesis of Constrained Robotic Systems

By

Maria Alba Perez

Doctor of Philosophy in Mechanical and Aerospace Engineering

University of California, Irvine, 2003

Professor J. Michael McCarthy, Chair

Constrained robotics systems are serial or parallel robots with less than six degrees of freedom. Dimensional synthesis is defined as the process of dimensioning a robot, that is, designing the link dimensions for a given task or set of tasks. In finite-position synthesis, we define the task as a series of positions that the robot must reach.

Dimensional synthesis of planar mechanisms was first solved using graphic methods, and later those methods were transformed into algebraic equations that described the constraints on the movement of the mechanism. This approach was successfully applied to spherical mechanisms and simple cases of spatial mechanisms. The methodology was not extended to general constrained robots due to the difficulty in stating the geometric constraints for robots with more than three links.

A systematic approach for the synthesis of spatial robots was developed based on using the kinematics equations of the robot. The kinematics equations are spatial transformations from a fixed frame to the end-effector of the robot, parameterized by

both the dimensions of the links and the joint variables.

In this dissertation, a method for the kinematic synthesis of constrained robots is presented. It is based on the use of dual quaternions to construct the kinematics equations of the robot from a reference position and to equate them to a set of task positions. A calculation was devised to compute the maximum number of task positions for each robot topology, and a classification of constrained robots was obtained according to this.

The design equations produced using this methodology have been solved numerically for both the link dimensions and the joint variables, and also a scheme has been introduced to eliminate the joint variables in order to obtain algebraic equations. These have been further simplified to closed algebraic expressions in several cases.

The dual quaternion synthesis methodology provides with a tool for the systematic design of constrained robots. Some of these results have been implemented in computer-aided design systems.

Chapter 1

Introduction

1.1 Overview

The focus of this research is in developing the mathematical and computational core of interactive systems for the computer-aided design of robots. The key for this development is the use of the kinematics theory for the analysis of multi-body systems, which was developed for its application to robotics, within the finite position synthesis theory for the design of spatial linkage systems. The former has systematic tools for analyzing any open-chain topology and many closed-chain robots. The kinematic synthesis of spatial linkages has been generalized from the initial planar theory to the space for some fundamental primitives, which constitute the component pieces of most robotic systems. The solution for the synthesis of the basic constrained chains and the connection of different solutions to form more complex systems will be the basis to develop synthesis procedures for robotic systems for the exact fitting of the workspace to a specified task manifold of dimension lower than six.

1.2 Kinematics of Robots

The term *kinematics* defines the branch of mechanics which deals with the motion without considering the forces associated to it. It includes position and its spatial derivatives: velocities, accelerations and higher derivatives.

A robot can be defined as a mechanical system that can be programmed to perform a number of tasks involving movement under automatic control. The main characteristic of a robot is its capability of movement, in general in a six-dimensional space that includes translational and rotational coordinates.

A mechanical system consists of a series of rigid links connected by joints. The joints allow the relative movement of adjacent links, and are generally powered and equipped with systems to control the movement. The most common types of joints in robotic systems are revolute joints, which allow rotary movement between links, and prismatic joints, which allow linear movement between consecutive links.

The mobility or number of degrees of freedom of the mechanism is defined as the minimum number of coordinates needed to specify the positions of all members of the system relative to a particular base frame. The criteria to find the mobility of mechanisms was first developed by Gruebler (1851-1935) and has been posteriorly expanded to spatial linkage systems, see [9]. In the case of open chains, the number of degrees of freedom is equal to the number of joints of the chain, while for parallel mechanisms other factors have to be considered, see [29], [80].

The study of the kinematics of mechanical systems dealt mainly with planar mech-

anisms and graphical methods until the kinematics of robots started to develop in the second half of the 20^{th} century. See for instance [21] for an introduction to the history of the theory of machines.

1.3 Kinematic Analysis

In its origins, the kinematics of robots focused mainly in analyzing the kinematics properties for a given robot structure. Research was devoted to extend the mathematical concepts of planar kinematics to the space, see Bottema [9] and Veldkamp [103]. In 1955, Denavit and Hartenberg [20] introduced the methodology of $4x4$ homogeneous matrix transformations to analyze robotic systems, which became a standard tool for the description of robotics systems [13], and especially in dealing with the forward and inverse kinematics, which relate the position of the end-effector to the values of the joint angles along the kinematic chain.

The solutions of the direct kinematics define the workspace of the robot. The concept of workspace of a robotic system was developed by Roth [85], Kumar and Waldron [41], Tsai and Soni [100] and Gupta and Roth [25] among others. The workspace of a mechanical system is the set of all positions (translations and orientations) that it can reach. There are many ways of representing the workspace of a robot. For the purpose of the kinematic analysis, one of the most common ways is to give the workspace as a $4x4$ homogeneous matrix. This describes the spatial transformations from a frame attached to the end-effector of the robot to a reference

frame as consecutive displacements along the kinematic chain. There exist other representations for the workspace, which fit better for different purposes. For a summary of different representations of displacements applied to robotics see [92].

Screw theory captures the geometry of Lie algebras and has been widely applied in kinematic analysis for describing the workspace of mechanisms, characterizing their mobility and especially for studying the instantaneous properties of the systems. For finite displacements, the first notions of screw theory can be found in the work of Chalses in the early 1800s. Chalses proved that a general displacement of a rigid body can be expressed as a rotation about a line followed by a translation along the same line, called a *screw motion*. Rodrigues (circa 1840) [6] formulated the expression for the rotation axis of a composite rotation. Screw theory was developed by Ball [7] and later applied to the kinematics of mechanisms by Dimentberg [22] and Hunt [37]. Its application to finite kinematics was further developed by Tsai [97], Bottema and Roth [9], Roth [81], Huang [32], [34], [36], and Parkin [69].

The effort of generalizing the representation of planar rotations by mean of complex numbers led to the invention of quaternions by Hamilton [28]. Dual quaternions were first introduced by Clifford [11] as a combination of the real quaternions with Clifford's dual numbers, and further developed by Study and McAulay around 1890. Dual quaternions form a Clifford algebra and represent spatial transformations.

Yang and Freudenstein [107] introduced the use of dual quaternions for the analysis of spatial mechanisms, based on the earlier work of Denavit and Hartenberg and

Dimentberg. Since then, dual quaternions and their planar version have been used widely in the analysis of robotic systems and computer graphics, see for instance the works of Murray[62], McCarthy [53], [23], Ravani and Ge [78], and others, see [106], [18], [31], [3].

All these analysis tools, combined with numerical and computational methods, have been used to solve most of the analysis problems.

1.4 Kinematic Synthesis

In the kinematic synthesis, a mechanism is designed "to meet certain motion specifications" [30]. We differentiate between *type synthesis* and *dimensional synthesis*. In type synthesis the designer decides about the type and number of joints of the chain [2] (to design for the number of joints was formerly called *number synthesis*). In dimensional synthesis the dimensions of the links are calculated to fit a specified motion.

One of the first works in defining the synthesis of mechanisms, in particular defining the type synthesis, is due to Franz Releaux (around 1875) [79]. He identified the fact that the relative motion between two bodies depends on the surfaces of contact. The two contacting elements constitute a kinematic pair; Releaux identified two types of pairs: higher pairs, in which the two bodies are in contact at a point or along a line, and lower pairs, in which the bodies share a surface of contact. This classification of joints as higher and lower pairs is still in use and will be presented more in detail in

Chapter 2. The first systematic enumeration of mechanical components is also due to Releaux; he identified six basic types of components used to create a mechanism. See [30] for more details.

The dimensional synthesis of mechanisms was historically divided in three categories: function generation, where a single input-single output was synthesized to follow a certain input-output function; path generation, in which the mechanism must reach a set of prescribed points; and rigid body guidance, where the mechanism is designed to reach a set of positions consisting of translation plus orientation. However, it is possible to express the first two problems as rigid body guidance problems, see [96]. When the mechanism is forced to reach exactly the specified set of positions, the rigid body guidance problem is called *exact position synthesis*.

Burmester (around 1888) developed geometric methods both for analysis and dimensional synthesis. His dimensional synthesis method was based on finding a finite number of precision points representing locations of the moving body that comply with the movement of the chain. Similar results were developed as mathematical problems by Schoenflies around the same time.

Freudenstein transformed the geometric methods of what is now called Burmester's theory into analytical equations, see for instance his work with Primrose and Sandor [76], and started the use of computers to generate the solutions, as well as the use of optimization methods [84]. The synthesis of planar mechanisms has been studied also by Hartenberg and Denavit [30], Sandor and Erdman [88], and Roth [82].

The exact synthesis for planar serial chains deals with only a reduced number of cases, those that consist of prismatic and / or revolute joints and have less than three degrees of freedom. The finite position synthesis theory was generalized to spherical linkages, see McCarthy [58]. The synthesis of constrained spherical robots deals also with a reduced number of mechanisms.

The two main techniques used for the planar synthesis are the geometric constraint and the complex number methods [88].

The *geometric constraint* method imposes constraints to the movement of the end-effector based on the structure of the chain. This yields algebraic equations that are solved to determine the dimensions of the linkage. This method has been used to solve several spatial dyads, among others the spatial RR chain by Suh [94]; spatial CC chain by McCarthy [54], Huang and Chang [35], Vance and Larochelle [102]; the SS chain by Innocenti [38], who found a polynomial solution for this problem, and Liao and McCarthy [48], who combined the solutions of the finite-dimensional syntehsis to form a 5-SS platform. This approach was applied to two- and three-jointed chains ending in S joints, such as the RPS and PPS chains as well as the CS chain, by Chen and Roth [12]. Nielsen and Roth [64] studied elimination methods to solve these design equations. The method has been also used recently to design parallel manipulators by Kim and Tsai [39].

The complex number method was introduced around 1960 by Sandor [88] for planar synthesis and has been generalized in different ways. It is based on creating loop

equations for a given chain from a reference position of the end-effector to a desired position. Sandor used complex numbers to represent both the vectors and the planar operators and later applied the same procedure to spatial chains using quaternions, see [86], [44]. The same principle has been applied to spatial synthesis expressing the links as vectors and the rotations as matrices in [40], [87], and in [89] they were used to state the design equations for spatial 3R and 4R linkages. However, some difficulties arised in the use of the loop equation method for spatial synthesis. The loop equations describe the location of the links and these introduce a set of extra parameters; extra equations are needed to account for them and for the orientations. The loop equation method for spatial linkages was used mainly for path generation due to the fact that the loop equations characterize the translation of the mechanism, while they are parameterized by the joint variables. In [89], [90], these equations are called the *displacement equations* and a set of *orientation equations* is added to account for the rotations.

The attempt to generalize the spatial synthesis led to the methodology derived by Tsai [97] around 1970, which is based on using the geometry underlying screw kinematics. Geometric elements in kinematics were defined by Yang. Roth [81], using a concept originated by Halphen (1882), developed the screw triangle as a tool in kinematics and extended it to screw polygons for the study of n positions of a rigid body. Tsai applied the screw triangle to the screws representing the joints of a dyad and the resultant composition of their displacements. Tsai's dissertation provides

a long list of chains and associated design equations. This approach introduced intermediate joint parameters as necessary to describe the chains. In Tsai and Roth (1973) [99] they obtained an algebraic solution for the 10 design equations of the spatial RR chain. The equivalent screw triangle provides with a systematic method for the synthesis of spatial chains; however, its use is limited to dyads and three-link chains because of the complexity of the equations obtained.

With the development of robotics, kinematics equations, also called forward kinematics, started being used for solving the synthesis problem. Park and Bobrow [66] used them as product of exponentials to optimize the location of cooperating 6-dof robots. De Sa [19] and Ravani [77] studied the use of the kinematic mapping in planar synthesis. This approach was introduced by Mavroidis [52] for spatial constrained robots by considering, instead of the action of the group, the design equations created from equating directly the elements of the group. This is based on formulating the 4×4 homogeneous transformation that defines the kinematics equations of the robot, equating it to the goal transformation. Using this formulation, Mavroidis and Lee solved the spatial RR chain, RRR chain and PRR chain, see [52], [46], [47]. This methodology provides with a systematic way of creating the design equations for any robot. On the other hand, the design equations obtained with this approach are in general more complicated than those derived directly from geometric constraints. For a selected robot topology, the kinematics equations are parameterized by the structural dimensions of the robot (link lengths and twist angles and location of the

robot) and by the joint variables. In the dimensional synthesis, only the structural parameters matter, hence a representation of the workspace that eliminates the joint variables would reduce, in general, the complexity of the solution process. The algebraic representation of the workspace given by the geometric constraint polynomial equations leads to a more compact formulation, easier solution and more information about the geometry of the problem, but it lacks the systematic methodology for defining the design equations.

In our approach, we combine the systematic methodology of using the robot kinematics equations with the geometric meaning of the equivalent screw triangle. We use dual quaternions to state the kinematics equations of the chain as successive screw displacements, described by Gupta [26], see also Tsai [101], and equate them to the task positions in dual quaternion form. The dual vector of the dual quaternion is the screw of the displacement used in Tsai's methodology. The design equations contain the axes of the robot in a reference configuration, as in [86], parameterized by the joint variables. However, it is possible to eliminate these to obtain algebraic equations that, in the simplest cases, have been proved to coincide with those obtained using geometric methods. The dual quaternion synthesis methodology is intended to be applied to any robot with less than six degrees of freedom to obtain a set of equations that is reasonably affordable to solve.

1.5 Design of Robotic Systems

The design of a robot is the process of choosing of a mechanical system (links and joints) capable of realizing a task or series of tasks in space. The design process involves finding or choosing a solution to satisfy a set of requirements. Part of these requirements are related to the movement of the robot, while others may be related to factors such as load capacity, repeatability, power consumption or cost. In the kinematics design problem, the designer receives or specifies a certain task that a robot has to perform, namely a movement in space, and he or she has to create the mechanical system that better fits he given kinematics task and also agrees with a set of performance requirements.

One of the advantages of robots is their capability to perform different tasks. Traditionally, the robots are designed with enough degrees of freedom so that the desired task needs only to lie inside the three-dimensional reachable workspace of the robot. This is accomplished by adjusting the dimensions of the links to reach the farthest point; the control and path planning are used to solve the problem of reaching the task inside the reachable volume. However, such a general scheme is not needed in many cases, and it may be economically more efficient to design robots tailored to a specific set of tasks.

It is under this framework that we define the *constrained robots*. A constrained robotic system is constructed from one or more serial chains, each of which imposes a constraint on the end-effector. These so-called "constrained robotic systems" provide

structural support in certain directions while allowing freedom of movement in others.

It seems that there is the need for a tool for the design of robotic systems with a kinematic synthesis module capable of offering different designs for a given task or set of tasks. Many of the synthesis procedures give multiple solutions. Together with the synthesis equations, optimization methods can be used to fit the workspace of the robot to a broader task or to optimize for some performance specifications. Among the performance specifications we may want to optimize we include dynamic requirements, structural requirements (like joint variable limits), trajectory planning or obstacle avoidance problems. For work in design optimization see [105], [66], [59], [10], [17], [43], and also the following works [65], [51], [57], [1] about metrics and performance indices.

In this dissertation we present a method for the kinematic synthesis of robotic systems of less than six degrees of freedom that is systematic, adaptable to be used within a design process, and can be applied to any robot topology. It is our goal to develop the synthesis method so that it can be used as a design tool, and at the same time use it to characterize the dimension of the spatial synthesis problem. However, it is important to remark that the dimensional kinematic synthesis is only one of the steps of the whole design process that has to be considered.

1.6 Contributions

The research developed in this dissertation is intended to create the mathematical tools to generalize the finite-position synthesis of spatial robots, with special interest in its implementation in computer-aided-design software. Its contributions are listed below.

- A procedure for creating design equations for any constrained robot based on the expression of the workspace of the robot as dual quaternions. Using these design equations, a method has been created for computing the maximum number of exact positions that we can define for each robot topology. See [73].

- An identification of the effect that the structure of the group of displacements has on the synthesis process. It led to the identification of orientation-limited robots and translation-unconstrained robots, which differ in the definition of task positions. Formulas were found to identify these cases. See [73].

- A methodology to eliminate non-structural parameters from the equations, based on the use of linear algebra and spaces of solutions, that provides in addition with the inverse kinematics. We compared the degree of complexity of the initial equations and the ones obtained after the elimination. See [71], [75].

- The application of the above methodology led to the synthesis solution of several constrained robots that, to our knowledge, had not been completely solved before using exact synthesis: RPP, RPR, TPR, RPRP, RPC, PRRR, CRR,

RRRR and RCC. Some of these synthesis methods have been implemented in a computer software. These results can be found in [73], [75], [74].

- A simplification procedure for the design equations based on locating the fixed frame in the principal axes of the set of screw transformations that form the workspace of the robot. This procedure was applied to the spatial RR robot to obtain a simplified solution. See [70], [72].

1.7 Organization of the Dissertation

The material of this dissertation has been organized in the following way. Chapter 2 contains the mathematical background on group theory, representations and algebras related to the spatial displacements. Chapter 3 develops the basic theory of the dual quaternion synthesis of constrained robots and relates this theory to previous synthesis methodologies. Chapter 4 presents an application of the use of quaternions for the synthesis of planar and spherical robots, and identifies special cases that correspond to subgroups of the group of spatial displacements. Chapter 5 contains a systematic enumeration of all the cases solved using the dual quaternion methodology, with the specific elimination process for each case and with a numerical example. Chapter 6 presents the solution for the spatial RR robot in which the equations are simplified using the geometry of the workspace. Finally, Chapter 7 contains the conclusions and future research.

Chapter 2

Kinematics Background

In this chapter we introduce the mathematical tools used in the kinematic synthesis of spatial linkages. Displacements in space are identified as having a structure of group with certain properties. Different representations of this group are advantageous for different purposes; the addition of an inner product allows us to complement the group to obtain a structure of algebra. This is used to describe the relation between the displacement and the space where it acts.

The first sections contain the group theory and algebra theory necessary for further development of the synthesis problem. Later we define kinematic structures and elements of the kinematics of robots. For basic definitions and proofs, the reader is referred to [4], [91] and [53].

2.1 The Group of Displacements

The Euclidean space is a set of points that can be described by three coordinates. The movement of the points can be viewed as the action of a group on this set. To study the movement in Euclidean space is to study the properties of this group.

The displacements of rigid bodies correspond to isometries of the three-dimensional Euclidean space, that is, displacements that preserve the Euclidean distance between points. The set of isometries of the Euclidean space E_3 with the composition of functions form a non-commutative group called the Euclidean group $E(3)$.

Motion corresponds to a subgroup called the special Euclidean group $SE(3)$ which is characterized by keeping the angles between points. The Euclidean group has structure of Lie group, that is, it is a continuous group whose elements form a differentiable manifold. We are not going to focus on the instantaneous properties of $SE(3)$ but still need to be able to define arbitrary displacements as close as needed.

2.1.1 Properties of the group of displacements

The subgroups of the Euclidean group play an important role in its structure. On Table 2.1 we find the continuous subgroups that are of interest to us.

The main subgroups that we deal with are $SO(3)$ and T, the subgroup of spatial rotations and the subgroup of spatial translations.

$SO(3)$, the special orthogonal group, is a subgroup of the linear group, formed by the elements of the orthogonal group $O(3)$ with determinant equal to one.

Table 2.1: Subgroups of the group of spatial displacements

Subgroup	*Description*	*Dimension*
$SO(3)$	Rotations	3
T	Translations	3
R_u	Rotations about an axis u	1
T_u	Translations along a direction u	1
$H_{u,p}$	Screw displacement along axis u of pitch p	1
C_u	Rotation and translation (cylindrical movement) along axis u	2
T_P	Planar translation on plane P	2
$SE(2)_P$	Planar displacements on plane P	3
S_o	Spherical rotations about a point o	3
$Y_{u,p}$	Translating screw of axis u and pitch p	3
X_u	Translation followed by rotation about axis u	4

Any spatial motion can be seen as the composition of an element of $SO(3)$ and an element of T; both subgroups are disjoint and their union generates the whole special Euclidean group. The translation group is normal, that is, for any rotation r and translation t, $rtr^{-1} \in T$, while the orthogonal group is not. With these conditions, the special euclidean group has a structure of semi-direct product, $SE(3) = SO(3) \rtimes \mathbb{R}^3$. This translates to the fact that we can operate rotations independently, while translations are modified by the rotations: we can write any element of $SE(3)$ as (r, t); then the group operation can be decomposed as

$$(r_1, t_1)(r_2, t_2) = (r_1 r_2, r_2 t_1 + t_2). \tag{2.1}$$

2.2 The Clifford Algebra of Dual Quaternions

Given an n-dimensional vector space, such as $\mathbb{R}^n$, the addition of an associative product that operates two vectors to obtain another vector of the space, creates an *algebra*. Examples of algebras are the traditional three-dimensional vector algebra with the cross product, or the matrix algebra of square matrices with the matrix product. A Clifford algebra is used in this work to represent the group of proper rigid motions, and will be defined in the following sections. For more information about Clifford algebras, see [53], [91], [15] or [24].

2.2.1 Exterior algebra

Exterior algebra was introduced by Grassmann to study linear varieties in a projective space. Let V be a vector space of dimension n, we can define a *wedge* operator, $\wedge$, with the properties:

- Associative: $(\mathbf{u} \wedge \mathbf{v}) \wedge \mathbf{w} = \mathbf{u} \wedge (\mathbf{v} \wedge \mathbf{w})$
- Distributive: $\mathbf{u} \wedge (a\mathbf{v} + b\mathbf{w}) = a\mathbf{u} \wedge \mathbf{w} + b\mathbf{u} \wedge \mathbf{w}$
- Skew-symmetry: $\mathbf{u} \wedge \mathbf{v} = -\mathbf{v} \wedge \mathbf{u}$

This creates an exterior algebra, or multi-vector algebra, denoted as $\Omega(V)$. If a basis of V is $\{\mathbf{e}_1, \ldots, \mathbf{e}_n\}$, the elements of the exterior algebra are linear combinations of multi-vectors of rank that goes from 0 to n, and correspond to all possible products of the basis vectors using the wedge product. Vectors of rank i are created as all the

nonzero combination in operating i basis vectors, $\mathbf{e}_1 \wedge \mathbf{e}_2 \wedge \ldots \wedge \mathbf{e}_i$ and so on; there are $\binom{n}{i}$ of them. The exterior algebra has dimension $2^n - 1$.

We can interpret the rank p vectors as defining p-dimensional parallelepipeds, such that rank 1 vectors define lines, rank 2 vectors define elements of area, and so on.

We can write the wedge product of j vectors as

$$\mathbf{v}_1 \wedge \ldots \wedge \mathbf{v}_j = \sum_{i_1 < \ldots < i_j} M^{i_1 \ldots i_j} \mathbf{e}_{i_1} \wedge \ldots \wedge \mathbf{e}_{i_j}, \tag{2.2}$$

where $Mi_1 \ldots i_j$ are $j \times j$ minors.

2.2.2 Clifford algebra

Given a vector space V, for instance $\mathbb{R}^n$, the geometric or Clifford algebra $\mathcal{C}(V)$ is obtained if we define a *geometric product* that combines the multi-vector structure of the wedge product and a scalar product -a symmetric bilinear form- defined on the vector space. A Clifford algebra becomes an exterior algebra if the scalar product is zero. The geometric product for two vectors can be defined as

$$\mathbf{pq} = - < \mathbf{p}, \mathbf{q} > + \mathbf{p} \wedge \mathbf{q} \tag{2.3}$$

and this gives in general a multi-vector of different rank. Notice that this definition cannot be applied directly to general multi-vectors, unless we define the scalar product more generally; however, using the linearity of the product, we just need to operate on the basis vectors. We can also use a definition of the geometric product based on its properties:

- Linear,
- Associative,
- There exists a scalar product for the vectors of $\mathbb{R}^n$: $< \mathbf{p}, \mathbf{q} >$ with the usual properties.
- We require geometric the product to satisfy the condition:

$$\mathbf{pq} + \mathbf{qp} = -2 < \mathbf{p}, \mathbf{q} > \tag{2.4}$$

If we apply this product to p vectors, we obtain a multivector of rank p plus a scalar. There are $\binom{n}{p}$ basis multivectors of rank p. A typical element of the Clifford algebra is a linear combination of the basis multivectors of all ranks up to n, inlcuding a scalar which is considered a multivector of rank zero. Thus, there are 2^n components in a Clifford algebra element. The dimension of the Clifford algebra as a vector space is 2^n.

The product of two elements of the Clifford algebra can be performed by expressing the elements as linear combinations of the elements of the basis and then distributing the product on the basis vectors.

The conjugate

Conjugation in the Clifford algebra is a bijective map that reverses the order of products, that is, $(\mathbf{pq})^* = \mathbf{q}^*\mathbf{p}^*$. To conjugate an element of the algebra, we just need to conjugate the basis elements ($\mathbf{e}_i^* = -\mathbf{e}_i$ for all basis elements) and apply

linearity; the conjugation does not affect the scalars.

The inverse

We cannot invert every element of a Clifford algebra, as it has zero divisors. The elements that are not zero divisors have an inverse and form a group. For the underlying vector space, because $\mathbf{pp} = - < \mathbf{p}, \mathbf{p} >$, the inverse is $\mathbf{p}^{-1} = -\mathbf{p}/ < \mathbf{p}, \mathbf{p} >$.

For those elements of the algebra of "length" equal to one, defined as $PP^* = 1$, the inverse of the element P is its conjugate.

2.2.3 Clifford subalgebras

The subset of the Clifford algebra of V spanned by multivectors of even rank will have even rank under the geometric product, and hence it spans a subalgebra which is the even Clifford algebra, $\mathcal{C}^+(V)$, of dimension 2^{n-1}.

Notice that there exists always an isomorphism between the Clifford algebra of V^n (vector space of dimension n) and the even subalgebra of V^{n+1}, with the standard scalar product.

2.2.4 The even Clifford subalgebra of $\mathbb{R}^3$: Quaternions

The even Clifford algebra of $\mathbb{R}^3$ with the usual Euclidean dot product, gives the following relations for the basis vector products: let $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ be the basis vectors, the product of two of them gives three rank-2 multivectors, and the product of all three of them gives one rank-3 multivector, but this does not belong to the even subalgebra.

The general element of the algebra can be written as

$$H = h_x \mathbf{e}_2\mathbf{e}_3 + h_y \mathbf{e}_3\mathbf{e}_1 + h_z \mathbf{e}_1\mathbf{e}_2 + h_w \tag{2.5}$$

If we denote $i = \mathbf{e}_2\mathbf{e}_3$, $j = \mathbf{e}_3\mathbf{e}_1$, $k = \mathbf{e}_1\mathbf{e}_2$, we obtain the identities first obtained by Hamilton [28] to define quaternions:

$$\begin{aligned} & i^2 = j^2 = k^2 = -1 \\ & ij = -ji,\ ik = -ki,\ jk = -kj \\ & ij = k,\ jk = i,\ ki = j \end{aligned} \tag{2.6}$$

We will denote the element of the Clifford algebra as

$$H = \mathbf{H} + h_w = h_x i + h_y j + h_z k + h_w \tag{2.7}$$

and using this vector notation we can define the conjugate quaternion, $H^* = -\mathbf{H} + h_w$.

The product of two elements of this algebra can be computed applying linearity to the geometric product of the elements of the basis; however, the vector notation allows for this simpler computation,

$$GH = (g_w\mathbf{H} + h_w\mathbf{G} + \mathbf{G} \times \mathbf{H}) + (h_w g_w - \mathbf{H} \cdot \mathbf{G}) \tag{2.8}$$

The even Clifford algebra $\mathcal{C}^+(\mathbb{R}^3)$ is isomporphic to the Clifford algebra $\mathcal{C}(\mathbb{R}^2)$, the *quaternions*. The group of quaternions is a double covering of the group $SO(3)$.

A rotation in $\mathbb{E}^3$ of angle ϕ about a normalized rotation axis $\mathbf{s}$ is written in quaternion form as

$$H = \mathbf{s} \sin\frac{\phi}{2} + \cos\frac{\phi}{2}. \tag{2.9}$$

The action of this rotation on vectors of $\mathbb{E}^3$, considering the vector quaternion (with fourth component equal to zero) $\mathbf{x}$, gives the rotated vector quaternion $\mathbf{X} = Q\mathbf{x}Q^*$. Because the inverse of a rotation is a rotation of angle $-\phi$ about the same axis, and $-\sin\frac{\phi}{2} = \sin(-\frac{\phi}{2})$ and $\cos\frac{\phi}{2} = \cos(-\frac{\phi}{2})$, the conjugate quaternion Q^* represents the inverse rotation.

The quaternion corresponding to a general rotation $[R]$ is

$$H = \begin{Bmatrix} h_x \\ h_y \\ h_z \\ h_w \end{Bmatrix} = \begin{Bmatrix} \sin\frac{\theta}{2}s_x \\ \sin\frac{\theta}{2}s_y \\ \sin\frac{\theta}{2}s_z \\ \cos\frac{\theta}{2} \end{Bmatrix}, \tag{2.10}$$

where $\mathbf{s}$ is the axis and θ is the angle of the rotation.

Because $h_x^2 +_y^2 +h_z^2 + h_w^2 = 1$, quaternions can be viewed as points on the unit hypersphere of $\mathbb{R}^3$.

2.2.5 The even Clifford subalgebra of $\mathbb{P}^2$: Planar Quaternions

Let $\mathbb{P}^2$ be the projective space obtained by embedding the Euclidean space $\mathbb{E}^2$ as the plane $z = 1$ in the three-dimensional space $\mathbb{R}^3$ of coordinates (x, y, z). This is the vector space in which the homogeneous transform operates,

$$\begin{Bmatrix} X \\ Y \\ 1 \end{Bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & d_x \\ \sin\theta & \cos\theta & d_y \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ y \\ 1 \end{Bmatrix}. \tag{2.11}$$

We introduce a degenerate scalar product in $\mathbb{P}^2$,

$$< \mathbf{p}, \mathbf{q} >= \mathbf{p}^T \begin{bmatrix} 1 & & \\ & 1 & \\ & & 0 \end{bmatrix} \mathbf{q} \tag{2.12}$$

which makes distances and angles measured parallel to the plane $z = 1$.

The geometric product defined with this degenerate scalar product will operate on the basis vectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ as follows:

$$\mathbf{e}_1\mathbf{e}_1 = -1, \quad \mathbf{e}_2\mathbf{e}_2 = -1, \quad \mathbf{e}_3\mathbf{e}_3 = 0. \tag{2.13}$$

The general element of $\mathcal{C}^+(\mathbb{P}^2)$ is

$$P = p_x\mathbf{e}_2\mathbf{e}_3 + p_y\mathbf{e}_3\mathbf{e}_1 + p_z\mathbf{e}_1\mathbf{e}_2 + p_z = p_x i\epsilon + p_y j\epsilon + p_z k + p_w, \tag{2.14}$$

if we denote $\mathbf{e}_2 = i$, $\mathbf{e}_1 = j$, $\mathbf{e}_1\mathbf{e}_2 = k$ and $\mathbf{e}_3 = \epsilon$. The i, j, k are the quaternion units of $\mathcal{C}(\mathbb{R}^2)$, and ϵ is the dual unit with the property $\epsilon^2 = 0$.

The planar quaternion corresponding to the general planar displacement $[T]$ in Eq.(2.11) is, in vector form,

$$P = \begin{Bmatrix} p_x \\ p_y \\ p_z \\ p_w \end{Bmatrix} = \begin{Bmatrix} \frac{d_x}{2}\cos\frac{\theta}{2} + \frac{d_y}{2}\sin\frac{\theta}{2} \\ \frac{d_y}{2}\cos\frac{\theta}{2} - \frac{d_x}{2}\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} \\ \cos\frac{\theta}{2} \end{Bmatrix} \tag{2.15}$$

Notice that the elements of the planar quaternion are constrained by the condition $p_z^2 + p_w^2 = 1$. If we consider the components of the planar quaternion as homogeneous coordinates in $\mathbb{P}^4$, then both solutions from this condition give the same projective point, and we can consider that each point of $\mathbb{P}^4$ defines a planar displacement.

2.2.6 The even Clifford subalgebra of $\mathbb{P}^3$: Dual Quaternions

Let $\mathbb{P}^3$ be the projective space obtained by embedding the Euclidean space $\mathbb{E}^3$ as the plane $w = 1$ in the four-dimensional space $\mathbb{R}^4$ of coordinates (x, y, z, w). This is the vector space in which the homogeneous transform operates,

$$\begin{Bmatrix} X \\ Y \\ Z \\ 1 \end{Bmatrix} = \begin{bmatrix} & [R] & & \mathbf{d} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \\ 1 \end{Bmatrix}. \tag{2.16}$$

If we define the degenerate scalar product,

$$< \mathbf{p}, \mathbf{q} >= \mathbf{p}^T \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 0 \end{bmatrix} \mathbf{q} \tag{2.17}$$

we obtain that $\mathbf{e}_1\mathbf{e}_1 = -1, \quad \mathbf{e}_2\mathbf{e}_2 = -1, \quad \mathbf{e}_3\mathbf{e}_3 = -1, \quad \mathbf{e}_4\mathbf{e}_4 = 0$, and the general element of the Clifford algebra,

$$\hat{Q} = q_1\mathbf{e}_2\mathbf{e}_3 + q_2\mathbf{e}_3\mathbf{e}_1 + q_3\mathbf{e}_1\mathbf{e}_2 + q_4 + q_5\mathbf{e}_4\mathbf{e}_1 + q_6\mathbf{e}_4\mathbf{e}_2 + q_7\mathbf{e}_4\mathbf{e}_3 + q_8\mathbf{e}_1\mathbf{e}_2\mathbf{e}_3\mathbf{e}_4 \tag{2.18}$$

is a linear combination of rank 4, 2, and 0 multivectors. This algebra has dimension $2^{4-1} = 8$.

We define $\mathbf{e}_1\mathbf{e}_2\mathbf{e}_3\mathbf{e}_4 = \epsilon$, and see that $\epsilon^2 = 0$, which identifies the dual number unit. If we call, as before, $i = \mathbf{e}_2\mathbf{e}_3$, $j = \mathbf{e}_3\mathbf{e}_1$, $k = \mathbf{e}_1\mathbf{e}_2$, then $\epsilon i = \mathbf{e}_4\mathbf{e}_1$, $\epsilon j = \mathbf{e}_4\mathbf{e}_2$, $\epsilon k = \mathbf{e}_4\mathbf{e}_3$. Because the generators commute in the even algebra, dual quaternions can be seen as a tensor product of the algebra of quaternions and the ring of dual

numbers, see [91]. We rewrite the general element as

$$\hat{Q} = (q_1 + q_5\epsilon)i + (q_2 + q_6\epsilon)j + (q_3 + q_7\epsilon)k + (q_4 + q_8\epsilon) = Q + \epsilon Q^0 \tag{2.19}$$

where Q and Q^0 are quaternions. This is our standard expression of a *dual quaternion*. We call Q the *real part* and Q^0 the *dual part*.

We express a general displacement $[T]$ as a dual quaternion

$$\hat{Q} = \begin{Bmatrix} q_x \\ q_y \\ q_z \\ q_w \end{Bmatrix} + \epsilon \begin{Bmatrix} q_x^0 \\ q_y^0 \\ q_z^0 \\ q_w^0 \end{Bmatrix} = \begin{Bmatrix} \sin\frac{\theta}{2}s_x \\ \sin\frac{\theta}{2}s_y \\ \sin\frac{\theta}{2}s_z \\ \cos\frac{\theta}{2} \end{Bmatrix} + \epsilon \begin{Bmatrix} \sin\frac{\theta}{2}s_x^0 + \frac{d}{2}\cos\frac{\theta}{2}s_x \\ \sin\frac{\theta}{2}s_y^0 + \frac{d}{2}\cos\frac{\theta}{2}s_y \\ \sin\frac{\theta}{2}s_z^0 + \frac{d}{2}\cos\frac{\theta}{2}s_z \\ -\frac{d}{2}\sin\frac{\theta}{2} \end{Bmatrix}. \tag{2.20}$$

The dual quaternion can be also expressed as

$$\hat{Q} = \cos\frac{\hat{\theta}}{2} + \sin\frac{\hat{\theta}}{2}\mathsf{S}, \tag{2.21}$$

with $\sin\frac{\hat{\theta}}{2} = \sin\frac{\theta}{2} + \epsilon\cos\frac{\theta}{2}$, $\cos\frac{\hat{\theta}}{2} = \cos\frac{\theta}{2} - \epsilon\sin\frac{\theta}{2}$, and the dual vector $\mathsf{S} = \mathbf{s} + \epsilon\mathbf{s}^0$ being the screw axis of the displacement.

To represent a displacement, the elements of the dual quaternion are constrained by the conditions

$$\begin{aligned} QQ^* = 1: &\quad q_x^2 + q_y^2 + q_z^2 + q_w^2 = 1, \\ Q^0Q^* + QQ^{0*} = 0: &\quad q_xq_x^0 + q_yq_y^0 + q_zq_z^0 + q_wq_w^0 = 0. \end{aligned} \tag{2.22}$$

The dual quaternions are elements of the semi-direct product group $Spin(3) \rtimes \mathbb{R}^3$, that is a double covering of SE(3); this group is isomorphic to $Spin(3) \rtimes \mathbb{R}^3/\mathbb{Z}_2$, which identifies quaternions $\hat{Q}$ and $-\hat{Q}$.

If we consider the components of the dual quaternion as homogeneous coordinates in $\mathbb{P}^7$, then $\hat{Q} = -\hat{Q}$, and we only need the second relation in Eq.(2.22). That condition defines a quadratic surface in the projective space, called the *Study quadric*. Points on the quadric represent spatial displacements, except those belonging to the 3-dimensional plane defined by $Q^0 Q^{0*} = 0$.

Relation between matrix and quaternion representation

The relation between the quaternion and the rotation matrix representation is captured by the expression of the rotation matrix as Euler parameters, which, when normalized, are the components of the quaternion, see [9]. For the dual part of the dual quaternion, we use the vector quaternion (that is, with no scalar component) $D = \mathbf{d}$, $\mathbf{d}$ being the translational component of the transformation, and compute

$$Q^0 = \frac{1}{2} DQ \tag{2.23}$$

The displacement of a point by the action of a matrix, $\mathbf{X} = [T]\mathbf{x}$, is computed using dual quaternions as

$$\hat{X} = \hat{Q}\hat{x}\hat{Q}^* = \mathbf{Q}\mathbf{x}\mathbf{Q}^* + \epsilon(\mathbf{Q}\mathbf{x}^0\mathbf{Q}^* + \mathbf{Q}^0\mathbf{x}\mathbf{Q}^* + \mathbf{Q}\mathbf{x}\mathbf{Q}^{0*}). \tag{2.24}$$

Here, $\hat{X}$ is the vector dual quaternion for the vector $\mathbf{X}$ in the fixed reference frame F and $\hat{x}$ is the vector dual quaternion for the vector in the moving reference frame, $\mathbf{x}$.

2.3 Representations of SE(3)

2.3.1 Homogeneous transformations

Four-dimensional matrices, called $4x4$ homogeneous transforms, are commonly used to describe the coordinate transformation from a moving frame M located at the end-effector of the robot to a fixed frame F located at the base. If $\mathbf{X}$ are the homogeneous coordinates of a point in the fixed frame and $\mathbf{x}$ are the homogeneous coordinates in the moving frame, we have the relation $\mathbf{X} = [T]\mathbf{x}$, where

$$\mathbf{X} = \begin{bmatrix} [R] & \mathbf{d} \\ 0 & 1 \end{bmatrix} \mathbf{x}. \tag{2.25}$$

We denote by $[R] \in SO(3)$ the rotation matrix, and by $\mathbf{d} \in \mathbb{R}^3$ the three-dimensional translation vector.

The composition of displacements corresponds to the matrix multiplication, which operates on the homogeneous transform as indicated in Eq.(2.1). For robotics applications, a displacement may be considered as the composition of screw transformations about coordinate axes. The most commonly used are transformations about the Z and the X axes.

The screw displacement along the joint axis Z, or $\mathbf{X} = [Z(\theta, d)]\mathbf{x}.$, is defined by

$$\begin{Bmatrix} X \\ Y \\ Z \\ 1 \end{Bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \\ 1 \end{Bmatrix}. \tag{2.26}$$

The screw displacement along the X-axis by the amounts a and α is $\mathbf{X} = [X(\alpha, a)]\mathbf{x}$,

$$\begin{Bmatrix} X \\ Y \\ Z \\ 1 \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \\ 1 \end{Bmatrix}. \tag{2.27}$$

Chalses' theorem [37] states that a general displacement can be written as a screw motion consisting of a rotation about an axis and a translation along the same axis. We can find the screw axis $\mathsf{S} = \mathbf{s} + \epsilon(\mathbf{c} \times \mathbf{s})$, angle ϕ and translation t from the expression of Eq.(2.25).

We use Cayley's formula and its inverse to pass from the rotation matrix to the rotation axis. The matrix $[B]$ in $[B] = [R - I][R + I]^{-1}$ is skew-symmetric and contains the components of the Rodrigues' vector $\mathbf{b}$ that defines the rotation axis $\mathbf{s}$ as $\mathbf{s} = \tan\frac{\phi}{2}\mathbf{b}$. We find the point on the axis and the translation following [58],

$$\mathbf{c} = \frac{\mathbf{b} \times (\mathbf{d} - \mathbf{b} \times \mathbf{d})}{2\mathbf{b} \cdot \mathbf{b}}, \quad \mathbf{d} = [I - R]\mathbf{c} + t\mathbf{s}. \tag{2.28}$$

2.3.2 Relative transformations

We can also describe spatial displacements as relative transformation matrices if we choose a reference position $[T_0]$ and perform a right translation. Let $[T_0]$ and $[T_1]$ be two transformations from a moving frame M to a fixed frame F. The relative displacement from $[T_0]$ to $[T_1]$ measured in F is $[T_{01}] = [T_1][T_0]^{-1}$, or

$$[T_{01}] = [R_{01}, \mathbf{d}_{01}] = [R_1 R_0^T, \mathbf{d}_1 - R_1 R_0^T \mathbf{d}_0]. \tag{2.29}$$

The relative displacement can be expressed as the relative screw displacement,

$$[T_{01}] = [T(\hat{\phi}, \mathsf{S})], \tag{2.30}$$

where S is the relative screw axis and $\hat{\phi}$ is the relative rotation and translation along the axis. Notice that the coordinates of the relative screw axis are expressed in the fixed frame.

2.3.3 Dual quaternions

We can create a dual quaternion representation for the relative displacement of Eq.(2.30) directly from the screw axis S and dual angle $\hat{\phi}$,

$$\hat{S}(\hat{\phi}) = \cos(\hat{\phi}/2) + \sin(\hat{\phi}/2)\mathsf{S} = \begin{Bmatrix} \sin(\hat{\phi}/2)\hat{s}_x \\ \sin(\hat{\phi}/2)\hat{s}_y \\ \sin(\hat{\phi}/2)\hat{s}_z \\ \cos(\hat{\phi}/2) \end{Bmatrix}, \tag{2.31}$$

where $\cos(\frac{\hat{\phi}}{2}) = \cos\frac{\phi}{2} + \epsilon(-\frac{d}{2}\sin\frac{\phi}{2})$ and $\sin(\frac{\hat{\phi}}{2}) = \sin\frac{\phi}{2} + \epsilon(\frac{d}{2}\cos\frac{\phi}{2})$.

Notice that this result must be equal to the one obtained from the quaternion product of the initial displacements in dual quaternion form, that is, $\hat{S}(\hat{\phi}) = \hat{S}_1(\hat{\phi}_1)\hat{S}_0(\hat{\phi}_0)^*$,

$$\begin{aligned} \hat{S}_1(\hat{\phi}_1)\hat{S}_0(\hat{\phi}_0)^* = &\cos\frac{\hat{\phi}_1}{2}\cos\frac{\hat{\phi}_0}{2} + \sin\frac{\hat{\phi}_1}{2}\sin\frac{\hat{\phi}_0}{2}\mathsf{S}_1 \cdot \mathsf{S}_0 - \\ &\cos\frac{\hat{\phi}_1}{2}\sin\frac{\hat{\phi}_0}{2}\mathsf{S}_0 + \sin\frac{\hat{\phi}_1}{2}\cos\frac{\hat{\phi}_0}{2}\mathsf{S}_1 - \sin\frac{\hat{\phi}_1}{2}\sin\frac{\hat{\phi}_0}{2}\mathsf{S}_1 \times \mathsf{S}_0. \end{aligned} \tag{2.32}$$

In general, the relative angle ϕ is not equal to the difference of the angles ϕ_0, ϕ_1 of the initial displacements. For the particular case when the screw axes of the displace-

ment coincide, $\mathsf{S}_0 = \mathsf{S}_1 = \mathsf{S}$, we obtain from the quaternion product in Eq.(2.32),

$$\hat{S}(\hat{\phi}) = \cos\frac{\phi_1 - \phi_0}{2} + \sin\frac{\phi_1 - \phi_0}{2}\mathsf{S} + \epsilon(-\frac{d_1 - d_0}{2}\sin\frac{\phi_1 - \phi_0}{2} + \sin\frac{\phi_1 - \phi_0}{2}\mathsf{S}^0 + \frac{d_1 - d_0}{2}\cos\frac{\phi_1 - \phi_0}{2}\mathsf{S}), \quad (2.33)$$

in which case the relative angle and translation can be computed from the initial ones as $\phi = \phi_1 - \phi_0$, and $d = d_1 - d_0$.

2.3.4 The Lie exponential

We can create a Lie algebra on a continuous group as the vector space of tangent vectors at the identity with a vector product called commutator or Lie bracket, see [63], [67] or [91]. The Lie exponential is a mapping from the Lie algebra to the group that allows us to express an element of the group as a certain function of the elements of its Lie algebra. For the group $SE(3)$ the Lie exponential seeks to express displacements as a function of velocities.

Consider the line $\mathsf{S} = \mathbf{s} + \epsilon\mathbf{c} \times \mathbf{s}$ in F, which is the screw axis of an instantaneous movement of a rigid body. If $\mathbf{w} = w\mathbf{s}$ is the angular velocity of this body around S and $\mathbf{v} = v\mathbf{s} = hw\mathbf{s}$ is its linear velocity along S, then the element of the Lie algebra $se(3)$ representing this movement is given by the 4×4 matrix $w[S]$, with

$$[S] = \begin{bmatrix} & [\mathbf{s}] & & \mathbf{c} \times \mathbf{s} + h\mathbf{s} \\ 0 & 0 & 0 & 0 \end{bmatrix}. \quad (2.34)$$

The matrix $[\mathbf{s}]$ is viewed as an operator that forms the 3×3 cross product matrix, that is $[\mathbf{s}]\mathbf{y} = \mathbf{s} \times \mathbf{y}$ for any vector $\mathbf{y}$. The 4×4 matrix transformation associated

with $[S]$ is obtained as the matrix exponential,

$$[T] = e^{[S]\phi}, \tag{2.35}$$

which, for a screw axis fixed in F, has the angle $\phi = wt$ and slide $d = vt$ around and along S.

For the relative displacement in Eq.(2.30), take the Lie algebra element with axis coincident with the screw axis of the relative displacement. Then, with $w = (\phi_1 - \phi_0)/\Delta t = \phi/\Delta t$ and $v = (d_1 - d_0)/\Delta t$ the velocity along the axis, the matrix exponential is

$$[T] = e^{[S]\phi} = \begin{bmatrix} & e^{t[\Omega]} & & \phi\mathbf{u} + (1 - \cos\phi)[\Omega]\mathbf{u} + (\phi - \sin\phi)[\Omega]^2\mathbf{u} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.36}$$

where we denote Ω the skew-symmetric matrix of the angular velociy, $[\Omega] = w[\mathbf{s}]$. The rotation matrix expression $[R] = e^{t[\Omega]}$ and the expression of the displacement vector $\mathbf{d}$ as a function of the velocity vector $\mathbf{u}$ can be found for instance in [91].

2.4 Kinematics of Robots

2.4.1 Basic joints

Releaux [79] identified six basic types of joints that share a surface of contact, that he called *lower pairs*. His classification is still used today in most robotics and mechanisms books. He defined revolute (R), cylindric (C), prismatic (P), spherical (S), helical (H) and plane (E) joints. In our research, we only consider the basic types of revolute and prismatic joints; any of the others can be obtained by combining the

two basic types with certain additional constraints. In addition to these, we refer to the universal joint (T), which can be modeled as two revolute joints at right angles. See Figure 2.1.

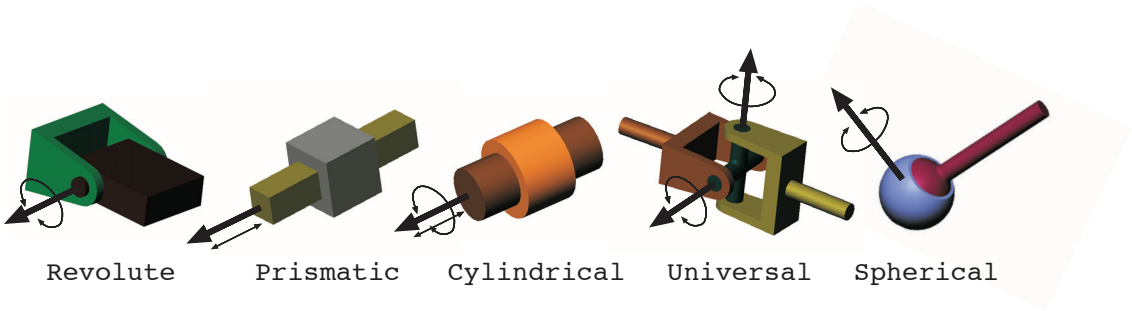

Figure 2.1: The different types of joints used in this research.

The revolute joint is a one-degree-of-freedom joint that allows a rotation of angle θ about the joint axis $\mathsf{G} = \mathbf{g} + \epsilon\mathbf{g}^0$. It can be expressed in dual quaternion form as

$$\hat{G}(\theta) = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}\mathsf{G} = \begin{Bmatrix} \sin\frac{\theta}{2}g_x \\ \sin\frac{\theta}{2}g_y \\ \sin\frac{\theta}{2}g_z \\ \cos\frac{\theta}{2} \end{Bmatrix} + \epsilon \begin{Bmatrix} \sin\frac{\theta}{2}g_x^0 \\ \sin\frac{\theta}{2}g_y^0 \\ \sin\frac{\theta}{2}g_z^0 \\ 0 \end{Bmatrix} \tag{2.37}$$

where we set the scalar as the fourth component of the dual quaternion.

The prismatic joint is also a one-degree-of-freedom joint, which allows translation along the direction $\mathbf{h}$ of the axis H. It is interesting to notice that the location of the prismatic axis does not matter for the movement of the prismatic joint. This is also

captured by its dual quaternion expression,

$$\hat{H}(d) = 1 + \epsilon\frac{d}{2}\mathsf{H} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{Bmatrix} + \epsilon \begin{Bmatrix} \frac{d}{2}h_x \\ \frac{d}{2}h_y \\ \frac{d}{2}h_z \\ 0 \end{Bmatrix} \tag{2.38}$$

The cylindric joint is a general screw motion, in which the rotation about and the translation along the screw axis S are nonzero and independent. It can be constructed as the composition of a revolute joint and a prismatic joint with same direction,

$$\hat{S}(\theta) = \cos\frac{\hat{\theta}}{2} + \sin\frac{\hat{\theta}}{2}\mathsf{S} = \begin{Bmatrix} \sin\frac{\theta}{2}s_x \\ \sin\frac{\theta}{2}s_y \\ \sin\frac{\theta}{2}s_z \\ \cos\frac{\theta}{2} \end{Bmatrix} + \epsilon \begin{Bmatrix} \sin\frac{\theta}{2}s_x^0 + \frac{d}{2}\cos\frac{\theta}{2}s_x \\ \sin\frac{\theta}{2}s_y^0 + \frac{d}{2}\cos\frac{\theta}{2}s_y \\ \sin\frac{\theta}{2}s_z^0 + \frac{d}{2}\cos\frac{\theta}{2}s_z \\ -\frac{d}{2}\sin\frac{\theta}{2} \end{Bmatrix}. \tag{2.39}$$

In a similar fashion, the universal (T) joint and the spherical (S) joint can be constructed from individual rotations. We will not consider helical or plane joints in this work.

2.4.2 Constrained robotic systems

We define a *constrained robotic system* as a workpiece, or end-effector, supported by one or more serial chains such that each chain imposes at least one constraint on its movement; we assume that the joints are independently actuated and free to move within their range of motion. Constrained robots are, basically, serial or parallel robots with less than six degrees of freedom. They allow movement in certain directions while providing structural support in other directions. Unlike robots with six or more degrees of freedom, constrained robots may have positions within the

three-dimensional volume of their workspace that are not reachable by the robot. In this context, the exact finite position synthesis makes sense for constrained robotic systems.

For serial robots, the degrees of freedom coincide with the number of joints of the robot, as we can see applying Gruebler's formula [96]. Hence, constrained serial robots are those that have from 2 to 5 joints and, for synthesis purposes, at least two links. Table 2.2 lists the different constrained serial robots that can be created with revolute, prismatic, universal, cylindrical and spherical joints. We consider the permutations of joints within the same set of joints as part of the same class of constrained robot, except for particular cases that will be discussed in Chapter 4. Robots with less than six joints that are not capable of at least one general displacement are not considered in this research and have not been not included.

Table 2.2: Constrained serial robots

DOF	*Structure*
2	RP, RR
3	RPP, PRP, RRP, RRR, – CP, RC, TP, TR
4	RPPP, RRPP, RPRP, RRRP, RRRR, – CPP, RCP, CC, TPP, PTP, RRC, TRP, TC, SP, TRR, TT, SR
5	RPPPP, RRPPP, RRRPP, RPRPR, RRRRP, RRRRR, – CPPP, CRPP, CCP, TPPP, CPRR, CRRP, CCR, CRC, TRPP, TPRP, TCP, PTC, SPP, PSP, CRRR , TRRP, TTP, TCR, SRP, SPR, SC, TRRR, TTR, SRR, ST

2.5 Summary

In this chapter, we reviewed the properties of the group of spatial displacements and presented the different representations that we use; we showed the relation between the homogeneous transformations, the matrix exponential and the dual quaternions for representing the workspace of a serial chain. We summarized the properties and structure of the Clifford algebra of dual quaternions that we use to formulate the design problem.

In the design problem, a spatial transformation is performed by a robot as the composition of transformations along the joints. In this chapter we also reviewed the basic types of joints and their dual quaternion expression, and defined and characterized the *constrained robots* that are the object of this research.

Chapter 3

Dual Quaternion Synthesis Theory

3.1 Introduction

In this chapter we present the methodology for the synthesis of constrained robots using dual quaternions.

The dual quaternion synthesis methodology uses Mavroidis [52] systematic method to create the design equations based on the kinematics equations of the robot; however, we use successive screw displacements described by Gupta [26] formulated using dual quaternion algebra [101]. Our approach uses the Plücker coordinates of the joint axes as design parameters and yields six independent design equations from each task position, see [71]. Like in the methodologies of Tsai and Roth [97], Sandor [86] and Mavroidis and Lee [52], the joint angles appear in our design equations. However, it is possible to eliminate them to solve directly for the axes of the joints. The equations obtained after this elimination are closer to the geometric constraints imposed by the chain [83]. Dual quaternions present separately the rotations and the conjugation of

translations and rotations that occurs when operating two general transformations. This gives some insight to the structure of the problem and allows us to classify constrained robots according to the movement that they can perform.

The chapter begins with the introduction to the methodology of creating the kinematics equations of a robot in matrix formulation, and how to transform them into dual quaternions. Then we create the design equations and relate the robot topology with the maximum number of task positions that can be defined. They form the parameterized design equations, that is, those design equations containing both design parameters and joint angle parameters. We then develop a procedure to create the reduced design equations free of joint variables. The algebraic and numerical methods that we use for solving the parameterized and the reduced equations are discussed briefly, and we present a table summarizing the constrained robots that have been solved using the dual quaternion approach. At the end of the chapter, we show how the dual quaternion approach can be identified with the complex number formulation of Sandor [88] if we specialize the dual quaternions to planar quaternions. We also show how to obtain the equations defining the geometry of Tsai's screw triangle [97] using the dual quaternion formulation.

3.2 Dual Quaternion Kinematics Equations of a Serial Robot

The kinematics equations of the robot equate the 4×4 homogeneous transformation $[D]$ between the end-effector and the base frame to the sequence of local coordinate transformations along the m joint axes of the chain,

$$[D] = [G][Z(\theta_1, d_1)][X(\alpha_{12}, a_{12})][Z(\theta_2, d_2)] \dots [X(\alpha_{m-1,m}, a_{m-1,m})][Z(\theta_m, d_m)][H]. \tag{3.1}$$

A serial robot can be kinematically defined as a series of joint axes denoted S_i, $i = 1, \dots, m$. The common normal, A_{ij}, to each pair of lines defines the link connecting the joints, see [13]. The distance a_{ij} and angle α_{ij} along the common normal line define the dimensions of the link. The parameters (θ, d) that define the movement at each joint and the link and twist (α, a) are collectively known as the Denavit-Hartenberg parameters. The transformation $[G]$ defines the position of the base of the chain relative to the fixed frame, and $[H]$ locates the tool relative to the last link frame, see Figure 3.1. Notice that for constrained serial chains it is necessary that $m \leq 5$.

The kinematics equations, written as homogeneous matrices, give a matrix representation of the workspace parameterized by the joint variables, (θ_i, d_i), and the link dimensions, (α_{ij}, a_{ij}).

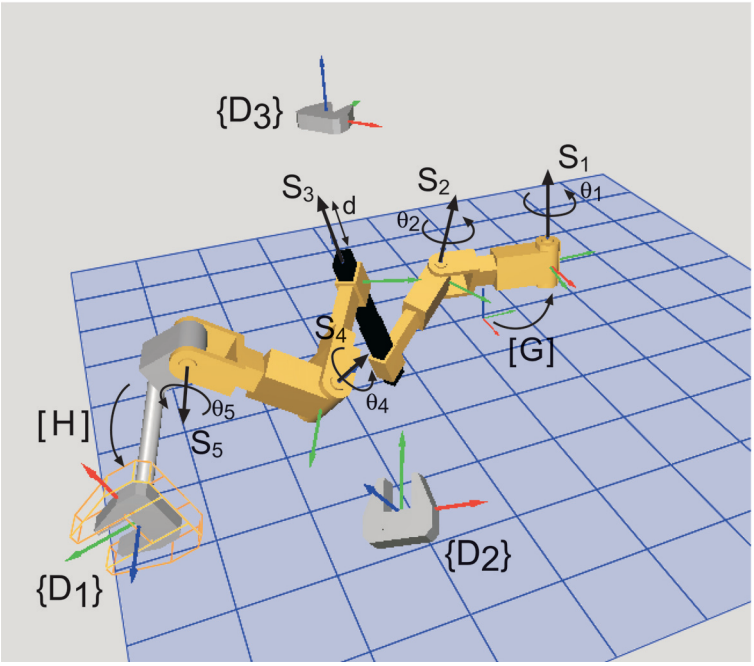

Figure 3.1: A constrained serial robot and three specified task positions.

3.2.1 Successive screw displacements

These kinematics equations can be transformed into successive screw displacements by choosing a reference position $[D_0]$. Let $[D_i]$ be the homogeneous matrix describing the transformation from the fixed frame to a moving frame $\{D_i\}$. We can compute $[D_{0i}] = [D_i][D_0]^{-1}$, that is,

$$\begin{aligned}[D_{0i}] =& [D_i][D_0]^{-1}, \\ =& ([G][Z(\theta_1^i, d_1^i)] \dots [Z(\theta_m^i, d_m^i)][H])([G][Z(\theta_1^0, d_1^0)] \dots [Z(\theta_m^0, d_m^0)][H])^{-1}. \end{aligned} \quad (3.2)$$

In order to simplify this equation we introduce the partial transformations $[M_j]$ up to but not including the j^{th} joint transformation for the reference configuration, so

that we have

$$\begin{aligned}
[M_1] =& [G], \\
[M_2] =& [G][Z(\theta_1^0, d_1^0)][X(\alpha_{12}, a_{12})], \\
& \dots \\
[M_j] =& [G][Z(\theta_1^0, d_1^0)][X(\alpha_{12}, a_{12})][Z(\theta_2^0, d_2^0)] \dots [X(\alpha_{j-1,j}, a_{j-1,j})].
\end{aligned} \tag{3.3}$$

Equation (3.2) can now be rewritten in the form

$$[D_{0i}] = [T(\Delta\theta_1, \mathsf{S}_1)][T(\Delta\theta_2, \mathsf{S}_2)] \dots [T(\Delta\theta_m, \mathsf{S}_m)], \tag{3.4}$$

where

$$\begin{aligned}
[T(\Delta\theta_1, \mathsf{S}_1)] =& [M_1][Z(\theta_1^i, d_1^i)][Z(\theta_1^0, d_1^0)]^{-1}[M_1]^{-1}, \\
[T(\Delta\theta_2, \mathsf{S}_2)] =& [M_2][Z(\theta_2^i, d_2^i)][Z(\theta_2^0, d_2^0)]^{-1}[M_2]^{-1}, \\
& \dots \\
[T(\Delta\theta_m, \mathsf{S}_m)] =& [M_m][Z(\theta_m^i, d_m^i)][Z(\theta_m^0, d_m^0)]^{-1}[M_m]^{-1},
\end{aligned} \tag{3.5}$$

and $\Delta\theta_j = \theta_j^i - \theta_j^0$ or $\Delta\theta_j = d_j^i - d_j^0$ depending on whether the joint is revolute or prismatic, respectively.

The displacements $[T(\Delta\theta_i, \mathsf{S}_i)]$ define the rotations about and translations along the joint axes S_i measured in the fixed frame relative to the reference configuration $[D_0]$. Notice that by expressing kinematics equations in this fashion, the base transformation $[G]$ is absorbed into the coordinates of the first joint axis and the tool transformation $[H]$ cancels.

3.2.2 Kinematics equations as product of exponentials

We can write the kinematics equations as product of the Lie exponentials, as described in Section 2.3.4, if we notice that the line S defining each joint axis is the screw axis of the instantaneous motion of each link for the displacement between the reference position and position i. The 4×4 matrix transformation associated with the element of the Lie algebra $[S]$ is obtained as the matrix exponential,

$$[T_{0i}] = e^{[S]\theta}[T_0], \tag{3.6}$$

and the kinematics equations become

$$[D_{0i}] = e^{[S_1]\theta_1} e^{[S_2]\theta_2} \dots e^{[S_m]\theta_m}, \tag{3.7}$$

where the reference configuration is such that $\theta_j = 0$ for all joint variables, and the matrix defining the transformation between the end-effector and the base frame when the robot is in the reference configuration has been made equal to the identity. This can always be achieved by considering one of the task transformations as the reference configuration and using relative displacements with respect to it to define the task.

3.2.3 Dual quaternion kinematics equations

The kinematics equations of the serial chain can be also computed using dual quaternions.

This formulation provides a compact representation of the displacement and a useful structure that assists for the elimination of the joint variables. The dual quater-

nion representation contains the rotation information in its real part, and the effect of rotations and translations on the translation information in its dual part.

A spatial displacement consisting of a rotation by θ and slide by d around and along a screw axis S is written as the dual quaternion,

$$\hat{S}(\hat{\theta}) = \sin(\frac{\hat{\theta}}{2})\mathsf{S} + \cos(\frac{\hat{\theta}}{2}), \tag{3.8}$$

see Eq.(2.21).

The dual quaternion of Eq.(3.8) encodes the same information as the 4×4 screw displacement matrix $[T(\Delta\theta, \mathsf{S})]$. We can define the dual quaternion kinematics equations for the serial chain by simply replacing the 4×4 matrices in Eq.(3.4) by their dual quaternion equivalents to obtain

$$\hat{D}^i = \hat{S}_1(\Delta\hat{\theta}_1^i)\hat{S}_2(\Delta\hat{\theta}_2^i)\ldots\hat{S}_m(\Delta\hat{\theta}_m^i). \tag{3.9}$$

For details about the computation of the dual quaternion components from the homogeneous matrix, see for instance [9] or [53].

The Plücker coordinates of the screw axes S_i, $i = 1, \ldots, m$, are defined in the base frame F and form convenient design parameters.

3.3 Design Equations

For a given kinematic chain, the design equations must impose that a set of specified task positions lie within the workspace of the robot. The algebraic shape of the workspace is determined by the structure, or topology, of the chain, but we can

adjust its physical dimensions by adjusting the dimensions of the links fo the robot. If we design for the maximum number of positions then there is no freedom left and the positions completely define the dimensions of the workspace and hence the dimensions of the robot; in this case we obtain a finite number of solutions.

3.3.1 The design equations

Let the n positions be defined by the 4×4 transforms $[P_i]$, $i = 1, \ldots, n$. Construct the $n-1$ relative transformation matrices $[P_{1i}] = [P_i][P_1]^{-1}$ and their associated dual quaternions $\hat{P}_i$, $i = 2, \ldots, n$. Equating the task dual quaternions $\hat{P}_i$ to the dual quaternion kinematics equations in Eq.(3.9), we impose that the spatial chain must be able to perform exactly those relative displacements. This yields the design equations

$$\mathcal{Q}_i : \quad \hat{S}_1(\Delta\hat{\theta}_1^i)\hat{S}_2(\Delta\hat{\theta}_2^i)\ldots\hat{S}_m(\Delta\hat{\theta}_m^i) - \hat{P}_i = 0, \quad i = 2, \ldots, n. \tag{3.10}$$

Each $\mathcal{Q}_i$ is an eight-dimensional vector equation. If not otherwise specified, we assume that each axis S_i defines either a revolute joint or a prismatic joint, but not both. This means in the dual angle $\Delta\hat{\theta}_i = \Delta\theta_i + \epsilon\Delta d_i$ only one of the values is the joint variable, either θ or d.

Using the quaternion product, the workspace of the most general serial chain,

consisting of five joints, is expressed as the dual quaternion $\hat{Q} = \hat{Q}^0 + \mathsf{Q}$, where

$$\begin{aligned}
\hat{Q}^0 = {} & c_1c_2c_3c_4c_5 - \\
& \sum_{\substack{\alpha_1<\alpha_2\in C_{2,5}\\ \alpha_3<\alpha_4<\alpha_5}} s_{\alpha_1}s_{\alpha_2}c_{\alpha_3}c_{\alpha_4}c_{\alpha_5}\mathsf{S}_{\alpha_1}\cdot\mathsf{S}_{\alpha_2} - \sum_{\substack{\alpha_1<\alpha_2<\alpha_3\in C_{3,5}\\ \alpha_4<\alpha_5}} s_{\alpha_1}s_{\alpha_2}s_{\alpha_3}c_{\alpha_4}c_{\alpha_5}(\mathsf{S}_{\alpha_1}\times\mathsf{S}_{\alpha_2})\cdot\mathsf{S}_{\alpha_3} \\
& - \sum_{\substack{\alpha_1\in C_{1,5}\\ \alpha_2<\alpha_3<\alpha_4<\alpha_5}} s_{\alpha_2}s_{\alpha_3}s_{\alpha_4}s_{\alpha_5}c_{\alpha_1}\big(((\mathsf{S}_{\alpha_2}\times\mathsf{S}_{\alpha_3})\times\mathsf{S}_{\alpha_4})\cdot\mathsf{S}_{\alpha_5} - (\mathsf{S}_{\alpha_2}\cdot\mathsf{S}_{\alpha_3})(\mathsf{S}_{\alpha_4}\cdot\mathsf{S}_{\alpha_5})\big) \\
& - s_1s_2s_3s_4s_5\Big((((\mathsf{S}_1\times\mathsf{S}_2)\times\mathsf{S}_3)\times\mathsf{S}_4)\cdot\mathsf{S}_5 - ((\mathsf{S}_1\times\mathsf{S}_2)\cdot\mathsf{S}_3)(\mathsf{S}_4\cdot\mathsf{S}_5) - \\
& (\mathsf{S}_1\cdot\mathsf{S}_2)((\mathsf{S}_3\times\mathsf{S}_4)\cdot\mathsf{S}_5)\Big),
\end{aligned} \tag{3.11}$$

and

$$\begin{aligned}
\mathsf{Q} = {} & \sum_{\substack{\alpha_1\in C_{1,5}\\ \alpha_2<\alpha_3<\alpha_4<\alpha_5}} s_{\alpha_1}c_{\alpha_2}c_{\alpha_3}c_{\alpha_4}c_{\alpha_5}\mathsf{S}_{\alpha_1} + \sum_{\substack{\alpha_1<\alpha_2\in C_{2,5}\\ \alpha_3<\alpha_4<\alpha_5}} s_{\alpha_1}s_{\alpha_2}c_{\alpha_3}c_{\alpha_4}c_{\alpha_5}\mathsf{S}_{\alpha_1}\times\mathsf{S}_{\alpha_2} + \\
& \sum_{\substack{\alpha_1<\alpha_2<\alpha_3\in C_{3,5}\\ \alpha_4<\alpha_5}} s_{\alpha_1}s_{\alpha_2}s_{\alpha_3}c_{\alpha_4}c_{\alpha_5}\Big((\mathsf{S}_{\alpha_1}\times\mathsf{S}_{\alpha_2})\times\mathsf{S}_{\alpha_3} - (\mathsf{S}_{\alpha_1}\cdot\mathsf{S}_{\alpha_2})\mathsf{S}_{\alpha_3}\Big) + \\
& \sum_{\substack{\alpha_1\in C_{1,5}\\ \alpha_2<\alpha_3<\alpha_4<\alpha_5}} s_{\alpha_2}s_{\alpha_3}s_{\alpha_4}s_{\alpha_5}c_{\alpha_1}\Big(((\mathsf{S}_{\alpha_2}\times\mathsf{S}_{\alpha_3})\times\mathsf{S}_{\alpha_4})\times\mathsf{S}_{\alpha_5} - (\mathsf{S}_{\alpha_2}\cdot\mathsf{S}_{\alpha_3})\mathsf{S}_{\alpha_4}\times\mathsf{S}_{\alpha_5} - \\
& ((\mathsf{S}_{\alpha_2}\times\mathsf{S}_{\alpha_3})\cdot\mathsf{S}_{\alpha_4})\mathsf{S}_{\alpha_5}\Big) + s_1s_2s_3s_4s_5\Big((((\mathsf{S}_1\times\mathsf{S}_2)\times\mathsf{S}_3)\times\mathsf{S}_4)\times\mathsf{S}_5 - \\
& (((\mathsf{S}_1\times\mathsf{S}_2)\times\mathsf{S}_3)\cdot\mathsf{S}_4)\mathsf{S}_5 - (\mathsf{S}_1\cdot\mathsf{S}_2)(\mathsf{S}_3\times\mathsf{S}_4)\times\mathsf{S}_5 - ((\mathsf{S}_1\times\mathsf{S}_2)\cdot\mathsf{S}_3)(\mathsf{S}_4\cdot\mathsf{S}_5)\Big)
\end{aligned} \tag{3.12}$$

where c_j, s_j stand for $\cos\frac{\Delta\hat{\theta}_j}{2}, \sin\frac{\Delta\hat{\theta}_j}{2}$, respectively, and $\alpha_i \in I = \{1,2,3,4,5\}$, the set of non-repeated indices. We denote by $C_{j,k}$ the combinations of j elements from a set of k elements. This formula can be applied to serial chains with less than 5 joints by simply eliminating the extra indices.

From the general formula in Eqs.(3.11) and (3.12) we can state the maximum multi-degree of the system of design equations. For the most general case, the multi-degree is $d = (2m)^{8(n-1)}2^{m(n-1)}$, n being the number of task dual quaternions and m the number of joints of the serial chain: each design equation is, at most, of degree $2m$ if we consider the sines and cosines of the joint angles separately and add the quadratic condition between them.

The degrees obtained with this bound are very big; for the most general robot, the spatial 5R, we obtain a bound of the order of 10^{190}. However, this bound is far from accurate, for several reasons. The equations are multi-linear in many cases; also, they include the solutions for the inverse kinematics, which are not a part of the synthesis problem. In addition to this, two of each set of eight equations can be eliminated if we introduce the Plücker constraints of the joint axes. As an example, for the spatial RR chain, $d \approx 10^{10}$, but the actual number of solutions is equal to six. Later in this Chapter we introduce a way to reduce the total degree by simplifying the design equations.

3.3.2 Maximum number of task positions

The maximum number of task positions corresponds to the maximum number of arbitrary positions that we can fit in the workspace of a given robot topology. When we define a task with the maximum number of positions for that particular robot structure, the given task completely shapes the workspace and we obtain a finite number of robots which are solutions of the design equations. In this Section we

explain how to calculate the maximum number of task positions for a given robot topology, based on counting the parameters of the design equations.

We distinguish between axes S_i of revolute joints and axes of prismatic joints. The axis of a revolute joint is defined by four independent parameters in the associated Plücker coordinate vector. Coordinate-wise, the expression of a revolute joint consists on six elements, three for the direction and three for the moment, plus two constraints, which make the direction a unit vector and the moment perpendicular to the direction,

$$\mathsf{S} = \mathbf{s} + \epsilon \mathbf{s}^0 = \begin{Bmatrix} s_x \\ s_y \\ s_z \end{Bmatrix} + \epsilon \begin{Bmatrix} s_x^0 \\ s_y^0 \\ s_z^0 \end{Bmatrix}, \tag{3.13}$$

$$\mathbf{s} \cdot \mathbf{s} = 1, \tag{3.14}$$

$$\mathbf{s} \cdot \mathbf{s}^0 = 0 \tag{3.15}$$

In contrast, the prismatic joint depends only on two parameters that define the direction of slide of the joint, that is, three coordinates defining a direction $\mathbf{s}$ minus the unit vector constraint.

Let r and t be the number of revolute and prismatic joints in the chain, where $m = r + t$. Then the number of structural parameters is $K = 4r + 2t$, assuming that all the axes are generally defined. Otherwise, we need to subtract the number of constraints among them, which we call c. Given n task positions, we also have $(n-1)r$ joint angles and $(n-1)t$ joint slides that must be determined for reaching the task positions. Thus, total number of design parameters in an n position task is $N = 4r + 2t + (n-1)r + (n-1)t$.

Only six of the eight components of a dual quaternion are independent, therefore only $6(n-1)$ of the design equations in Eq.(3.10) are independent. Here we can add the extra constraints on the joint axes, to obtain $E = 6(n-1) + c$ equations.

We equate the number of equations E and the number of unknowns N to obtain

$$n_{\max} = \frac{3r + t + 6 - c}{6 - r - t}, \quad r + t \leq 5, \tag{3.16}$$

which defines the number of task positions $n_{\max}$ needed to solve exactly for the design variables in the constrained serial chain. Notice that n_{max} achieves a maximum of 21 task positions for the 5R chain. However, this formula determines the maximum number of combined orientations plus translations that the robot can reach. Due to the semi-direct product structure of the group of rigid displacements, this formula is not always directly applicable and some cases must be distinguished, in which orientations can be solved for separately.

3.3.3 Counting for orientations

The design equations can be separated into three equations that define the orientation of the end-effector and three equations that define a combination of translation and rotation; the real part of the dual quaternion design equations corresponds to the orientations. The $3(n-1)$ orientation design equations include the $2r$ unknowns that define the directions of the revolute joints and the $(n-1)r$ associated rotation angles. These equations can be solved -in some cases, independently from the translations-

for a maximum number of task orientations given by

$$n_R = \frac{3+r}{3-r}, \quad r \leq 2. \tag{3.17}$$

This equation is meaningful only for $r = 1$ and $r = 2$, which have the associated values of $n_R = 2$ and $n_R = 5$ maximum task orientations. For $r = 3$ –and beyond– we can reach any task rotation.

3.3.4 Counting for translations

The dual part of the dual quaternion contains the translation terms obtained as a combination of rotations and translations. Both revolute and prismatic joints produce translation; in this case, a necessary condition for a robot to be constrained by the translations is that the number of joints is less than three. Because of this, the direct counting for translations (noted n_{Tl}) is limited to a few special cases and it is not of practical utility.

The counting formula is created by equating the unknowns to the three independent components of the dual part of the dual quaternion,

$$n_{Tl} = \frac{3r+t+3-c}{3-r-t}, \quad r+t \leq 2, \tag{3.18}$$

where c again accounts for extra constraints between axes.

Looking at Eqs.(3.18) and(3.16) we can see that, except in cases with extra constraints c, the maximum number of complete arbitrary positions is always smaller than the maximum number of translations.

However, for those cases in which the rotations are given, counting for the translations may be necessary. This occurs when the orientation of the task position is specified in advance and we must dimension the robot to reach a series of translations. In this case, the revolute joint angles $\theta_i, \quad i = 2 \ldots, r$ are fixed and so are the directions of the revolute joints, but not their locations. We obtain

$$n_T = \frac{2r + t + 3 - c}{3 - t}, \quad t \leq 2, \tag{3.19}$$

where c is again the number of extra constraints on the axes. Notice that we need to have less than three prismatic joints.

3.3.5 The combined counting

Considering apart the translation-limited cases, which are few, to determine the maximum number of task positions that we can reach, we need to compute both n_{max} and n_R. The maximum number of complete positions is limited by n_R. If $n_R > n_{max}$, then we can reach up to n_{max} complete positions. If $n_R < n_{max}$, we can reach at most n_R arbitrary positions. In this case, we can also reach $e = n_T - n_R$ more positions consisting of arbitrary translation terms and of orientation terms that lie in the workspace of the spherical chain obtained using the orientation design equations.

In order to be able to solve for finite number of constrained robots, we need to either define those orientation-specified task positions or add some extra constraints c to the parameters that determine only translation.

The formulas presented in previous sections are not directly applicable in some

cases, presented in Chapter 4. In those cases, certain arrangement of joint axes creates movement within a subgroup of $SE(3)$ and the extra constraints that need to be added are not only distances and angles between joint axes.

3.3.6 Permutation of joints

Notice that the counting is independent of the hierarchical position of the joints along the chain, except for the special cases presented in Chapter 4. According to that, chains with same number and type of joints can be designed for the same maximum number of task positions. Also the structure of the equations will be similar, the only difference being the order of the cross product terms in Eq.(3.11) and (3.12). Some of the chains present the particularity that any of the topologies obtained permuting the roots give the same number of design solutions. This has been shown for the permutations of the RPR - PRR - RRP topology.

3.4 Classification of Orientation-Limited Chains

For those constrained serial chains in which $n_R < n_{max}$, we can solve the orientation equations independently. We call serial chains that have this property *orientation limited chains*. Table 3.1 lists the classes of serial chains that are orientation limited—the permutation of the R and P joints has been included only when it is relevant for the counting. The cylindrical joint C consists of four parameters and two joint variables, equivalent to consider a revolute joint and a prismatic joint plus two constraints that

make the direction of P parallel to the direction of R. The universal joint T is defined as two revolute joints plus two constraints making the directions orthogonal and the locations coincident.

Table 3.1: The orientation-limited constrained serial chains.

Robot	DOF	r	t	c	$\mathbf{n_R}$	$n_{\max}$	n_T	e
RP	2	1	1	0	2	3	3	1
CP	3	1	2	2	2	$2\frac{2}{3}$	4	2
RPP	3	1	2	0	2	3	5	3
PRP	3	1	2	0	2	$3\frac{2}{3}$	7	5
CPP	4	1	3	2	2	5	any	any
RPPP	4	1	3	0	2	6	any	any
TPP	4	2	2	2	4	5	6	2
PTP	4	2	2	2	4	6	7	3
RPC	4	2	2	2	5	6	7	2
RRPP	4	2	2	0	5	6	7	2
RPRP	4	2	2	0	5	7	9	4
TPPP	5	2	3	2	4	13	any	any
CPC	5	2	3	4	5	11	any	any
RPPC	5	2	3	2	5	13	any	any
RRPPP	5	2	3	0	5	15	any	any

We solve an orientation-limited chain to reach n_R arbitrary task positions plus e excess positions for which the directions and angles of the revolute joints are known.

3.5 Classification of Generally-Constrained Chains

Constrained serial chains that have three or more revolute joints are clearly not orientation-limited, and so are the chains for which $n_{max} < n_R$, in which cases the number of task positions is defined by the parameter $n_{\max}$ given in Eq.(3.16).

Table 3.2: Constrained serial chains that are not orientation-limited, 2 to 4 degrees of freedom.

Robot	DOF	r	t	c	n_R	$\mathbf{n}_{\max}$
C	2	1	1	2	2	2
RR	2	2	0	0	5	3
TP	3	2	1	2	4	$3\frac{2}{3}$
RC	3	2	1	2	5	$3\frac{2}{3}$
RRP	3	2	1	0	5	$4\frac{1}{3}$
TR	3	3	0	2	any	$4\frac{1}{3}$
RRR	3	3	0	0	any	5
CC	4	2	2	4	5	5
TC	4	3	1	4	any	6
RRC	4	3	1	2	any	7
TPR	4	3	1	2	any	7
RRRP	4	3	1	0	any	8
TT	4	4	0	4	any	7
TRR	4	4	0	2	any	8
RRRR	4	4	0	0	any	9

Table 3.2 and Table 3.3 list the number of task positions $n_{\max}$ that is possible to define for the design of these chains. We include in this table robots like the CC,

for which $n_{max} = n_R$. These could be also considered as orientation-limited robots because we can solve for the orientations separately.

Table 3.3: Constrained serial chains that are not orientation-limited, five degrees of freedom.

Robot	DOF	r	t	c	n_R	$\mathbf{n}_{\max}$
RCC	5	3	2	4	any	13
TPC	5	3	2	4	any	12
PTC	5	3	2	4	any	13
RRPC	5	3	2	2	any	14
RPRC	5	3	2	2	any	15
RRRPP	5	3	2	0	any	15
RPRPR	5	3	2	0	any	17
TPT	5	4	1	4	any	15
TRC	5	4	1	4	any	15
RRRC	5	4	1	2	any	17
RRRRP	5	4	1	0	any	19
TRT	5	5	0	4	any	17
TRRR	5	5	0	2	any	19
RRRRR	5	5	0	0	any	21

3.6 Translation-Unconstrained Chains

Some chains with five or less degrees of freedom can be *translation-unconstrained*. These chains have one or two revolute joints and three or more prismatic joints. In these cases, the complete task positions that we can define are limited to two for

robots with one revolute joint, or five for robots with two revolute joints, and any translation can be reached regardless of its orientation. Translation-unconstrained robots are not considered constrained robots for us and will not be studied as such. These include the RPPP, CPP, RPPPP, CPPP, RRPPP, TPPP, CRPP, and CCP robots.

3.7 Parameterized Design Equations

The synthesis procedure for a given robot structure is as follows: we compute the maximum number of goal positions using n_{max}, n_R and n_T, define extra constraints c for the axes if needed, and write Eq.(3.10) for the maximum number n of goal dual quaternions. We obtain a set of $8(n-1)+c$ design equations.

As we have pointed out, only six of the eight components of the dual quaternion equality are independent. From the point of view of the solutions of the set of design equations, it is the addition of the Plücker constraints for each axis that allows us to consider only six independent equations. A proof of this is presented in Chapter 4.

As the set of design equations we can use, instead of the whole eight components of each dual quaternion, the six independent components plus the Plücker constraints

for each axis,

$$Q^i = \begin{Bmatrix} q_x + \epsilon q_{x0} \\ q_y + \epsilon q_{y0} \\ q_z + \epsilon q_{z0} \end{Bmatrix}^i = \begin{Bmatrix} p_x + \epsilon p_{x0} \\ p_y + \epsilon p_{y0} \\ p_z + \epsilon p_{z0} \end{Bmatrix}^i, \quad i = 2, \ldots n,$$

$$\mathbf{s}_j \cdot \mathbf{s}_j = 1, \quad \mathbf{s}_j \cdot \mathbf{s}_j^0 = 0, \quad j = 1, \ldots r,$$

$$\mathbf{s}_k \cdot \mathbf{s}_k = 1, \quad k = 1, \ldots t, \tag{3.20}$$

plus the extra geometric constraints c we may want to add. We prefer to use the set in Eq.(3.20) because the Plücker constraints are generally simpler than the components of the dual quaternion.

This is the set of *parameterized design equations*. When using these equations, we solve for the structural parameters defining the joint axes at the reference position, S_i, but also for the values of the joint variables to reach each of the task positions, $\Delta\hat{\theta}_i^j$.

Notice that solving also for the joint variables increases the dimension of the problem by $(r+t)(n-1)$. For instance, for the PRRR robot, the number of total variables is 49, out of which 28 correspond to the values of the joint variables, but for the 5R robot the total number of variables is 130, out of which 100 correspond to joint variables. The number of solutions we have to track tends also to increase because of the multiple solutions for the inverse kinematics. Table 3.4 shows the number of joint variables and structural variables for different robots.

On the other hand, the parameterized equations are not extremely complex, and their multi-degree is at most $2(r+t)$ in each single equation. This makes them well

Table 3.4: Number of structural and joint variables for different constrained robots

Robot	DOF	n	e	structural vars.	joint vars.
C	2	2	0	6	2
RP	2	2	1	9	3
RR	2	3	0	12	4
PC	3	2	3	9	9
PRP	3	2	5	12	13
RC	3	$3\frac{2}{3}$	0	12	6
TP	3	$3\frac{2}{3}$	0	12	6
RRP	3	$4\frac{1}{3}$	0	15	9
RRR	3	5	0	18	12
CC	4	5	0	12	16
PRC	4	5	2	15	20
RPRP	4	5	4	18	24
RRC	4	7	0	18	24
TPR	4	7	0	18	24
RRRP	4	8	0	21	28
RRRR	4	9	0	24	32
RCC	5	13	0	18	60
TPT	5	15	0	21	70
RPRC	5	15	0	21	70
RPRPR	5	17	0	24	80
RRRC	5	17	0	24	80
RRRRP	5	19	0	27	90
RRRRR	5	21	0	30	100

suited for some numerical iterative solution methods.

3.8 Reduced Design Equations

The parameterized design equations can be solved directly for both joint axes and joint variables by using numerical methods. However, we may want to eliminate the joint variables, if possible, and solve for the parameters of the axes only. We call this process "implicitization" of the parametric equations, see [14]. The implicitization reduces the dimension of the system by $(r+t)(n-1)$. The system of *reduced equations* contains only the parameters that define the joint axes. The reduction is also a first step in the elimination to find closed algebraic solutions for the design equations.

The implicitization takes place at each goal dual quaternion and while eliminating the joint variables, it provides formulas for the inverse kinematics of the robot. The procedure used here is general and can be constructed systematically, but for every case there is a better way to choose the number and order of the joint variables that we want to eliminate. In the following section we explain the methodology and present some examples. The process is called *complete reduction* when all the joint variables are eliminated linearly, and *partial reduction* when only some of the joint variables are eliminated linearly and then the equations are further reduced in consecutive steps, if needed.

3.8.1 The complete reduction

The implicitization uses a linear transformation in the vector space of dual quaternions that can be solved for the joint variables. Figure 3.2 shows the transformation for an

RRRR robot.

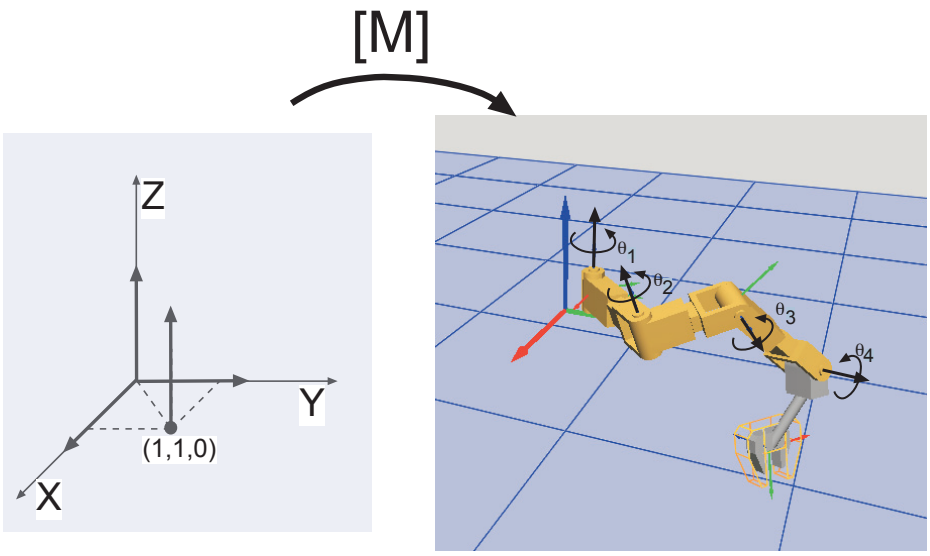

Figure 3.2: The transformation from the coordinate robot to the actual dimensions.

It has the advantage of separating the joint variables, contained in the vector $\hat{V}$, from the structural variables of the joint axes contained in the matrix $[\hat{M}]$,

$$\hat{Q}(\hat{\theta}_1, \ldots, \hat{\theta}_m) = [\hat{M}]\hat{V}(\hat{\theta}_1, \ldots, \hat{\theta}_m) = \hat{P}. \tag{3.21}$$

Solving the linear system, we obtain the joint variables contained in the vector $\hat{V}$,

$$\hat{V}(\hat{\theta}_1, \ldots, \hat{\theta}_m) = [\hat{M}]^{-1}\hat{P} \tag{3.22}$$

The solution of the system provides the inverse kinematics, and the subspaces of the solutions, stated as relations among the components of the eight-dimensional vector $\hat{V}$, are used to define the reduced design equations.

The implicitization can be visualized in some cases as a transformation from a "coordinate robot", whose axes are aligned with the coordinate axes in some fashion,

to the actual robot dimensions. This is a non-rigid transformation, as it does not keep angles and distances between lines. However, it is a linear transformation from dual quaternions to dual quaternions.

For serial chains with more than three joints, the vector $\hat{V}$ has more than eight components; in fact its dimension is 2^m, m being the total number of joints. For $m = 2$ we use either the real or the dual part of the dual quaternion to construct the linear system, depending on the type of joints; for $m = 3$, $\hat{V}$ has eight components and the linear system contains the whole dual quaternion. For $m = 4$ and $m = 5$, we need to create a sum of linear systems. In this case, the procedure does not seem to be of practical use and instead, several steps of reduction are performed.

We illustrate this using the spatial 5R chain. In this case, $\hat{V}$ has 32 components. We separate it in four vectors of dimension 8 just for handling purposes, and sort their terms to obtain the system

$$\left[\hat{M}_1\right]\begin{Bmatrix} c_{12345} \\ s_1c_{2345} \\ s_2c_{1345} \\ s_3c_{1245} \\ c_{1235}s_4 \\ c_{1234}s_5 \\ s_{12}c_{345} \\ s_{13}c_{245} \end{Bmatrix} + \left[\hat{M}_2\right]\begin{Bmatrix} s_{14}c_{235} \\ s_{15}c_{234} \\ c_{145}s_{23} \\ c_{135}s_{24} \\ c_{134}s_{25} \\ c_{125}s_{34} \\ c_{124}s_{35} \\ c_{123}s_{45} \end{Bmatrix} + \left[\hat{M}_3\right]\begin{Bmatrix} s_{123}c_{45} \\ s_{124}c_{35} \\ s_{125}c_{34} \\ s_{134}c_{25} \\ s_{135}c_{24} \\ s_{145}c_{23} \\ c_{15}s_{234} \\ c_{14}s_{235} \end{Bmatrix} + \left[\hat{M}_4\right]\begin{Bmatrix} c_{13}s_{245} \\ c_{12}s_{345} \\ s_{1234}c_5 \\ s_{1235}c_4 \\ s_{1245}c_3 \\ s_{1345}c_2 \\ c_1s_{2345} \\ s_{12345} \end{Bmatrix} = \hat{P}, \tag{3.23}$$

where c_{ij} denotes $\cos\frac{\theta_i}{2}\cos\frac{\theta_j}{2}$, and so on. The matrices have the structure

$$
\begin{aligned}
[\hat{M}_1] &= \begin{bmatrix} \hat{S}(0) & \hat{S}_1 & \hat{S}_2 & \hat{S}_3 & \hat{S}_4 & \hat{S}_5 & \hat{S}_{12} & \hat{S}_{13} \end{bmatrix} \\
[\hat{M}_2] &= \begin{bmatrix} \hat{S}_{14} & \hat{S}_{15} & \hat{S}_{23} & \hat{S}_{24} & \hat{S}_{25} & \hat{S}_{34} & \hat{S}_{35} & \hat{S}_{45} \end{bmatrix} \\
[\hat{M}_3] &= \begin{bmatrix} \hat{S}_{123} & \hat{S}_{124} & \hat{S}_{125} & \hat{S}_{134} & \hat{S}_{135} & \hat{S}_{145} & \hat{S}_{234} & \hat{S}_{235} \end{bmatrix} \\
[\hat{M}_1] &= \begin{bmatrix} \hat{S}_{245} & \hat{S}_{345} & \hat{S}_{1234} & \hat{S}_{1235} & \hat{S}_{1245} & \hat{S}_{1345} & \hat{S}_{2345} & \hat{S}_{12345} \end{bmatrix},
\end{aligned} \tag{3.24}
$$

where the $\hat{S}$ are dual quaternions written in columns as 8-vectors, and $\hat{S}_{ij} = \hat{S}_i\hat{S}_j$.

3.8.2 The partial reduction

Partial reduction is used with advantage in many cases because of the high cost associated with inverting the matrix $[M]$. It follows basically the same principles as the complete reduction, but in this case the coordinate robot has less degrees of freedom than the actual robot,

$$\hat{Q}(\hat{\theta}_1, \ldots, \hat{\theta}_m) = [\hat{M}(\hat{\theta}_{k+1}, \ldots, \hat{\theta}_m)]\hat{V}(\hat{\theta}_1, \ldots, \hat{\theta}_k) = \hat{P}. \tag{3.25}$$

Notice that in Eq.(3.25) the joint variables contained in $\hat{V}$ need not be the first k joint variables of the robot, and can be chosen arbitrarly. The solution for the k variables of the coordinate robot is now a function of the $m-k$ joint variables left plus the structural variables. Further steps are performed to eliminate those joint variables, in a case-by-case basis.

Here we present three partial eliminations that we use to solve for different constrained robots. They correspond to the cases of eliminating one rotation angle, two

rotation angles, and two rotation angles plus one translation. All three cases present an easy algebraic solution.

Reduction of one revolute joint variable

For robots with only one revolute joint, eliminating its joint variable leads to the expected results of identifying the direction of the revolute joint axis with the direction of the goal quaternion. Notice that, in this case, a maximum of two orientations can be reached.

We collect by sines and cosines of θ and consider any robot with only a revolute joint axis, S. The real part of the dual quaternion kinematics equations of the robot, $\hat{Q}_R$, shown in Eqs.(3.11) and (3.12), contains only the joint direction $\mathbf{s}$, and we can write it as

$$\hat{Q}_R(\mathsf{S}(\theta)) = [\hat{M}]V(\theta) = \hat{P}_R, \tag{3.26}$$

which expands to

$$\begin{bmatrix} s_x & 0 \\ s_y & 0 \\ s_y & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \sin\frac{\theta}{2} \\ \cos\frac{\theta}{2} \end{Bmatrix} = \begin{Bmatrix} p_x \\ p_y \\ p_z \\ p_w \end{Bmatrix} \tag{3.27}$$

We impose that every 2×2 linear system of rank 2 must give the same solution, that is, the set of systems

$$\begin{bmatrix} s_i & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \sin\frac{\theta}{2} \\ \cos\frac{\theta}{2} \end{Bmatrix} = \begin{Bmatrix} p_i \\ p_w \end{Bmatrix}, \quad i = x, y, z, \tag{3.28}$$

yields the set of solutions

$$\begin{aligned} \cos\frac{\theta}{2} &= p_w \\ \sin\frac{\theta}{2} &= \frac{p_i}{s_i}, \quad i = x, y, z. \end{aligned} \tag{3.29}$$

This solution states that the direction and angle of the revolute axis S is the same as the direction and angle of the goal dual quaternion. Equating the values of $\sin\frac{\theta}{2}$ we obtain the two reduced design equations,

$$\begin{aligned} s_y p_x - s_x p_y &= 0 \\ s_z p_x - s_x p_z &= 0, \end{aligned} \tag{3.30}$$

which correspond to making the axes parallel, that is, $\mathbf{s} \times \mathbf{p} = 0$. Together with the Plücker condition, we obtain the solution for $\mathbf{s}$

$$\begin{aligned} s_x &= \frac{p_x}{\sqrt{p_x^2 + p_y^2 + p_z^2}} \\ s_y &= \frac{p_y}{\sqrt{p_x^2 + p_y^2 + p_z^2}} \\ s_z &= \frac{p_z}{\sqrt{p_x^2 + p_y^2 + p_z^2}}. \end{aligned} \tag{3.31}$$

There exist two solutions, corresponding to the two positive and negative parallel directions; the second solution can be obtained by negating the first one. This is true for all cases and by default we will not consider them as different designs in any case.

Reduction of two revolute joint variables

We use the following coordinate robot:

$$V(\theta_1, \theta_2) = X(\theta_1)Y(\theta_2) = \begin{Bmatrix} \sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2} \\ \cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2} \\ \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2} \\ \cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2} \end{Bmatrix} \tag{3.32}$$

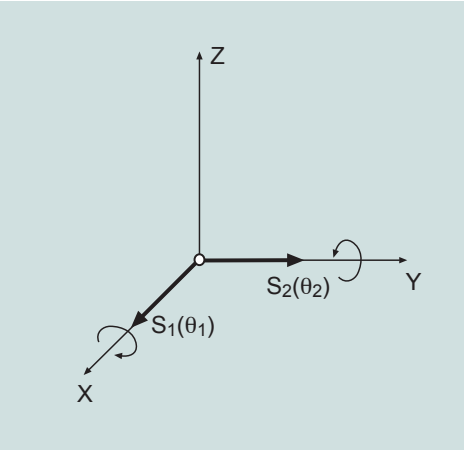

Figure 3.3: The coordinate robot for two revolute variables.

Notice that the dual part is equal to zero; the rotations can be eliminated by using only the real part of the dual quaternion. The solution of the system will be a function of the rest of revolute joints of the robot.

For the most general case, the 5R robot,

$$\hat{Q}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5) = \hat{S}_1(\Delta\theta_1)\hat{S}_2(\Delta\theta_2)\hat{S}_3(\Delta\theta_3)\hat{S}_4(\Delta\theta_4)\hat{S}_5(\Delta\theta_5), \tag{3.33}$$

the expression of the design equations becomes,

$$[\hat{M}(\theta_3, \theta_4, \theta_5)]V(\theta_1, \theta_2) = \hat{P}_R, \tag{3.34}$$

where $\hat{P}_R$ denotes the real part of the dual quaternion, and the columns of the 4×4 matrix $[\hat{M}]$ are the quaternions,

$$\left[\hat{M}\right] = \left[S_1 S_{345}(\theta) \quad S_2 S_{345}(\theta) \quad S_{12} S_{345}(\theta) \quad S_{345}(\theta)\right]. \tag{3.35}$$

where

$$\begin{aligned} S_1 S_{345}(\theta) &= S_1(\pi) S_3(\theta_3) S_4(\theta_4) S_5(\theta_5) \\ S_2 S_{345}(\theta) &= S_2(\pi) S_3(\theta_3) S_4(\theta_4) S_5(\theta_5) \\ S_{12} S_{345}(\theta) &= S_1(\pi) S_2(\pi) S_3(\theta_3) S_4(\theta_4) S_5(\theta_5) \\ S_{345}(\theta) &= S_3(\theta_3) S_4(\theta_4) S_5(\theta_5) \end{aligned} \tag{3.36}$$

Notice that $S_i(\pi) = 0 + \mathbf{s}_i$, the vector quaternion in the direction of the i^{th} joint axis. It is also important to notice that we can eliminate any pair of joint variables, and expressing it similarly as the linear function of quaternions in Eq.(3.35) but keeping the order of the axes in the actual serial chain.

In the special case of a robot with only two revolute joints, like the one in Figure 3.4,

$$\hat{Q}(\theta_1, \theta_2, d_3, \ldots d_m) = \hat{S}_1(\Delta\theta_1) \hat{S}_2(\Delta\theta_2) \hat{S}_3(\Delta d_3) \ldots \hat{S}_m(\Delta d_m), \tag{3.37}$$

the partial reduction simplifies to

$$[\hat{M}] V(\theta_1, \theta_2) = \hat{P}_R, \tag{3.38}$$

where $\hat{P}_R$ denotes the quaternion that forms the real part of the dual quaternion.

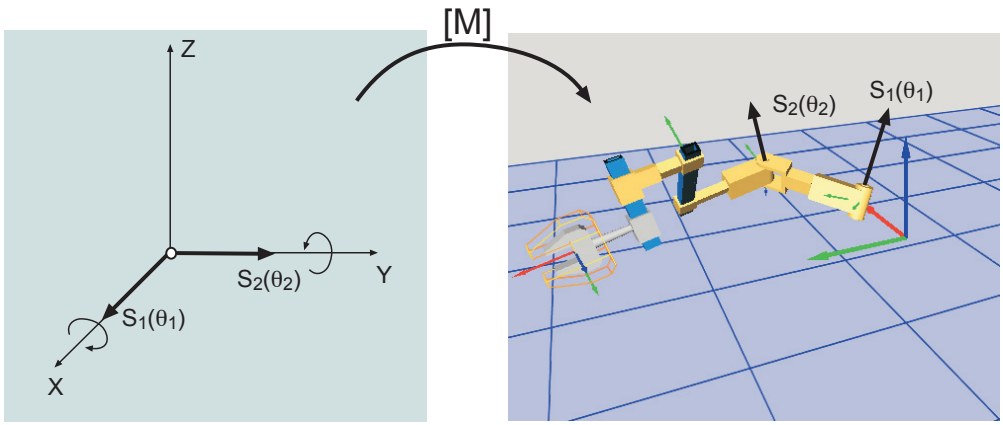

Figure 3.4: The coordinate robot transformed to an RRPP robot.

The columns of the 4×4 matrix $[M]$ are the vector quaternions,

$$\left[\hat{M}\right] = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \mathbf{s}_1 \times \mathbf{s}_2 & \vec{0} \\ 0 & 0 & -\mathbf{s}_1 \cdot \mathbf{s}_2 & 1 \end{bmatrix} = \begin{bmatrix} S_1(\pi) & S_2(\pi) & S_1(\pi)S_2(\pi) & S(0) \end{bmatrix}. \tag{3.39}$$

We invert this matrix by finding the row vectors orthogonal to the column vectors in $[M]$,

$$\left[\hat{M}^{-1}\right] = \frac{1}{(\mathbf{s}_1 \times \mathbf{s}_2) \cdot (\mathbf{s}_1 \times \mathbf{s}_2)} \begin{bmatrix} \mathbf{s}_1 - (\mathbf{s}_1 \cdot \mathbf{s}_2)\mathbf{s}_2 & 0 \\ \mathbf{s}_2 - (\mathbf{s}_1 \cdot \mathbf{s}_2)\mathbf{s}_1 & 0 \\ \mathbf{s}_1 \times \mathbf{s}_2 & 0 \\ (\mathbf{s}_1 \cdot \mathbf{s}_2)\mathbf{s}_1 \times \mathbf{s}_2 & (\mathbf{s}_1 \times \mathbf{s}_2) \cdot (\mathbf{s}_1 \times \mathbf{s}_2) \end{bmatrix}, \tag{3.40}$$

and solve for $V(\theta_1, \theta_2)$,

$$\begin{aligned} \sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2} &= \frac{\mathbf{s}_1 \cdot \mathbf{p} - \mathbf{s}_1 \cdot \mathbf{s}_2 \mathbf{s}_2 \cdot \mathbf{p}}{(\mathbf{s}_1 \times \mathbf{s}_2) \cdot (\mathbf{s}_1 \times \mathbf{s}_2)} \\ \cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2} &= \frac{\mathbf{s}_2 \cdot \mathbf{p} - \mathbf{s}_1 \cdot \mathbf{s}_2 \mathbf{s}_1 \cdot \mathbf{p}}{(\mathbf{s}_1 \times \mathbf{s}_2) \cdot (\mathbf{s}_1 \times \mathbf{s}_2)} \\ \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2} &= \frac{\mathbf{p} \cdot (\mathbf{s}_1 \times \mathbf{s}_2)}{(\mathbf{s}_1 \times \mathbf{s}_2) \cdot (\mathbf{s}_1 \times \mathbf{s}_2)} \\ \cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2} &= \frac{\mathbf{p} \cdot (\mathbf{s}_1 \times \mathbf{s}_2)(\mathbf{s}_1 \cdot \mathbf{s}_2)}{(\mathbf{s}_1 \times \mathbf{s}_2) \cdot (\mathbf{s}_1 \times \mathbf{s}_2)} + p_w, \end{aligned} \tag{3.41}$$

where $\hat{P}_R = p_w + \mathbf{p} = \{p_x, p_y, p_z, p_w\}$.

The results in Eq.(3.41) provide us with the inverse kinematics for the variables θ_1 and θ_2. These values are substituted into the dual part $\hat{Q}_D(\theta_1, \theta_2, d_3, \ldots d_m) - \hat{P}_D$ of the dual quaternion design equations to obtain equations in the prismatic variables d_i only. The prismatic variables can be then linearly eliminated.

The reduced design equations are created from the solutions to the inverse kinematics.

The components of $V(\theta_1, \theta_2)$ are not independent variables; the relation between them captures the fact that they are cosines and sines of angles,

$$\mathcal{R}_1 \quad : \quad \frac{\sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}}{\cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}} = \frac{\sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}}{\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2}}. \tag{3.42}$$

When using the values of Eq.(3.41), we obtain the first reduced design equation $\mathcal{R}_1$,

$$\begin{aligned}&\big[\big(\mathbf{p}\cdot(\mathbf{s}_1\times\mathbf{s}_2)\big)\big(\mathbf{p}\cdot(\mathbf{s}_1\times\mathbf{s}_2)(\mathbf{s}_1\cdot\mathbf{s}_2)+p_w(\mathbf{s}_1\times\mathbf{s}_2)\cdot(\mathbf{s}_1\times\mathbf{s}_2)\big)-\\ &\big(\mathbf{p}\cdot((-\mathbf{s}_1\times\mathbf{s}_2)\times\mathbf{s}_2)\big)\big(\mathbf{p}\cdot(-\mathbf{s}_1\times(\mathbf{s}_1\times\mathbf{s}_2))\big)\big]/(\mathbf{s}_1\times\mathbf{s}_2)\cdot(\mathbf{s}_1\times\mathbf{s}_2)=0\end{aligned} \tag{3.43}$$

This equation can be collected in the form

$$(p_w\mathbf{s}_1\cdot\mathbf{p}\times\mathbf{s}_2+\mathbf{s}_1\cdot(\mathbf{p}\times(\mathbf{p}\times\mathbf{s}_2)))((\mathbf{s}_1\times\mathbf{s}_2)\cdot(\mathbf{s}_1\times\mathbf{s}_2))^2/(\mathbf{s}_1\times\mathbf{s}_2)\cdot(\mathbf{s}_1\times\mathbf{s}_2)=0 \tag{3.44}$$

If we require that $\mathbf{s}_1 \times \mathbf{s}_2 \neq 0$, which restricts the pair of revolute joints from generating a pure planar movement, then we obtain

$$\mathcal{R}: \quad p_w\mathbf{s}_1\cdot\mathbf{p}\times\mathbf{s}_2+\mathbf{s}_1\cdot(\mathbf{p}\times(\mathbf{p}\times\mathbf{s}_2))=0. \tag{3.45}$$

It is interesting to notice that this equation corresponds to both the matrix formulation of geometric constraint equations [58] and the equivalent screw triangle formulation [97], in the following way,

$$\mathcal{R} = \mathbf{s}_1 \cdot ([A_{Euler} - I]\mathbf{s}_2) = \tan\frac{\psi}{2} - \frac{\mathbf{s}_1 \cdot (\mathbf{p}_u \times \mathbf{s}_2)}{(\mathbf{p}_u \times \mathbf{s}_1) \cdot (\mathbf{p}_u \times \mathbf{s}_2)}, \tag{3.46}$$

where A_{Euler} corresponds to the rotation matrix expressed by using the Euler parameters [9], and $\mathbf{p}_u = \mathbf{p}/|\mathbf{p}|$ with $\sin\frac{\psi}{2} = |\mathbf{p}|$ and $\cos\frac{\psi}{2} = p_w$. This shows how all three formulations lead to the same set of solutions, and represents the best case of reduction, when the implicitization creates the set of polynomial equations with no extraneous roots.

Reduction of two revolute and one prismatic joint variables

For this reduction we use the coordinate robot:

$$\hat{V}(\theta_1, \theta_2, d) = \hat{X}(\theta_1)\hat{Z}(d)\hat{Y}(\theta_3) = \begin{Bmatrix} \sin\frac{\theta_1}{2}\cos\frac{\theta_3}{2} \\ \cos\frac{\theta_1}{2}\sin\frac{\theta_3}{2} \\ \sin\frac{\theta_1}{2}\sin\frac{\theta_3}{2} \\ \cos\frac{\theta_1}{2}\cos\frac{\theta_3}{2} \end{Bmatrix} + \epsilon \begin{Bmatrix} -\frac{d}{2}\cos\frac{\theta_1}{2}\sin\frac{\theta_3}{2} \\ -\frac{d}{2}\sin\frac{\theta_1}{2}\cos\frac{\theta_3}{2} \\ \frac{d}{2}\cos\frac{\theta_1}{2}\cos\frac{\theta_3}{2} \\ \frac{d}{2}\sin\frac{\theta_1}{2}\sin\frac{\theta_3}{2} \end{Bmatrix} \tag{3.47}$$

We apply the reduction for instance to an RPRRR robot,

$$\hat{Q}(\theta_1, d, \theta_3, \theta_4, \theta_5) = \hat{S}_1(\Delta\theta_1)\hat{S}_2(\Delta d)\hat{S}_3(\Delta\theta_3)\hat{S}_4(\Delta\theta_4)\hat{S}_5(\Delta\theta_5), \tag{3.48}$$

and the design equations become,

$$[\hat{M}(\theta_4, \theta_5)]\hat{V}(\theta_1, d, \theta_3) = \hat{P}. \tag{3.49}$$

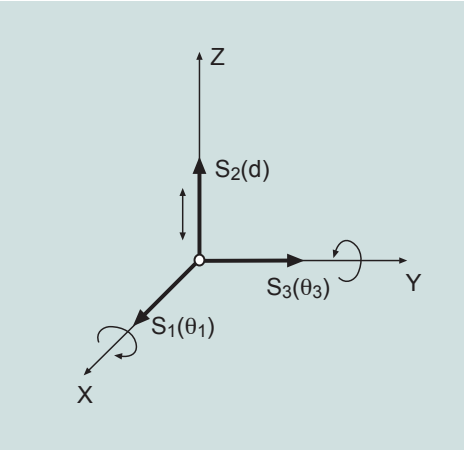

Figure 3.5: The coordinate robot for two revolute and one prismatic variables.

If we write $\hat{V}$ and $\hat{P}$ as eight-dimensional vectors, $[\hat{M}]$ is an 8×8 matrix with general structure

$$[\hat{M}] = \begin{bmatrix} A & \vdots & 0 \\ \cdots & \vdots & \cdots \\ B & \vdots & C \end{bmatrix}, \tag{3.50}$$

where

$$\begin{bmatrix} A \\ \cdots \\ B \end{bmatrix} = \begin{bmatrix} \hat{S}_1\hat{S}_{45}(\theta) & \hat{S}_3\hat{S}_{45}(\theta) & \hat{S}_{13}\hat{S}_{45}(\theta) & \hat{S}_{45}(\theta) \end{bmatrix} \tag{3.51}$$

contains only the factors corresponding to revolute joints, and

$$\Big[C\Big] = \Big[-\hat{S}_{23}\hat{S}_{45}(\theta) \quad -\hat{S}_{12}\hat{S}_{45}(\theta) \quad \hat{S}_2\hat{S}_{45}(\theta) \quad \hat{S}_{123}\hat{S}_{45}(\theta)\Big]_R, \tag{3.52}$$

where the subindex R in Eq.(3.52) indicates real part of the quaternion –dual parts never multiply together. The abbreviatons are used as indicated in Eq.(3.36).

As in the previous case, we can pick any group of three joint variables to perform the elimination as long as the order is maintained.

We solve for the components of the vector $\hat{V}(\theta_1, d, \theta_2)$ by inverting the matrix $[\hat{M}]$,

$$\hat{V}(\theta_1, d, \theta_2) = \left[\hat{M}\right]^{-1} \hat{P} \tag{3.53}$$

The inversion of matrix $[\hat{M}]$ is not very complicated if we notice that

$$\left[\hat{M}\right]^{-1} = \begin{bmatrix} A^{-1} & \vdots & 0 \\ \cdots & \vdots & \cdots \\ -C^{-1}BA^{-1} & \vdots & C^{-1} \end{bmatrix}. \tag{3.54}$$

This inversion is a special case of the general Schur's complement formula.

The expression of $[A]^{-1}$ has been already computed in Eq.(3.34), and if the robot has only two revolute joints, it coincides with Eq.(3.40). The submatrix $[B]$ contains the dual part of the same dual quaternions, while the submatrix $[C]$ contains dual quaternions of both prismatic and revolute joints. We solve for the joint variables θ_1, θ_2 and d as the rotational components,

$$\begin{Bmatrix} \sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2} \\ \cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2} \\ \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2} \\ \cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2} \end{Bmatrix} = [A]^{-1}\hat{P}_R \tag{3.55}$$

and the translational components,

$$\begin{Bmatrix} -\frac{d}{2}\cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2} \\ -\frac{d}{2}\sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2} \\ \frac{d}{2}\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2} \\ \frac{d}{2}\sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2} \end{Bmatrix} = -([C]^{-1}[B][A]^{-1})\hat{P}_R + [C]^{-1}\hat{P}_0, \tag{3.56}$$

where $\hat{P}_R = \mathbf{p} + p_w$, $\hat{P}_0 = \mathbf{p}^0 + p_{w0}$ are the real and dual parts of the task dual quaternion, respectively.

Corresponding to these solutions for the inverse kinematics, there are some relations among the components of $\hat{V}$ which are going to form the set of reduced equations; for the components in Eq.(3.55),

$$\mathcal{R}_1 \quad : \quad \frac{\sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}}{\cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}} = \frac{\sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}}{\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2}}, \tag{3.57}$$

and for the components in Eq.(3.56) we can define the two more independent relations. Those can be expressed in different ways; the ones below make the equations well-conditioned,

$$\begin{aligned} \mathcal{L}_1 : \quad & \frac{\frac{\mathrm{d}}{2}\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2}}{\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2}} = \frac{\frac{\mathrm{d}}{2}\sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}}{\sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}}, \\ \mathcal{L}_2 : \quad & \frac{\frac{\mathrm{d}}{2}\cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}}{\cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}} = \frac{\frac{\mathrm{d}}{2}\sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}}{\sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}}. \end{aligned} \tag{3.58}$$

These equations may be free of additional joint variables, and hence form the set of reduced design equatons, or they may contain some more joint variables, in which case we perform further reduction steps to create the set of reduced design equations.

3.8.3 The final set of reduced equations

The reduced equations differ according to the robot topology and the type of reduction we decide to follow, but, in general, at the end of the reduction process we will obtain a condition $\mathcal{R}_1$ from the elimination of revolute joint variables, and several conditions, $\{\mathcal{L}_\mathrm{i}\}$ from the elimination of revolute and prismatic joint variables in the dual part of the dual quaternion.

The number of conditions needed for each case can be computed from the counting formula in Eq.(3.16); we are eliminating $(r+t)(n-1)$ variables from a set of $6(n-1)$

equations. Hence we need to obtain $k = 6 - (r + t)$ reduced design equations after the implicitization process.

The complete set of equations consists of $\{\mathcal{R}_1, \{\mathcal{L}_\mathrm{i}\}\}^\mathrm{j}$ equations for each of the $n - 1$ task dual quaternions. To these we must add the Plücker conditions for each axis and any extra constraint we want to impose. The total set of reduced equations for a serial chain consisting of m joints, that can be synthesized for a maximum of n positions, is

$$\begin{aligned} &\{\mathcal{R}_1, \mathcal{L}_1, \ldots, \mathcal{L}_{\mathrm{k}-1}\}^\mathrm{j}, \quad \mathrm{j} = 2, \ldots \mathrm{n}, \\ &\mathbf{s}_i \cdot \mathbf{s}_i = 1, \quad \mathbf{s}_i \cdot \mathbf{s}_i^0 = 0, \quad i = 1, \ldots, m, \\ &c_1, \ldots, c_c. \end{aligned} \tag{3.59}$$

It does not contain any joint variable and can be solved for the Plücker coordinates of the axes.

3.9 Solving the Design Equations

We use numerical methods to solve the parameterized design equations; the goal of the implicitization process is, in the first place, to reduce the size of the parameterized system. However, this reduction may introduce extraneous roots that need to be eliminated. The final goal is to arrive to the set of equations that captures exactly the geometric constraints of the serial chain, and, if possible, to obtain a closed algebraic solution.

The parameterized design equations are numerically well-behaved. We used the

Newton-Raphson and Levenberg-Marquardt solver *fsolve* of MATLAB to test the design equations and to solve for those chains for which the implicitization results too complicated. The calculations were performed on an Apple G4 computers, at 733 MHz and 500 Mhz. In addition to the solutions of the exact synthesis problem, the parameterized design equations can be used also for design optimization along a trajectory, see [56].

The reduced design equations were solved numerically using same methods. For some cases, the equations were further reduced using resultant methods and closed algebraic solutions were found.

In Table 3.5 we present a summary of all the constrained robots for which the dual quaternion design equations were created and solved. In the table, "DOF" indicates the degrees of freedom of the chain, n is the maximum number of complete positions that we can define, c counts for extra constraints imposed on the axes, and e indicates the number of extra positions, that is, task translations with orientation belonging to the workspace. Under "Des.Eq.", we indicate whether the design equations have been solved in parameterized (p) or reduced (r) form, or if a closed solution (c) has been found. "Sols." denotes the number of solutions found and t the time it took to compute them, whereas "Method." indicates the solution method: it can be algebraic (A) –and, in particular, when resultants (R) were used to reduce the problem. Or it can be a numeric method, polynomial homotopy continuation (PHC) or Newton-Raphson algorithms (NR). The last column, "Section", indicates

the section of Chapter 5 where the results can be found. The following chapter presents the particular solution process for these cases.

Table 3.5: Constrained robots solved using dual quaternion synthesis.

Robot	DOF	n	c	e	Des. Eq.	Sols.	t	Method.	Section
RP	2	2	0	1	c	1	$<$ 1 sec.	A	5.2.1
RR	2	3	0	0	c	6 (2 real)	$<$ 1 sec.	A	5.2.2
RPP	3	2	2	3	c	6	$<$ 1 sec.	A	5.3.1
RRP	3	4	1	0	c	12	$<$ 1 sec.	R	5.3.2
TP	3	3	4	0	c	12	$<$ 1 sec.	R	5.3.3
RRR	3	5	0	0	p/r	19	18 h.	NR	5.3.4
RPC	4	5	4	0	c	6	$<$ 1 sec.	R	5.4.2
RPRP	4	5	0	4	p/r	21 (7 real)	43 min.	PHC	5.4.1
TPR	4	7	2	0	p/r	6	12 h.	NR	5.4.5
CRR	4	7	2	0	p/r	52	14 h.	NR	5.4.4
PRRR	4	8	0	0	p	91	9.8 h.	NR	5.4.3
RRRR	4	9	0	0	p	11	48 h.	NR	5.4.6
PPS	5	6	0	0	c	10	$<$ 1 sec.	A	5.5.1
RCC	5	13	0	0	p	1	14.7 h.	NR	5.5.2

3.10 Relation to Previous Synthesis Methods

In this section we demonstrate the relation between the dual quaternion formulation and some of the synthesis methods presented in Chapter 1. To compare with the planar complex formulation of Sandor [88], called the *standard form*, we reduce the dual quaternion synthesis to planar quaternions. We follow [55] for some of the

derivations. We then compare the result of the dual quaternion product with the geometric screw triangle used by Tsai [97]. For the calculations on the screw triangle, we follow [9].

3.10.1 Planar quaternions and the standard form

A planar quaternion can be derived from a dual quaternion with a general expression as that of Eq.(2.18) if we realize that the displacement in the **z**-direction and the $\mathbf{x}-\mathbf{y}$ components of the rotation must be zero. We obtain the expression in Eq.(2.14).

For a relative displacement of angle θ and translation $\mathbf{d}$, the planar quaternion is expressed as a rotation about the pole $\mathbf{p}$ of the relative displacement,

$$P = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}k + p_y \sin\frac{\theta}{2}i\epsilon - p_x \sin\frac{\theta}{2}j\epsilon, \tag{3.60}$$

or, using the displacement vector $\mathbf{d} = (d_x, d_y)$,

$$P = \cos\frac{\theta}{2} + \sin\frac{\theta}{2}k + (\frac{d_x}{2}\cos\frac{\theta}{2} + \frac{d_y}{2}\sin\frac{\theta}{2})i\epsilon - (\frac{d_x}{2}\sin\frac{\theta}{2} - \frac{d_y}{2}\cos\frac{\theta}{2})j\epsilon. \tag{3.61}$$

McCarthy [55] decomposes the general planar displacement in pure translation and pure rotation, which can be expressed as complex numbers,

$$\begin{aligned} R(\theta) &= \cos\frac{\theta}{2} + k\sin\frac{\theta}{2} = e^{k\frac{\phi}{2}}, \\ T(\mathbf{d}) &= 1 + \frac{1}{2}(d_x + d_y k)i\epsilon = 1 + \frac{1}{2}\mathbf{D}i\epsilon. \end{aligned} \tag{3.62}$$

Using this notation, a general planar displacement can be computed as $T(\mathbf{p})R(\theta)T(-\mathbf{p})$, which gives the expression

$$P = T(\mathbf{p})R(\theta)T(-\mathbf{p}) = (1 - \frac{1}{2}\mathbf{P}(1 - e^{k\phi})i\epsilon)e^{k\frac{\phi}{2}} \tag{3.63}$$

The Standard Form

The complex number synthesis procedure developed in [88] for rigid-body guidance generates the design equations from loop equations.

A planar displacement is expressed as a complex number whose module contains the translation and whose angle accounts for the rotation. The composition of planar displacements is expressed as the product of modules and exponential functions.

If we are expressing a displacement from a reference configuration Z, then the change in length and orientation of the vector by amounts $\rho = |Z'|/|Z|$ and ϕ will yield the complex number,

$$Z' = |Z'|e^{\theta+\phi} = \frac{|Z'|}{|Z|}|Z|e^{\theta}e^{\phi} = Z\rho e^{\phi} \tag{3.64}$$

The relative displacement ρe^{ϕ} is denoted as stretch rotation operator in [88].

We demonstrate the procedure for a planar RR chain. In Figure 3.6, we have a planar RR chain in two positions$\{M_1\}$ and $\{M_2\}$, whose links are defined by the complex numbers W and Z in the first position. In the standard form, angles are measured from a fixed reference frame $\{\mathsf{F}\}$. They are denoted β_i and α_i when the links W and Z are in position i. However, in the planar quaternion formulation, we use for the second and consecutive links the relative angle from the previous link, denoted here by α_{iR}.

Considering position 1 as the reference configuration, we compute the relative angles $\beta_{1i} = \beta_i - \beta_1 = \beta_{1iR}$ and $\alpha_{1i} = \alpha_i - \alpha_1$, but $\alpha_{1i} = \beta_{1i} + \alpha_{1iR}$. Notice that, as the moving frame is attached to the second link, it is also true that $\alpha_{1i} = \theta_i - \theta_1$. We

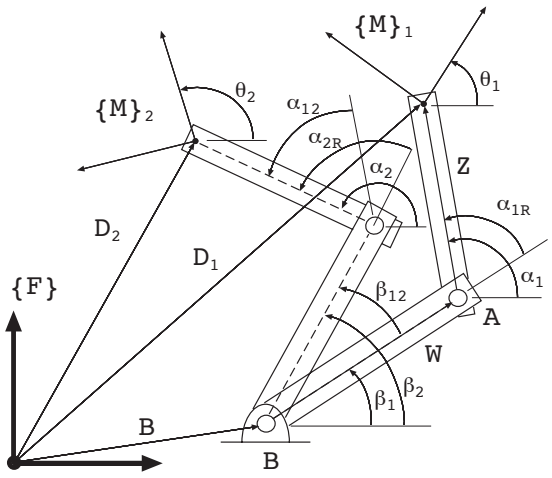

Figure 3.6: A planar RR chain in two positions.

also denote $\delta_{1i} = D_i - D_1$ the relative translation.

The standard form equations are created by stating the loop equations for the chain from position 1 to position i; we use $i = 2$ for this example. Compute the vector addition going from position 2 to position 1,

$$B + We^{i\beta_{12}} + Ze^{i\alpha_{12}} - D_2 + D_1 - Z - W - B = 0, \tag{3.65}$$

to obtain the standard form,

$$W(e^{i\beta_{12}} - 1) + Z(e^{i\alpha_{12}} - 1) = \delta_{12}. \tag{3.66}$$

Planar Quaternion Synthesis

We create the design equations for the same RR chain using the planar quaternion procedure in the form of Eq.(3.63), as a rotation about **A** followed by a rotation about **B** and equated to the relative planar quaternion for the task position,

$$B(\beta_{12R})A(\alpha_{12R}) = P_{12}. \tag{3.67}$$

The expansion of the planar quaternion product gives

$$\big((1+\frac{1}{2}\mathbf{B}(1-e^{k\beta_{12}})i\epsilon+\frac{1}{2}\mathbf{A}(1-e^{k\alpha_{12R}})i\epsilon e^{-k\beta_{12}}\big)e^{k\frac{\beta_{12}+\alpha_{12R}}{2}}=(1+\frac{1}{2}\mathbf{P}(1-e^{k\theta_{12}})i\epsilon)e^{k\frac{\theta_{12}}{2}}, \tag{3.68}$$

where $\mathbf{P}$ is the pole of the relative transformation from $\{M_1\}$ to $\{M_2\}$. From this expression we can eliminate the equality $e^{k\frac{\beta_{12}+\alpha_{12R}}{2}}=e^{k\frac{\theta_{12}}{2}}$ and simplify the scalars to obtain an equation in the terms of $i\epsilon$,

$$\mathbf{B}(1-e^{k\beta_{12}})+\mathbf{A}e^{k\beta_{12}}(1-e^{k\alpha_{12R}})=\mathbf{P}(1-e^{k\theta_{12}}). \tag{3.69}$$

Noting that $\mathbf{P}(1-e^{k\theta_{12}})=\mathbf{D}_2-e^{k\theta_{12}}\mathbf{D}_1$, see [58], and that we can write $\mathbf{B}=\mathbf{A}-\mathbf{W}$, $\mathbf{A}=\mathbf{D}_1-\mathbf{Z}$ and substitute $\theta_{12}=\alpha_{12}$, $\alpha_{12R}=\alpha_{12}-\beta_{12}$, we obtain

$$(\mathbf{D}_1-\mathbf{Z}-\mathbf{W})(1-e^{k\beta_{12}})+(\mathbf{D}_1-\mathbf{Z})e^{k\beta_{12}}(1-e^{k(\alpha_{12}-\beta_{12})})=\mathbf{D}_2-e^{k\theta_{12}}\mathbf{D}_1, \tag{3.70}$$

which simplifies to the standard form of Eq.(3.66),

$$W(e^{i\beta_{12}}-1)+Z(e^{i\alpha_{12}}-1)=\delta_{12}.$$

The dual quaternion synthesis, when specialized to planar quaternions, gives equations that contain an orientation part and a displacement part that is equivalent to that obtained using the standard form.

3.10.2 Dual quaternions and the screw triangle

Given three displacements $[T_1]$, $[T_2]$, $[T_3]$, the set of screw axes S_{12}, S_{23}, S_{13}, representing the relative displacements $[T_{12}]$, $[T_{23}]$, $[T_{13}]$ form a *screw triangle*, see Figure 3.7.

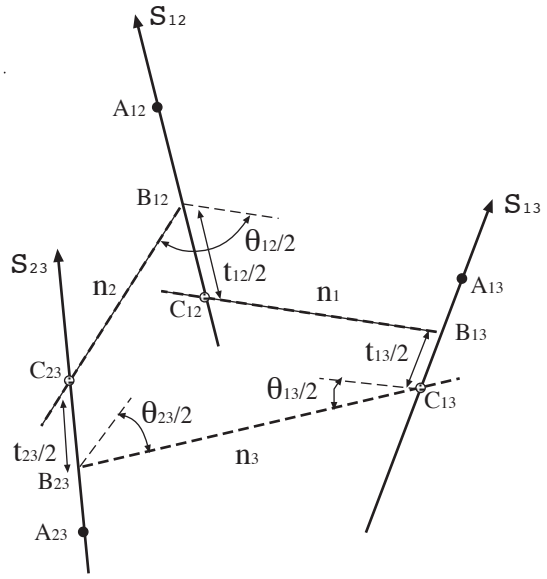

Figure 3.7: The screw triangle.

The angle θij and distance t_{ij} between common normals $\mathbf{n}_i$ and $\mathbf{n}_j$ along the axis S_{ij} correspond to half the rotation angle and translation of the relative displacement $[T_{ij}]$. This geometric property of the screw triangle was studied by Bottema and Roth [9] and applied to mechanism synthesis by Tsai [97]. He considered two of the screw axes as the axes of a dyad. The third screw is the displacement obtained from the composition of screw displacements of the dyad.

The equations that capture the geometry of the screw triangle can be used to solve for the joint axes of the dyad if the resultant displacement is known. The initial

equations,

$$\tan\frac{\theta_{12}}{2} = \frac{\mathbf{s}_{12}\cdot(\mathbf{s}_{23}\times\mathbf{s}_{13})}{(\mathbf{s}_{13}\times\mathbf{s}_{12})\cdot(\mathbf{s}_{12}\times\mathbf{s}_{23})},$$

$$\frac{t_{12}}{2} = \frac{\mathbf{s}_{12}-(\mathbf{s}_{12}\cdot\mathbf{s}_{23})\mathbf{s}_{23}}{(\mathbf{s}_{12}\times\mathbf{s}_{23})^2}\cdot(\mathbf{A}_{23}-\mathbf{A}_{12}) - \frac{\mathbf{s}_{12}-(\mathbf{s}_{12}\cdot\mathbf{s}_{13})\mathbf{s}_{13}}{(\mathbf{s}_{12}\times\mathbf{s}_{13})^2}\cdot(\mathbf{A}_{13}-\mathbf{A}_{12}), \quad (3.71)$$

are transformed in [9] into expressions that can be solved for the parameters of one axis as a function of the other two axes.

The dual quaternion product

Consider the dual quaternion expressions, $\hat{S}_{12}$, $\hat{S}_{23}$, of the relative displacements $[T_{12}]$, $[T_{23}]$ described above, and let $\hat{S}_{13}$ be the result of their composition. The dual quaternion product,

$$\begin{aligned}\hat{S}_{13} =& \hat{S}_{12}\hat{S}_{23} = (\cos\frac{\hat{\theta}_{12}}{2}\cos\frac{\hat{\theta}_{23}}{2} - \sin\frac{\hat{\theta}_{12}}{2}\cos\frac{\hat{\theta}_{23}}{2}\mathsf{S}_{12}\cdot\mathsf{S}_{23}) + \\ & (\cos\frac{\hat{\theta}_{12}}{2}\sin\frac{\hat{\theta}_{23}}{2}\mathsf{S}_{23} + \sin\frac{\hat{\theta}_{12}}{2}\cos\frac{\hat{\theta}_{23}}{2}\mathsf{S}_{12} + \sin\frac{\hat{\theta}_{12}}{2}\sin\frac{\hat{\theta}_{23}}{2}\mathsf{S}_{12}\times\mathsf{S}_{23}), \quad (3.72)\end{aligned}$$

can be manipulated to obtain expressions that allows us to solve directly for the screw axis S_{13} and the rotation and translation θ_{13}, d_{13}.

For obtaining the expression of the direction and rotation angle, we consider the real part of Eq. (3.72). Divide the vector part by the scalar to obtain

$$\tan\frac{\theta_{13}}{2}\mathbf{s}_{13} = \frac{\tan\frac{\theta_{12}}{2}\mathbf{s}_{12} + \tan\frac{\theta_{23}}{2}\mathbf{s}_{23} + \tan\frac{\theta_{12}}{2}\tan\frac{\theta_{23}}{2}\mathbf{s}_{12}\times\mathbf{s}_{23}}{1-\tan\frac{\theta_{12}}{2}\tan\frac{\theta_{23}}{2}\mathbf{s}_{12}\cdot\mathbf{s}_{23}}. \quad (3.73)$$

To solve for the rotation angle, we dot the expression in Eq.(3.73) with itself. This

yields

$$\tan\frac{\theta_{13}}{2}=\frac{\sqrt{\tan^2\frac{\theta_{12}}{2}+\tan^2\frac{\theta_{23}}{2}+2\tan\frac{\theta_{12}}{2}\tan\frac{\theta_{23}}{2}\mathbf{s}_{12}\cdot\mathbf{s}_{23}+\tan^2\frac{\theta_{12}}{2}\tan^2\frac{\theta_{23}}{2}(\mathbf{s}_{12}\times\mathbf{s}_{23})^2}}{1-\mathbf{s}_{12}\cdot\mathbf{s}_{23}\tan\frac{\theta_{12}}{2}\tan\frac{\theta_{23}}{2}}, \tag{3.74}$$

which is the same as the expression obtained in [9]. For the direction of the screw axis we divide by this value in Eq.(3.73),

$$\mathbf{s}_{13}=\frac{\tan\frac{\theta_{12}}{2}\mathbf{s}_{12}+\tan\frac{\theta_{23}}{2}\mathbf{s}_{23}+\tan\frac{\theta_{12}}{2}\tan\frac{\theta_{23}}{2}\mathbf{s}_{12}\times\mathbf{s}_{23}}{\sqrt{\tan^2\frac{\theta_{12}}{2}+\tan^2\frac{\theta_{23}}{2}+2\tan\frac{\theta_{12}}{2}\tan\frac{\theta_{23}}{2}\mathbf{s}_{12}\cdot\mathbf{s}_{23}+\tan^2\frac{\theta_{12}}{2}\tan^2\frac{\theta_{23}}{2}(\mathbf{s}_{12}\times\mathbf{s}_{23})^2}}. \tag{3.75}$$

We find expressions for a point in the screw axis $\mathbf{A}_{13}$ and for the displacement t_{13} from the dual part of the quaternion product. The cross product of $\mathbf{s}_{13}$ with the dual vector part of the screw axis yields the perpendicular point $\mathbf{A}_{13}$ in the line S_{13} from the origin,

$$\mathbf{A}_{13}=\frac{1}{\tan\frac{\theta_{13}}{2}}\mathbf{s}_{13}\times(\frac{t_{13}}{2}\mathbf{s}_{13}+\tan\frac{\theta_{13}}{2}\mathbf{s}^0_{13}). \tag{3.76}$$

We divide the dual vector part of the product, $\frac{t_{13}}{2}\cos\frac{\theta_{13}}{2}\mathbf{s}_{13}+\sin\frac{\theta_{13}}{2}\mathbf{s}^0_{13}$ by the real scalar $\cos\frac{\theta_{13}}{2}$. The dot product with the real vector $\mathbf{s}_{13}$ yields, if we call k the square root in the denominator of Eq.(3.75),

$$\begin{aligned}\frac{t_{13}}{2}=&\frac{1}{k}\big((\mathbf{A}_{12}-\mathbf{A}_{23})\cdot(\mathbf{s}_{12}\times\mathbf{s}_{23})\tan\frac{\theta_{12}}{2}\tan\frac{\theta_{23}}{2}+\\&\frac{t_{12}}{2}(\tan\frac{\theta_{12}}{2}+\tan\frac{\theta_{23}}{2}\mathbf{s}_{12}\cdot\mathbf{s}_{23})+\frac{t_{23}}{2}(\tan\frac{\theta_{23}}{2}+\tan\frac{\theta_{12}}{2}\mathbf{s}_{12}\cdot\mathbf{s}_{23})\big).\end{aligned} \tag{3.77}$$

The expressions in Eqs.(3.74, 3.75, 3.76, 3.77) correspond to those derived from the screw triangle by Bottema and Roth [9]. The equality of the direction equations had

been proved also in section 3.8.2 using a different approach. Hence the geometry of the screw triangle can be obtained by using the dual quaternion product of displacements.

3.11 Summary

In this chapter, we developed the basic theory of the finite-position dual quaternion synthesis of constrained robots. We identified the relation between the kinematics equations in 4×4 homogeneous matrix form and in dual quaternion form, and showed that the latter corresponds to successive screw displacements and the product of exponentials based on Lie group methods. We created the design equations by equating the dual quaternion kinematics equations to a set of task dual quaternions.

We were able to create a formula to count for the maximum number of positions that can be defined for each robot topology. In doing so, we identified special cases in which the robot is orientation-limited or translation-unconstrained, due to the semi-direct product structure of the group of spatial displacements.

The design equations obtained with this method are parameterized by the joint variables: assigning the whole range of values to the joint variables, we obtain the workspace of the robot. We devised a methodology to eliminate these non-structural variables that yields a set of algebraic equations that, in the simplest cases, were proved to be equivalent to the equations obtained stating the geometric constraints of the chain. The set of design equations were then solved using different methods, depending on their complexity.

The dual quaternion synthesis methodology was compared to planar synthesis using complex numbers and to the equivalent screw triangle methodology for spatial synthesis, and we were able to show that both methodologies can be derived from the dual quaternion synthesis.

Chapter 4

Synthesis for Subgroups of SE(3)

4.1 Introduction

The quaternion synthesis of spherical robots is a special case of the dual quaternion synthesis of spatial robots. A dual quaternion contains Hamilton's quaternion as its real part; real parts operate separately to represent the orientation of the general displacement. The solution of the quaternion synthesis is directly applicable in some cases of spatial movement, when the displacement of the robot is limited by the orientations.

Planar quaternion synthesis is developed to design the two general cases of planar constrained robots. In Section 3.10 we showed the specialization of dual quaternions into planar quaternions. In this Chapter, we compare the design equations obtained using planar quaternions to those created as geometric constraints.

A different set of special cases is studied in the last part of this Chapter. For these cases, the joints of the robot form a subgroup of the group of spatial displacements,

but the task positions need to be considered in its general form.

4.2 Quaternion Synthesis of Spherical Robots

Spherical robots are those whose workspace is contained on the surface of a sphere. They consist of revolute joints whose axes intersect at a point; the end-effector moves in a pure rotation about this point. The workspace of a spherical robot is a subset of the group of spatial rotations, the special orthogonal group SO(3).

Some work has been done in spherical robots with prismatic joints; however, this joint can be transformed into a revolute joint by making the axis of the revolute joint perpendicular to the circle described by the prismatic joint. There are two general cases of constrained spherical robots, the R and the RR robot.

4.2.1 Kinematics equations of constrained spherical robots

The kinematics equations are created using the same procedure described in Chapter 3. The rotation matrix kinematics equations,

$$[R_{0i}(\theta_1, \ldots, \theta_k)] = [R_1(\Delta\theta_1^i)] \ldots [R_k(\Delta\theta_k^i)]. \tag{4.1}$$

are transformed into quaternion form,

$$Q(\theta_1, \ldots, \theta_k) = S_1(\theta_1) \ldots S_k(\theta_k). \tag{4.2}$$

4.2.2 Design equations for constrained spherical robots

A discrete approximation of the desired workspace is given in the form of $n-1$ task rotations $R_i, i = 2 \dots, n$. As in the general case, we equate the kinematics equations to the task rotations to obtain design equations. The set of parameterized design equations is

$$\left\{ \begin{array}{c} q_x(\theta_1^i, \dots, \theta_k^i) \\ q_y(\theta_1^i, \dots, \theta_k^i) \\ q_z(\theta_1^i, \dots, \theta_k^i) \end{array} \right\} = \left\{ \begin{array}{c} r_x^i \\ r_y^i \\ r_z^i \end{array} \right\}, \quad i = 2, \dots, n.$$

$$\mathbf{s}_j \cdot \mathbf{s}_j = 1, \quad j = 1 \dots, k. \tag{4.3}$$

As noted before, only three of the components of the quaternion are independent. We can prove that the set of solutions generated by the four quaternion components is the same as that generated by three components of the quaternion considering the unit vector condition for the quaternion. To do this, we express the last design equation as a polynomial combination of the other three and the unit quaternion condition,

$$\begin{aligned}(q_w - r_w)(q_w + r_w) = &- (q_x + r_x)(q_x - r_x) - (q_y + r_y)(q_y - r_y) - (q_z + r_z)(q_z - r_z) + \\ &(q_x^2 + q_y^2 + q_z^2 + q_w^2 - 1)\end{aligned} \tag{4.4}$$

where the polynomial $(q_w + p_w)$ does not have any root in the set, except for the case $q_w = 0$, which is not of interest for us.

Next step is to substitute the unit quaternion condition $q_x^2 + q_y^2 + q_z^2 + q_w^2 = 1$ by the unit vector conditions for the rotation axes. We only need to prove it for the cases of

one and two rotation axes, but the proof can be generalized using induction.

For the case $k = 1$, we can see directly that

$$q_x^2 + q_y^2 + q_z^2 + q_w^2 = 1 \iff \sin^2\frac{\theta}{2}(s_x^2 + s_y^2 + s_z^2) + \cos^2\frac{\theta}{2} = 1 \iff s_x^2 + s_y^2 + s_z^2 = 1. \tag{4.5}$$

Because of this, the set of solutions generated by the original quaternion equality is the same as that generated by the set $< q_x - r_x, q_y - r_y, q_z - r_z, s_x^2 + s_y^2 + s_z^2 - 1 >$.

For the case $k = 2$, the squared components of the quaternion are

$$\begin{aligned} &q_x^2 + q_y^2 + q_z^2 + q_w^2 = (\sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}\mathbf{s}_1 + \cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}\mathbf{s}_2 + \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}\mathbf{s}_1 \times \mathbf{s}_2)^2 + \\ &(\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2} - \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}\mathbf{s}_1 \cdot \mathbf{s}_2)^2 = \sin^2\frac{\theta_1}{2}\cos^2\frac{\theta_2}{2}\mathbf{s}_1 \cdot \mathbf{s}_1 + \\ &\cos^2\frac{\theta_1}{2}\sin^2\frac{\theta_2}{2}\mathbf{s}_2 \cdot \mathbf{s}_2 + \sin^2\frac{\theta_1}{2}\sin^2\frac{\theta_2}{2}\mathbf{s}_1 \cdot \mathbf{s}_1\mathbf{s}_2 \cdot \mathbf{s}_2 + \cos^2\frac{\theta_1}{2}\cos^2\frac{\theta_2}{2} = \\ &(\sin^2\frac{\theta_1}{2}\mathbf{s}_1 \cdot \mathbf{s}_1 + \cos^2\frac{\theta_1}{2})(\sin^2\frac{\theta_2}{2}\mathbf{s}_2 \cdot \mathbf{s}_2 + \cos^2\frac{\theta_2}{2}) \end{aligned} \tag{4.6}$$

This last expression is equal to one if and only if $\mathbf{s}_1 \cdot \mathbf{s}_1 = 1$ and $\mathbf{s}_2 \cdot \mathbf{s}_2 = 1$. Again we can use the unit vector conditions for the axes instead of the unit condition for the quaternion.

4.2.3 Synthesis of a spherical R robot

The spherical R robot is a one-degree-of-freedom robot consisting of one revolute axis $\mathbf{g}$ arbitrarly oriented, that allows rotation of angle θ about it, see Figure 4.1.

The kinematics equations for a spherical R robot are defined by the angle about

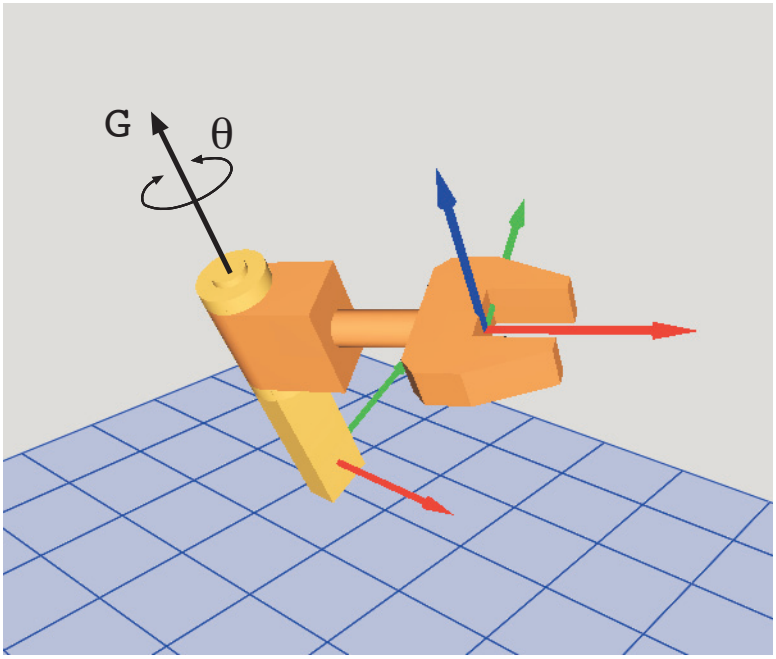

Figure 4.1: A robot consisting of a revolute joint

the fixed rotation axis **g**,

$$Q(\theta) = G(\theta) = \begin{Bmatrix} \sin\frac{\theta}{2} g_x \\ \sin\frac{\theta}{2} g_y \\ \sin\frac{\theta}{2} g_z \\ \cos\frac{\theta}{2} \end{Bmatrix}. \tag{4.7}$$

These are equated to the task orientations. The maximum number of orientations are given by Eq.(3.17). For a single revolute joint, the maximum number of orientations is two.

$$Q(\theta^i,) = R^i, \quad i = 1, 2. \tag{4.8}$$

The parameterized design equations are reduced by eliminating linearly the joint

variable θ. We construct the matrix

$$\begin{bmatrix} 0 & g_x \\ 0 & g_y \\ 0 & g_z \\ 1 & 0 \end{bmatrix} \begin{Bmatrix} \cos\frac{\theta}{2} \\ \sin\frac{\theta}{2} \end{Bmatrix} = \begin{Bmatrix} r_x \\ r_y \\ r_z \\ r_w \end{Bmatrix} \tag{4.9}$$

Every one of the three 2×2 linear system of rank 2 must give a solution,

$$\begin{bmatrix} 0 & g_x \\ 1 & 0 \end{bmatrix} \begin{Bmatrix} \cos\frac{\theta}{2} \\ \sin\frac{\theta}{2} \end{Bmatrix} = \begin{Bmatrix} r_x \\ r_w \end{Bmatrix}; \tag{4.10}$$

We obtain the set of solutions

$$\begin{aligned} \cos\frac{\theta}{2} &= r_w \\ \sin\frac{\theta}{2} &= \frac{r_i}{g_i}, \quad i = x, y, z, \end{aligned} \tag{4.11}$$

which states that the direction and angle of the revolute joint axis $\mathbf{g}$ need to be the same as the direction and angle of the task dual quaternion.

The reduced design equations are created by equating the values of $\sin\frac{\theta}{2}$,

$$\begin{aligned} g_y r_x - g_x r_y &= 0 \\ g_z r_x - g_x r_z &= 0 \end{aligned} \tag{4.12}$$

and together with the condition $|\mathbf{g}| = 1$, we can solve for the direction of the revolute joint $\mathbf{g}$. The design equations impose parallelism between the directions of the revolute joint and the task position, that is, $\mathbf{g} \times \mathbf{r} = 0$,

$$\begin{aligned} g_x &= \frac{r_x}{\sqrt{r_x^2 + r_y^2 + r_z^2}} \\ g_y &= \frac{r_y}{\sqrt{r_x^2 + r_y^2 + r_z^2}} \\ g_z &= \frac{r_z}{\sqrt{r_x^2 + r_y^2 + r_z^2}}. \end{aligned} \tag{4.13}$$

There exist two solutions, corresponding to opposite directions.

Hence, the two solutions for the spherical R robot are obtained by aligning the revolute axis with the axis of the relative rotation and then rotating by the relative rotation angle. The second solution is obtained by negating both axis and rotation angle.

4.2.4 Synthesis of a spherical RR robot

The spherical *RR* robot is a two-degree of freedom robot consisting of two revolute axes, the fixed axis **g** and the moving axis **w**, which are concurrent. The rotation about the fixed axis **g** is called θ and ϕ is the rotation about **w**, see Figure 4.2.

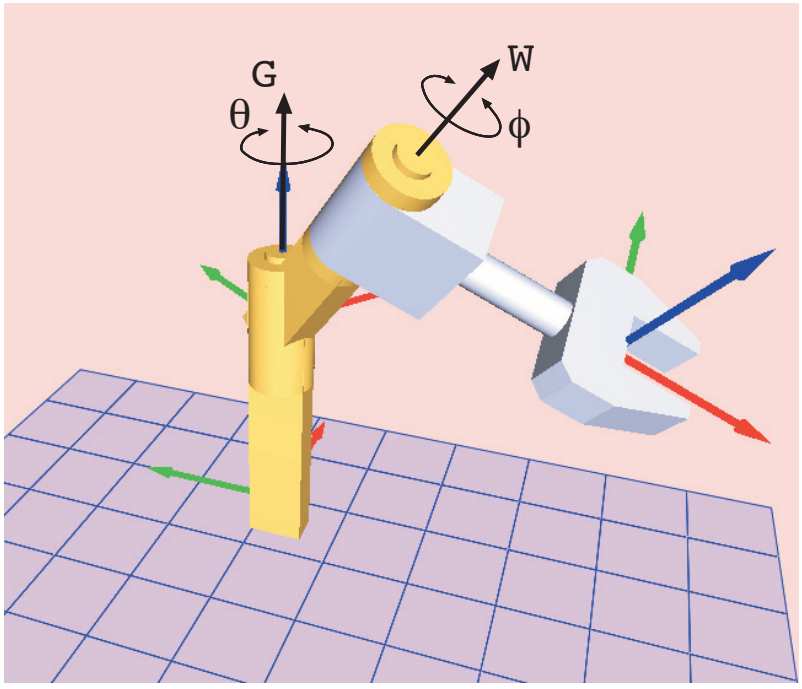

Figure 4.2: A spherical RR robot.

The quaternion representation for the relative rotations of the chain is given by

$$Q_{RR} = G(\theta)W(\phi). \tag{4.14}$$

When applying the quaternion product we obtain the expression $Q_{RR} = q^0 + \mathbf{q}$, where the point q^0 and the vector $\mathbf{q}$ are

$$\begin{aligned} q^0 &= \cos\frac{\theta}{2}\cos\frac{\phi}{2} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}\mathbf{g}\cdot\mathbf{w}, \\ \mathbf{q} &= \sin\frac{\theta}{2}\cos\frac{\phi}{2}\mathbf{g} + \cos\frac{\theta}{2}\sin\frac{\phi}{2}\mathbf{w} + \sin\frac{\theta}{2}\sin\frac{\phi}{2}(\mathbf{g}\times\mathbf{w}). \end{aligned} \tag{4.15}$$

The componentwise expansion leads to a set of equations in the components of the fixed rotation axis, $\mathbf{g} = (g_x, g_y, g_z)$, and the moving rotation axis $\mathbf{w} = (w_x, w_y, w_z)$. If we write the point as the fourth component of the direction vector we have

$$Q_{RR} = \left\{ \begin{array}{c} \cos\frac{\theta}{2}\sin\frac{\phi}{2}\mathbf{w} + \sin\frac{\theta}{2}\cos\frac{\phi}{2}\mathbf{g} + \sin\frac{\theta}{2}\sin\frac{\phi}{2}\mathbf{g}\times\mathbf{w} \\ \cos\frac{\theta}{2}\cos\frac{\phi}{2} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}\mathbf{g}\cdot\mathbf{w} \end{array} \right\}. \tag{4.16}$$

The number of rotations needed to obtain finite number of solutions is calculated, using the formula in Eq.(3.17), to be a maximum of $n_R = 5$ rotations. We create the design equations,

$$Q_{RR}(\theta^i, \phi^i) - R^i = \vec{0}, \quad i = 2, \ldots 5, \tag{4.17}$$

to obtain the set of parameterized design equations.

To eliminate the joint parameters θ and ϕ, we use the procedure developed in

Section 3.8.2. Construct the linear system

$$\begin{bmatrix} \mathbf{g} & \mathbf{w} & \mathbf{g}\times\mathbf{w} & \vec{0} \\ 0 & 0 & -\mathbf{g}\cdot\mathbf{w} & 1 \end{bmatrix} \begin{Bmatrix} \sin\frac{\theta}{2}\cos\frac{\phi}{2} \\ \cos\frac{\theta}{2}\sin\frac{\phi}{2} \\ \sin\frac{\theta}{2}\sin\frac{\phi}{2} \\ \cos\frac{\theta}{2}\cos\frac{\phi}{2} \end{Bmatrix} = \begin{Bmatrix} \mathbf{p} \\ p_w \end{Bmatrix} \tag{4.18}$$

in which vectors form the columns of the matrix. The system has a solution when the matrix is invertible. The determinant of the matrix is $(\mathbf{g}\times\mathbf{w})\cdot(\mathbf{g}\times\mathbf{w})$, which is zero only for the degenerate case when both directions are parallel.

We invert the matrix (see Eq.(3.40)) and calculate the inverse kinematics for the angles θ and ϕ. The reduced equation is given in Eq. (3.42) by the relation among sines and cosines of the joint variables.

We could also use other trigonometric relations, but this one has the virtue of not adding extraneous roots, and even eliminating the norm of the vector $\mathbf{g}\times\mathbf{w}$. When applying this relation, we obtain the expression in Eq.(3.45), which expands to

$$\begin{aligned} \mathcal{R}: \quad & g_x((-r_y^2-r_z^2)w_x + (r_x r_y - r_w r_z)w_y + (r_w r_y + r_x r_z)w_z) + \\ & g_y((r_x r_y + r_w r_z)w_x + (-r_x^2-r_z^2)w_y + (r_y r_z - r_w r_x)w_z) + \\ & g_z((r_x r_z - r_w r_y)w_x + (r_w r_x + r_y r_z)w_y + (-r_x^2-r_y^2)w_z) = 0 \end{aligned} \tag{4.19}$$

Notice that this equation is bilinear and homogeneous in the components of $\mathbf{g}$ and $\mathbf{w}$. As it has been proved in Section 3.8.2, this expression is equivalent to the matrix geometric constraint imposing constant angle between $\mathbf{g}$ and $\mathbf{w}$, and also equal to the direction equation of the equivalent screw triangle formulation.

The set of reduced design equations consists of one equation like Eq.(4.19) per relative rotation plus the two unit vector constraints,

$$\begin{aligned} &\mathcal{R}^i, \quad i = 1, \ldots, 4 \\ &g_x^2 + g_y^2 + g_z^2 = 1, \\ &w_x^2 + w_y^2 + w_z^2 = 1. \end{aligned} \tag{4.20}$$

We solve the set of Eq.(4.20) for the 6 unknowns corresponding to the two joint axes **g** and **w**. For this case, there exists an algebraic closed solution, consisting of a two-step matrix elimination, see for instance [58]. The result is a set of linear equations plus a univariate six-degree polynomial; the spherical RR robot synthesis has at most six complex solutions.

4.3 Planar Quaternion Synthesis of Planar Robots

Planar robots move in parallel planes [58], allowing both translation and rotation of the end-effector within the plane. They consist of revolute joints whose axes are perpendicular to this plane, and prismatic joints whose directions belong to the plane. The workspace of a planar robot is a subset of the special euclidean group of planar displacements, SE(2). Planar quaternions have been used for the synthesis and analysis of planar robots, see [16, 42, 61],

A constrained serial chain has at most two degrees of freedom. There are three basic constrained serial chains, the single revolute joint R, the planar RP (or PR) and the planar RR robots. Chains consisting of one or two prismatic joints cannot

reach a general planar displacement.

4.3.1 Kinematics equations of constrained planar robots

We transform the matrix kinematics equations, written as 3×3 planar displacement matrices,

$$[T_{0i}(\theta_1, \ldots, \theta_k)] = [T_1(\Delta\theta_1^i)] \ldots [T_k(\Delta\theta_k^i)]. \tag{4.21}$$

into planar quaternion form, following the notation of Chapter 2 and Chapter 3,

$$Q(\theta_1, \ldots, \theta_k) = p_1(\theta_1) \ldots p_k(\theta_k). \tag{4.22}$$

where θ_i designates a generic variable that can be either rotation or translation, and p_i are the poles of the relative displacements for each link of the robot, which coincide with the joints of the robot.

4.3.2 Design equations for constrained planar robots

A discrete approximation of the desired workspace is given in the form of $n-1$ task planar displacements $P_i, i = 2 \ldots, n$. As in the general case, we equate the kinematics equations to the task rotations to obtain the set of parameterized design equations,

$$\begin{Bmatrix} q_x(\theta_1^i, \ldots, \theta_k^i) \\ q_y(\theta_1^i, \ldots, \theta_k^i) \\ q_z(\theta_1^i, \ldots, \theta_k^i) \\ q_w(\theta_1^i, \ldots, \theta_k^i) \end{Bmatrix} = \begin{Bmatrix} p_x^i \\ p_y^i \\ p_z^i \\ p_w^i \end{Bmatrix}, \quad i = 2, \ldots, n. \tag{4.23}$$

The maximum number of planar displacement are computed from a counting formula derived in a similar fashion as Eq.(3.16). Every revolute joint is a point on the

plane, given by two coordinates. A prismatic joint is a direction on the plane, consisting of two coordinates with a unit vector condition. Three of the four components of the planar quaternion are independent. For r revolute joints and t prismatic joints, we have

$$n_P = \frac{3+r}{3-r-t}. \tag{4.24}$$

4.3.3 Synthesis of a planar R robot

The planar R robot is a one-degree-of-freedom robot consisting of one revolute axis $\mathbf{g}$ that allows rotation of angle θ about it. The kinematics equations,

$$G(\theta) = \begin{Bmatrix} \sin\frac{\theta}{2} g_y \\ -\sin\frac{\theta}{2} g_x \\ \sin\frac{\theta}{2} \\ \cos\frac{\theta}{2} \end{Bmatrix}. \tag{4.25}$$

are equated to the task planar displacements. In this case, the maximum number of task positions is two.

$$G(\theta^i,) = P^i, \quad i = 1, 2. \tag{4.26}$$

The general planar displacement P is expressed as a planar quaternion in Eq.(3.61). In this case we can solve directly from the design equations. We obtain

$$\begin{aligned} g_x &= \frac{-p_y}{p_z} = \frac{d_x}{2}\sin\frac{\theta}{2} - \frac{d_y}{2}\cos\frac{\theta}{2} \\ g_y &= \frac{-p_y}{p_z} = \frac{d_x}{2}\cos\frac{\theta}{2} + \frac{d_y}{2}\sin\frac{\theta}{2} \end{aligned} \tag{4.27}$$

This is, as expected, the expression of the pole for the relative transformation, see Eq.(3.60).

4.3.4 Synthesis of a planar RP robot

The planar RP robot is a two-degree of freedom robot consisting of one revolute axes, the fixed axis $\mathbf{g}$, and one prismatic moving axis $\mathbf{w}$. Notice that $\mathbf{g}$ is a point of the projective plane, while $\mathbf{w}$ is a direction. The rotation about the fixed axis $\mathbf{g}$ is called θ and d is the translation along $\mathbf{w}$, see Figure 4.3.

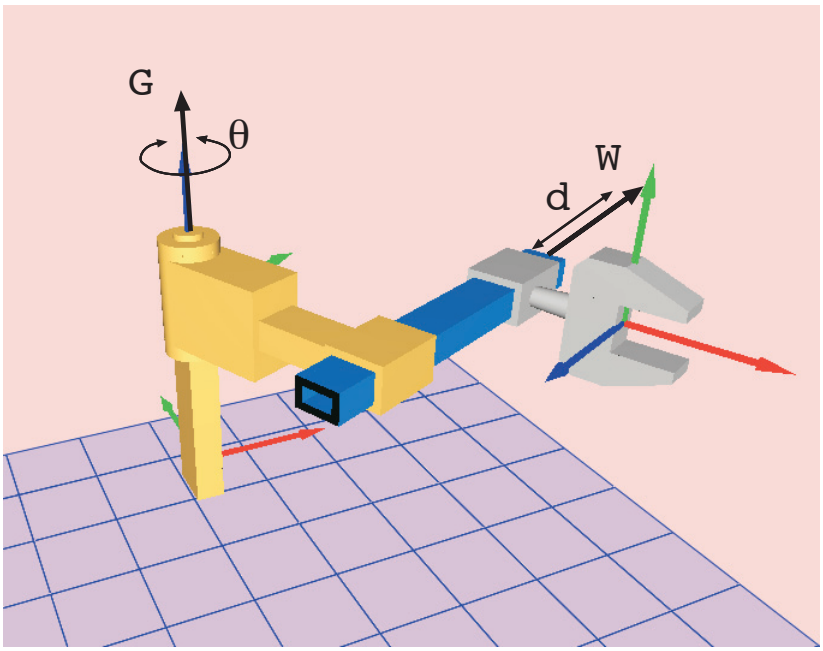

Figure 4.3: A planar RP robot.

The planar quaternion representation for the relative displacements of the chain,

$$Q_{RP} = G(\theta)W(d), \tag{4.28}$$

expands to

$$Q_{RP} = \begin{Bmatrix} \frac{d}{2}\cos\frac{\theta}{2}w_x + \sin\frac{\theta}{2}(g_y - \frac{d}{2}w_y) \\ \frac{d}{2}\cos\frac{\theta}{2}w_y + \sin\frac{\theta}{2}(-g_x + \frac{d}{2}w_x) \\ \sin\frac{\theta}{2} \\ \cos\frac{\theta}{2} \end{Bmatrix}. \tag{4.29}$$

We can solve the planar RP robot, using the formula in Eq.(4.24), for a maximum of $n_P = 4$ displacements.

$$Q_{PR}(\theta^i, d^i) - P^i = \vec{0}, \quad i = 2, \ldots 4, \tag{4.30}$$

To eliminate the joint variables θ and d, construct the linear system

$$\begin{bmatrix} -\frac{w_y}{2} & \frac{w_x}{2} & g_y & 0 \\ \frac{w_x}{2} & \frac{w_y}{2} & -g_x & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} d\sin\frac{\theta}{2} \\ d\cos\frac{\theta}{2} \\ \sin\frac{\theta}{2} \\ \cos\frac{\theta}{2} \end{Bmatrix} = \begin{Bmatrix} p_x \\ p_y \\ p_z \\ p_w \end{Bmatrix} \tag{4.31}$$

and solve to obtain the inverse kinematics for the robot,

$$\begin{Bmatrix} d\sin\frac{\theta}{2} \\ d\cos\frac{\theta}{2} \\ \sin\frac{\theta}{2} \\ \cos\frac{\theta}{2} \end{Bmatrix} = \begin{Bmatrix} 2(p_y w_x - p_x w_y + p_z(g_x w_x + g_y w_y)) \\ 2(p_x w_x + p_y w_y + p_z(g_x w_y - g_y w_x)) \\ p_z \\ p_w \end{Bmatrix} \tag{4.32}$$

The reduced design equation is obtained as the subspace of solutions that relates the joint variables,

$$\mathcal{R}: \quad \frac{2(p_y w_x - p_x w_y + p_z(g_x w_x + g_y w_y))}{p_z} - \frac{2(p_x w_x + p_y w_y + p_z(g_x w_y - g_y w_x))}{p_w} = 0 \tag{4.33}$$

This equation for each planar task position, plus the unit vector condition for $\mathbf{w}$, forms the set of reduced design equations,

$$\{\mathcal{R}\}^i, \quad i = 1, 2, 3$$

$$w_x^2 + w_y^2 = 1. \tag{4.34}$$

These equations are solved for $\mathbf{g}$ and $\mathbf{w}$ to obtain a unique solution for the RP robot.

4.3.5 Synthesis of a planar RR robot

The planar RR robot is a two-degree-of-freedom robot consisting of two revolute axes, the fixed axis $\mathbf{g}$ and the moving axis $\mathbf{w}$, which are points on the plane . The rotation about the fixed axis $\mathbf{g}$ is called θ and ϕ is the rotation about $\mathbf{w}$, see Figure 4.4.

The planar quaternion representation for the relative displacements of the chain,

$$Q_{RR} = G(\theta)W(\phi) = \begin{Bmatrix} \sin\frac{\theta}{2}\cos\frac{\phi}{2}g_y + \cos\frac{\theta}{2}\sin\frac{\phi}{2}w_y - \sin\frac{\theta}{2}\sin\frac{\phi}{2}(g_x - w_x) \\ -\sin\frac{\theta}{2}\cos\frac{\phi}{2}g_x - \cos\frac{\theta}{2}\sin\frac{\phi}{2}w_x - \sin\frac{\theta}{2}\sin\frac{\phi}{2}(g_y - w_y) \\ \sin\frac{\theta}{2}\cos\frac{\phi}{2} + \cos\frac{\theta}{2}\sin\frac{\phi}{2} \\ \cos\frac{\theta}{2}\cos\frac{\phi}{2} - \sin\frac{\theta}{2}\sin\frac{\phi}{2} \end{Bmatrix}, \tag{4.35}$$

is equated to a maximum of $n_P = 5$ displacements,

$$Q_{RR}(\theta^i, \phi^i) - P^i = \vec{0}, \quad i = 2, \ldots 5, \tag{4.36}$$

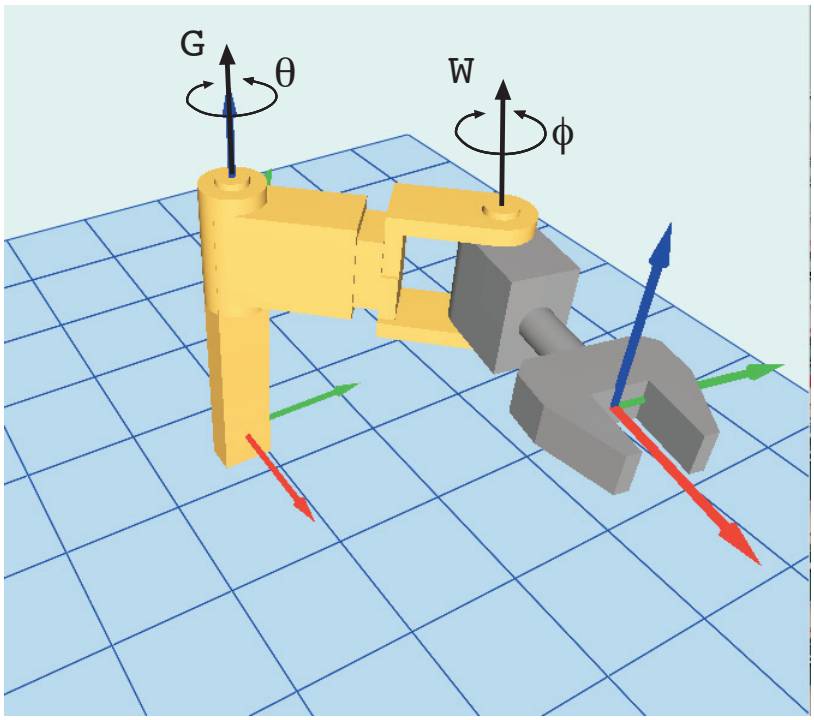

Figure 4.4: A planar RR robot.

Solve linearly for the joint variables θ and ϕ,

$$\begin{bmatrix} g_y & w_y & w_x - g_y & 0 \\ -g_x & -w_x & w_y - g_y & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} \begin{Bmatrix} \sin\frac{\theta}{2}\cos\frac{\phi}{2} \\ \cos\frac{\theta}{2}\sin\frac{\phi}{2} \\ \sin\frac{\theta}{2}\sin\frac{\phi}{2} \\ \cos\frac{\theta}{2}\cos\frac{\phi}{2} \end{Bmatrix} = \begin{Bmatrix} p_x \\ p_y \\ p_z \\ p_w \end{Bmatrix} \tag{4.37}$$

to obtain the inverse kinematics for the robot. The reduced design equation is obtained as the subspace of solutions that relates the joint variables, and that is the same as the one defined in Eq.(3.42). Eliminating the external roots, we obtain

$$\begin{aligned} \mathcal{R}: \quad & (-p_w p_x - p_y p_z)w_x + (-p_w p_y + p_x p_z)w_y + g_x(p_w p_x - p_y p_z - p_z^2 w_x - p_w p_z w_y) + \\ & g_y(p_w p_y + p_x p_z + p_w p_z w_x - p_z^2 w_y) - p_x^2 - p_y^2 = 0 \end{aligned} \tag{4.38}$$

The set of reduced design equations contains the four equations obtained evaluating

$\mathcal{R}$ at each of the task positions. We solve them for $\mathbf{g}$ and $\mathbf{w}$ to obtain four solutions for the RR robot.

4.4 Dual Quaternion Synthesis with Spherical Joints

A spherical joint allows rotation about an arbitrarly orientable axis passing by a point. Three perpendicular revolute axes intersecting at a point as in Figure 4.5 produce the same displacement, that is, an arbitrary rotation from the given point. The set of displacements produced by the spherical joint forms the subgroup S_o of spherical rotations about a point o. The subgroup has dimension 3, and three more parameters are needed to define the point.

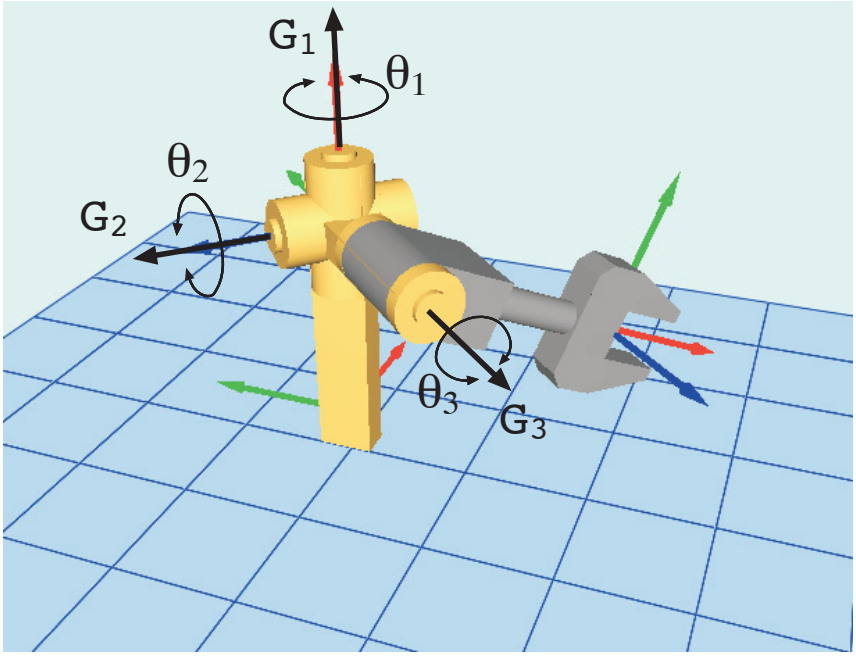

Figure 4.5: The spherical joint as three intersecting revolute joints.

Three perpendicular axes intersecting at a point are defined by twelve parameters

plus three joint variables minus six equations constraining the joint axes; this is a total of nine parameters. These need to be reduced to the six parameters defining the subgroup S_o.

4.4.1 Three perpendicular revolute joints

Let G_1, G_2, G_3 be three revolute joint axes such that $\mathsf{G}_1 \cdot \mathsf{G}_2 = 0$, $\mathsf{G}_1 \cdot \mathsf{G}_3 = 0$ and $\mathsf{G}_2 \cdot \mathsf{G}_3 = 0$. These six conditions imply that the directions are mutually orthogonal and that each two lines intersect; the intersection point is common to the three lines.

Each of the joint axes has the dual quaternion expression of Eq.(2.37). We create the spherical joint as the dual quaternion product of the individual axes,

$$\hat{S}(\theta_1, \theta_2, \theta_3) = \hat{G}_1(\theta_1)\hat{G}_2(\theta_2)\hat{G}_3(\theta_3). \tag{4.39}$$

When expanded, we obtain

$$\hat{S}(\theta_1, \theta_2, \theta_3) = \begin{Bmatrix} \alpha_1\mathbf{g}_1 + \alpha_2\mathbf{g}_2 + \alpha_3\mathbf{g}_3 \\ \alpha_4 \end{Bmatrix} + \epsilon \begin{Bmatrix} \alpha_1\mathbf{g}_1^0 + \alpha_2\mathbf{g}_2^0 + \alpha_3\mathbf{g}_3^0 \\ 0 \end{Bmatrix}, \tag{4.40}$$

where the α_i appear as combinations of the joint variables,

$$\begin{aligned}
\alpha_1 &= \sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}\cos\frac{\theta_3}{2} + \cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}\sin\frac{\theta_3}{2}, \\
\alpha_2 &= \cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}\cos\frac{\theta_3}{2} - \sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}\sin\frac{\theta_3}{2}, \\
\alpha_3 &= \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}\cos\frac{\theta_3}{2} + \cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2}\sin\frac{\theta_3}{2}, \\
\alpha_4 &= \cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2}\cos\frac{\theta_3}{2} - \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}\sin\frac{\theta_3}{2}.
\end{aligned} \tag{4.41}$$

When equating Eq.(4.39) to a goal displacement $\hat{P}$,

$$\hat{S}(\theta_1, \theta_2, \theta_3) = \hat{P}, \tag{4.42}$$

we can solve linearly for the combinations of joint variables in the α_i factors using the real part of the equation,

$$\begin{bmatrix} \mathbf{g}_1 & \mathbf{g}_2 & \mathbf{g}_3 & \vec{0} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{Bmatrix} = \begin{Bmatrix} \mathbf{p} \\ p_w \end{Bmatrix}. \tag{4.43}$$

The matrix is orthogonal and the values for the joint angles,

$$\alpha_1 = \mathbf{g}_1 \cdot \mathbf{p}, \quad \alpha_2 = \mathbf{g}_2 \cdot \mathbf{p}, \quad \alpha_3 = \mathbf{g}_3 \cdot \mathbf{p}, \quad \alpha_4 = p_w, \tag{4.44}$$

are used to define the reduced design equation given by the relation among the α_i factors,

$$\mathcal{R}: \quad (\mathbf{g}_1 \cdot \mathbf{p})^2 + (\mathbf{g}_2 \cdot \mathbf{p})^2 + (\mathbf{g}_3 \cdot \mathbf{p})^2 + p_w^2 = 1. \tag{4.45}$$

If we notice that $1 - p_w^2 = \mathbf{p} \cdot \mathbf{p}$, this expression can be written as

$$\mathcal{R}': \quad (\mathbf{g}_1 \cdot \mathbf{p})\mathbf{g}_1 + (\mathbf{g}_2 \cdot \mathbf{p})\mathbf{g}_2 + (\mathbf{g}_3 \cdot \mathbf{p})\mathbf{g}_3 = \mathbf{p}. \tag{4.46}$$

which simply states that the sum of the projections of $\mathbf{p}$ on the three joint directions is equal to $\mathbf{p}$. Notice that this statement is true *for any three perpendicular directions*; hence, there is no condition in the choice of the directions $\mathbf{g}_1$, $\mathbf{g}_2$, $\mathbf{g}_3$, and any three orthogonal directions can be used to obtain the rotation of the spherical joint.

We now substitute the expressions of the joint variables in the dual part of Eq.(4.42). If we express the dual part of each joint axis as $\mathbf{g}_i^0 = \mathbf{c} \times \mathbf{g}_i$, where $\mathbf{c}$ is the common intersection point, the dual part of the equations becomes

$$\mathcal{M}: \quad (\mathbf{g}_1 \cdot \mathbf{p})\mathbf{c} \times \mathbf{g}_1 + (\mathbf{g}_2 \cdot \mathbf{p})\mathbf{c} \times \mathbf{g}_2 + (\mathbf{g}_3 \cdot \mathbf{p})\mathbf{c} \times \mathbf{g}_3 = \mathbf{p}^0, \tag{4.47}$$

and this is equal to

$$\mathcal{M}' : \quad \mathbf{c} \times ((\mathbf{g}_1 \cdot \mathbf{p})\mathbf{g}_1 + (\mathbf{g}_2 \cdot \mathbf{p})\mathbf{g}_2 + (\mathbf{g}_3 \cdot \mathbf{p})\mathbf{g}_3) = \mathbf{p}^0. \tag{4.48}$$

Observe that the expression in parenthesis is the left hand side of Eq.(4.46); if we use this value, we obtain

$$\mathcal{M}'' : \quad \mathbf{c} \times \mathbf{p} = \mathbf{p}^0 \tag{4.49}$$

The reduced design equation $\mathcal{M}''$ shows that, for the translations, only the intersection point $\mathbf{c}$ is the design parameters, and the directions of the revolute joint axes can be chosen arbitrarily as long as they are orthogonal.

The final parameters that define the spherical joint are hence the intersection point $\mathbf{c}$ plus the three joint variables θ_i, the same parameters needed to define the subgroup S_o. These results need to be taken into consideration when designing robots with spherical joints.

4.4.2 An orientable axis

Consider the spherical joint as a rotation about an arbitrary axis that can be re-oriented in a general way. Let S_1 be the screw axis defined by the motion of the spherical joint to reach position 1, and S_j be the screw axis to reach position j. The relative screw S_{1j} passes through the same point, the center of the spherical joint $\mathbf{c}$, see Figure 4.6.

Focusing only in the direction of the screw axis, if we construct the kinematics equations as matrix displacements along the chain and we use the product of three

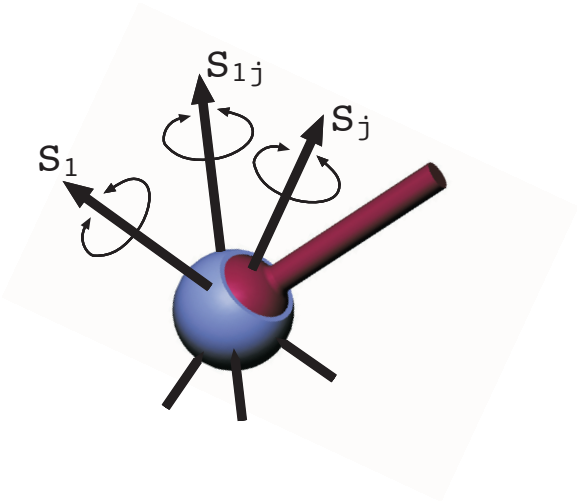

Figure 4.6: The spherical joint as an orientable screw axis.

Euler angles to define the direction of rotation,

$$[S_j] = [X(\alpha_j, 0)][Y(-\beta_j, 0)][Z(\theta_j, 0)] \tag{4.50}$$

and we compute the relative displacement $[S_{1j}]$ with respect to the reference configuration $[S_1]$, we can find the axis of the relative displacement, for instance using Cayley's formula,

$$\mathbf{s}_{1j} = \frac{1}{\sqrt{3 - \cos\alpha_j \cos\beta_j - \cos\theta_j(\cos\alpha_j + \cos\beta_j) - \sin\alpha_j \sin\beta_j \sin\theta_j}} \begin{Bmatrix} 2(\sin\frac{\alpha_j}{2}\cos\frac{\beta_j}{2}\cos\frac{\theta_j}{2} - \cos\frac{\alpha_j}{2}\sin\frac{\beta_j}{2}\sin\frac{\theta_j}{2}) \\ 2(\sin\frac{\alpha_j}{2}\cos\frac{\beta_j}{2}\sin\frac{\theta_j}{2} + \cos\frac{\alpha_j}{2}\sin\frac{\beta_j}{2}\cos\frac{\theta_j}{2}) \\ 2(\cos\frac{\alpha_j}{2}\cos\frac{\beta_j}{2}\sin\frac{\theta_j}{2} - \sin\frac{\alpha_j}{2}\sin\frac{\beta_j}{2}\cos\frac{\theta_j}{2}) \end{Bmatrix}. \tag{4.51}$$

which is presented here making the angles to reach the first position equal to zero.

This same result can be obtained from Eq.(4.39) if we make the revolute joints $\mathbf{g}_1 = (1, 0, 0)$, $\mathbf{g}_2 = (0, -1, 0)$ and $\mathbf{g}_3 = (0, 0, 1)$, and $\theta_1 = \alpha$, $\theta_2 = -\beta$, $\theta_3 = \theta$. The result of expressing the S-joint as an orientable axis is a particular case of the case

presented above, in which we specify the rotation axes to be the coordinate axes for the S joint at the first position.

4.5 Dual Quaternion Synthesis for Two Consecutive Prismatic Joints

When a serial robot is to be designed with two consecutive prismatic joints as in Figure 4.7, these can be made to be coplanar, as the location of a prismatic joint is not a design parameter. The set of displacements produced by the two prismatic joints forms the subgroup T_P of planar translations on a plane P. The subgroup has dimension 2, and two more parameters are needed to define the direction normal to the plane; for synthesis purposes, the location of the plane is again arbitrary.

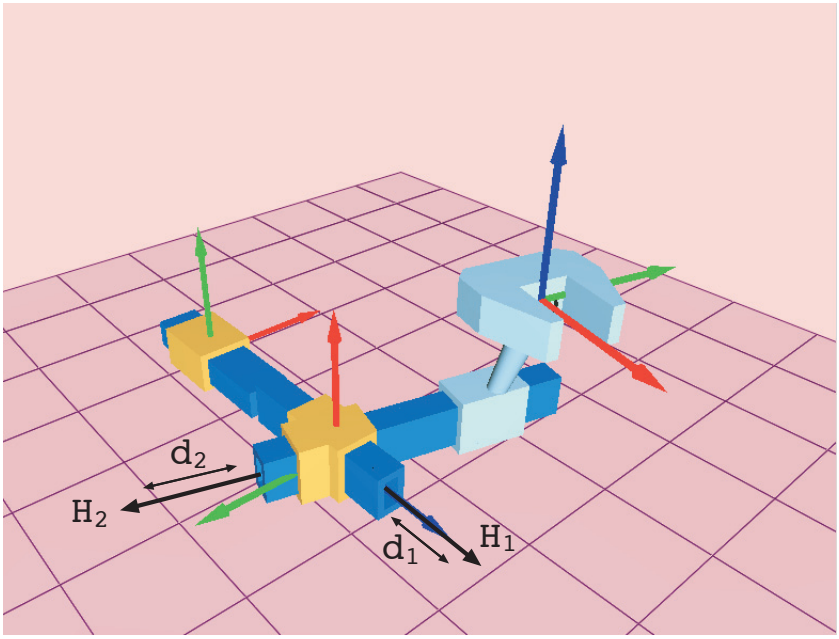

Figure 4.7: Two consecutive prismatic joints.

Let $\mathbf{h}_1$ and $\mathbf{h}_2$ be the directions of the two prismatic joints. Each of the joint axes has the dual quaternion expression of Eq.(2.38). We create the displacements of the two joints as the dual quaternion product,

$$\hat{H}(d_1, d_2) = \hat{H}_1(d_1)\hat{H}_2(d_2). \tag{4.52}$$

When expanded, the dual part yields

$$\hat{H}^0(d_1, d_2) = \begin{Bmatrix} \frac{d_1}{2}\mathbf{h}_1 + \frac{d_2}{2}\mathbf{h}_2 \\ 0 \end{Bmatrix}. \tag{4.53}$$

We solve linearly for the joint variables d_1 and d_2 in the design equations,

$$\begin{bmatrix} \frac{1}{2}\mathbf{h}_1 & \frac{1}{2}\mathbf{h}_2 \end{bmatrix} \begin{Bmatrix} d_1 \\ d_2 \\ 1 \end{Bmatrix} = \mathbf{p}^0. \tag{4.54}$$

For the system to have a solution, the determinant of the augmented matrix must be zero; this condition creates the reduced design equation

$$\mathcal{M}: \quad (\mathbf{h}_1 \times \mathbf{h}_2) \cdot \mathbf{p}^0 = 0. \tag{4.55}$$

The parameters of the prismatic joints always appear as a cross product, and we can substitute it by the common normal, $\mathbf{h}_1 \times \mathbf{h}_2 = \mathbf{n}$. Within the plane defined by $\mathbf{n}$, any two independent directions can be used to define the joint axes.

The design parameters for two consecutive prismatic joints are the two slides and the vector defining the normal direction to $\mathbf{h}_1$ and $\mathbf{h}_2$. It coincides with the parameters needed to define the subgroup T_P.

This result needs to be considered in the design process. Notice that two prismatic joints separated by a revolute joint do not create a fixed plane and hence this simplification is not applicable.

4.6 Summary

In this chapter, we apply the methodology developed in Chapter 3 to robots whose geometry creates special cases. In these cases, the movement of the robot occurs within a subgroup of the group of spatial displacements.

We study synthesis of planar constrained robots using planar quaternions, and synthesis of spherical robots using quaternions. Two cases that can appear within a spatial robot are also studied using dual quaternions: the spherical joint and the case of two consecutive prismatic joints.

Chapter 5

Application to the Synthesis of Constrained Robots

5.1 Overview

In this chapter we apply the methodology developed in Chapter 3 to a series of constrained robotic systems. All of them are solved as serial chains; in some cases, the solutions can be assembled to create parallel robots.

The chapter is organized in different sections for robots with two, three, four and five degrees of freedom. In each section, all possible cases that have been solved are described; particular cases of interest that are created by adding extra constraints on the axes are also presented and solved separately. For each set of revolute and prismatic axes, only one possible permutation is presented (for instance, the RPR only and not the PRR, RRP), when the counting, solution method and number of roots are equivalent. Cases in which the robot has three or more prismatic joints are

not presented, because, as explained in 3, they are not constrained in the translations.

For each robot, we introduce the topology and a brief summary of previous synthesis results, construct the design equations and solve them either in parameterized form or performing the reduction. In those cases for which a closed algebraic solution has been found, this is also included. An example for each robot is presented.

This case-by-case presentation illustrates the generality of the method but also the different strategies that can be applied to the design equations in order to make them easier to solve or to obtain additional information about the nature of the problem.

5.2 Synthesis of Two-degree-of-freedom Robots

The simplest cases of robots that can reach at least one complete relative transformation have two degrees of freedom. Considering revolute and prismatic joints, there are only two cases to solve for in this section: the RP robot and the RR robot.

5.2.1 Spatial RP robot

The spatial RP robot is a two-degree-of-freedom robot consisting of one revolute axis and one prismatic axis. Consider the fixed revolute axis $\mathsf{G} = \mathbf{g} + \epsilon\mathbf{g}^0 = (g_x, g_y, g_z) + \epsilon(g_x^0, g_y^0, g_z^0)$ and the moving prismatic axis H, which can be in an arbitrary position one respect to the other (but notice that if they are perpendicular, we obtain a planar motion). The rotation about the fixed axis G is called θ and it is followed by the translation d along $\mathsf{H} = \mathbf{h} + \epsilon\mathbf{h}^0 = (h_x, h_y, h_z) + \epsilon(h_x^0, h_y^0, h_z^0)$, see Figure 5.1.

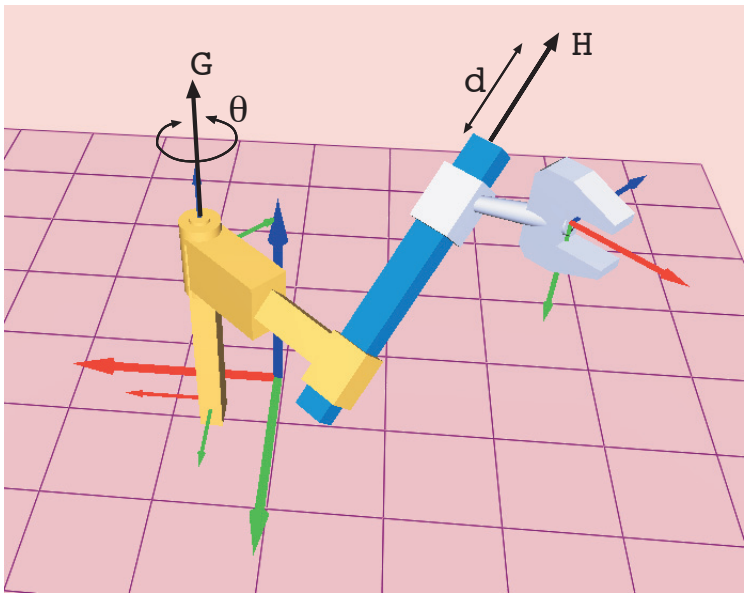

Figure 5.1: The spatial RP robot

A brief history

The study of the spatial RP chain, and in particular of the C chain, can be related to the first results of spatial kinematics if we identify the C chain with the screw axis of a relative displacement, see Chapter 1 for more details.

The design equations

The dual quaternion representation for the relative displacements of the chain is given by

$$\hat{Q}_{RP} = \hat{G}(\theta, 0)\hat{H}(0, d), \tag{5.1}$$

while the only possible permutation of this topology gives

$$\hat{Q}_{PR} = \hat{H}(0, d)\hat{G}(\theta, 0). \tag{5.2}$$

These equations expand to $\hat{Q}_{RP} = Q^0 + \mathsf{Q}$, where the dual point Q^0 and the dual vector Q are

$$\begin{aligned} Q^0_{RP} &= \cos\frac{\theta}{2} - \epsilon\frac{d}{2}\sin\frac{\theta}{2}\mathsf{G}\cdot\mathsf{H}, \\ \mathsf{Q}_{RP} &= \sin\frac{\theta}{2}\mathsf{G} + \epsilon\frac{d}{2}\cos\frac{\theta}{2}\mathsf{H} + \epsilon\frac{d}{2}\sin\frac{\theta}{2}(\mathsf{G}\times\mathsf{H}) \end{aligned} \tag{5.3}$$

for the first case, and

$$\begin{aligned} Q^0_{PR} &= \cos\frac{\theta}{2} - \epsilon\frac{d}{2}\sin\frac{\theta}{2}\mathsf{G}\cdot\mathsf{H}, \\ \mathsf{Q}_{PR} &= \sin\frac{\theta}{2}\mathsf{G} + \epsilon\frac{d}{2}\cos\frac{\theta}{2}\mathsf{H} + \epsilon\frac{d}{2}\sin\frac{\theta}{2}(\mathsf{H}\times\mathsf{G}) \end{aligned} \tag{5.4}$$

for the second. Notice that the only difference is a sign change due to the cross product in the vector part.

The design variables are the coordinates of the joint axis G and the direction of the prismatic joint axis, $\mathbf{h}$. The maximum number of task positions according to the equations (3.16), (3.17) and (3.19) is $n_{max} = 2.5$ and $n_R = 2$, $n_T = 3$. It is a case of orientation-limited chain. We can define extra constraints on the translational variables or we can solve for 2 complete arbitrary positions plus one extra position with orientation belonging to the workspace. The special case of a robot consisting of a single cylindrical joint C corresponds to the RP robot if we define two extra constraints making the direction of H parallel to the direction of G. In this case, the count gives a maximum of $n_{max} = 2$ task positions and there exists a unique solution.

To create the design equations we equate the kinematics equations to each of the

goal dual quaternion $\hat{P}^i$, that is,

$$\hat{Q}_{RP}(\theta^i, d^i) - \hat{P}^i = \vec{0}. \tag{5.5}$$

We expand the design equations,

$$\begin{Bmatrix} \sin\frac{\theta}{2}\mathbf{g} + \epsilon(\sin\frac{\theta}{2}\mathbf{g}^0 + \frac{d}{2}\cos\frac{\theta}{2}\mathbf{h} \pm \frac{d}{2}\sin\frac{\theta}{2}(\mathbf{g}\times\mathbf{h})) \\ \cos\frac{\theta}{2} - \epsilon\frac{d}{2}\sin\frac{\theta}{2}(\mathbf{g}\cdot\mathbf{h}) \end{Bmatrix} - \begin{Bmatrix} \mathbf{p} + \epsilon\mathbf{p}^0 \\ p_w + \epsilon p_w^0 \end{Bmatrix} = \vec{0}, \tag{5.6}$$

where the $\pm$ sign corresponds to the RP and PR cases.

Solving the design equations

For this case we do not solve the parameterized equations, as it is easy to eliminate the joint parameters θ and d. We perform the implicitization as described in Section 3.8 to eliminate the joint variable θ, to obtain

$$\begin{aligned} g_x &= \frac{p_x}{\sqrt{p_x^2 + p_y^2 + p_z^2}} \\ g_y &= \frac{p_y}{\sqrt{p_x^2 + p_y^2 + p_z^2}} \\ g_z &= \frac{p_z}{\sqrt{p_x^2 + p_y^2 + p_z^2}}. \end{aligned} \tag{5.7}$$

If we decide to solve for an extra general translation with rotation in the workspace, we have two sets of vector equations. We substitute the solution values for the cosine and sine of θ in the moment equations,

$$\begin{aligned} &d(g_z(h_x p_w - h_y p_z) + g_y h_z p_z) + 2g_{x0}p_z - 2p_{x0}g_z = 0 \\ &d(g_z(h_y p_w + h_x p_z) - g_x h_z p_z) + 2g_{y0}p_z - 2p_{y0}g_z = 0 \\ &d(g_z h_z p_w + g_x h_y p_z - g_y h_x p_z) + 2g_{z0}p_z - 2p_{z0}g_z = 0, \end{aligned} \tag{5.8}$$

then we solve linearly for d at each equation,

$$
\begin{aligned}
d_x &= \frac{2(p_{x0}g_z - g_{x0}p_z)}{g_z(h_x p_w - h_y p_z) + g_y h_z p_z} \\
d_y &= \frac{2(p_{y0}g_z - g_{y0}p_z)}{g_z(h_y p_w + h_x p_z) - g_x h_z p_z} \\
d_z &= \frac{2(p_{z0}g_z - g_{z0}p_z)}{g_z h_z p_w + g_x h_y p_z - g_y h_x p_z},
\end{aligned} \tag{5.9}
$$

and impose that the value of d must be the same in all cases, that is, $d_x = d_y$ and $d_x = d_z$. These 2 equations per relative displacement, together with the the Plücker constraints, form the set of reduced equations to solve for $\mathbf{g}_0$ and $\mathbf{h}$,

$$
\begin{aligned}
&\{d_x = d_y, d_x = d_z\}^i, \quad i = 2, 3, \\
&g_x g_x^0 + g_y g_y^0 + g_z g_z^0 = 0 \\
&h_x^2 + h_y^2 + h_z^2 = 1.
\end{aligned} \tag{5.10}
$$

We use resultant techniques to further simplify the system . We clear the denominators of the design equations and use the Plücker condition in Eq.(5.10) to eliminate one of the components of $\mathbf{g}_0$, for instance g_{x0} and solve linearly for the other two,

$$
\begin{bmatrix}
A_{11} & A_{12} & A_{13} \\
A_{21} & A_{22} & A_{23} \\
A_{31} & A_{32} & A_{33} \\
A_{41} & A_{42} & A_{43}
\end{bmatrix}
\begin{Bmatrix}
g_{y0} \\ g_{z0} \\ 1
\end{Bmatrix}
= \left\{\vec{0}\right\}, \tag{5.11}
$$

where the coefficients of the matrix are linear functions of $\mathbf{h}$, that is, $A_{ij} = K_x h_x + K_y h_y + K_z h_z$. In order for the system to have solutions, we impose that the 3×3 minors must be equal to zero. The result is a set of four homogeneous polynomials of

cubic total degree in the components of $\mathbf{h}$. We can set $h_z = 1$ for instance and solve linearly for the powers of h_x,

$$\begin{bmatrix} B_{11} & B_{12} & B_{13} & B_{14} \\ B_{21} & B_{22} & B_{23} & B_{24} \\ B_{31} & B_{32} & B_{33} & B_{34} \\ B_{41} & B_{42} & B_{43} & B_{44} \end{bmatrix} \begin{Bmatrix} h_x^3 \\ h_x^2 \\ h_x \\ 1 \end{Bmatrix} = \begin{Bmatrix} 0 \end{Bmatrix}. \tag{5.12}$$

A solution will exist when the determinant of this matrix is equal to zero. The determinant is a six-degree polynomial in h_y,

$$Det([B]) = c_0 + c_1 h_y + c_2 h_y^2 + c_3 h_y^3 + c_4 h_y^4 + c_5 h_y^5 + c_6 h_y^6 = 0 \tag{5.13}$$

with at most four real roots, that we can solve to obtain solution values for h_y. Substitute the solutions and solve linearly for the rest of the variables. Three of the four roots correspond to extraneous roots introduced when clearing denominators, and we obtain one RP solution.

Example

The task positions are shown in Table 5.1. The third one corresponds to a general translation with orientation bounded to the workspace of the chain.

Table 5.1: Task positions for an RP chain

Position	*Axis*	*Rotation*	*Translation*
position 1	$(1.0, 0.0, 0.0) + \epsilon(0.0, 0.0, 0.0)$	0^o	0
position 2	$(0.82, -0.10, 0.56) + \epsilon(-0.71, -2.04, 0.67)$	253.2^o	0.98
position 3	$(0.82, -0.10, 0.56) + \epsilon(0.55, -1.66, -1.11)$	225.4^o	0.53

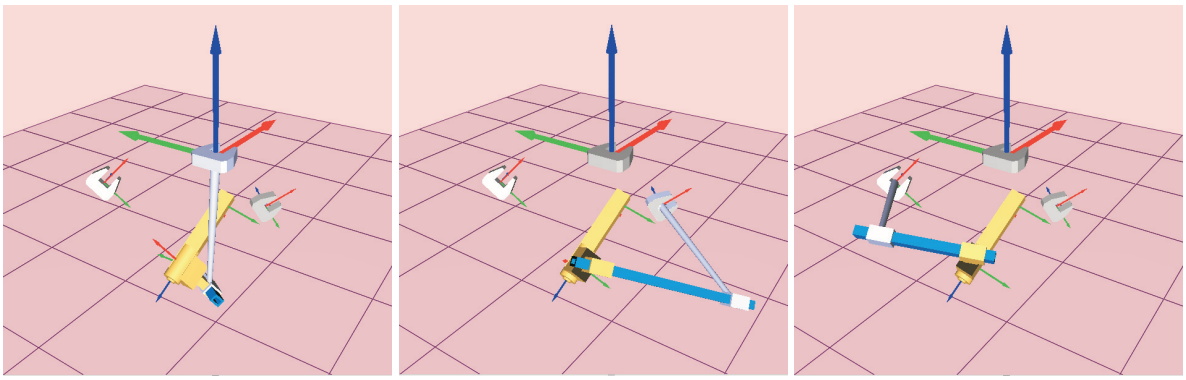

Figure 5.2: The spatial RP robot reaching two complete positions plus one translation

The solution is presented in Table(5.2). Figure 5.2 shows the RP chain while reaching each of the positions. The RP robot was visualized using the software SYNTHETICA, [93].

Table 5.2: The solution for the RP chain

Joint Axis	*Direction*	*Moment*
G	$(-0.82, 0.10, -0.56)$	$(-0.15, 1.67, 0.52)$
H	$(-0.58, -0.81, 0.03)$	$(-2.17, 1.53, -0.06)$

The solution for the particular case of a C robot consists of making the joint axes, angle and displacement equal to the relative screw that defines the two task positions.

5.2.2 Spatial RR robot

The spatial RR robot is a two-degree-of-freedom chain consisting of two revolute axes, the fixed axis G of rotation θ and the moving axis W of rotation ϕ, which are not parallel nor concurrent, see Figure 5.3.

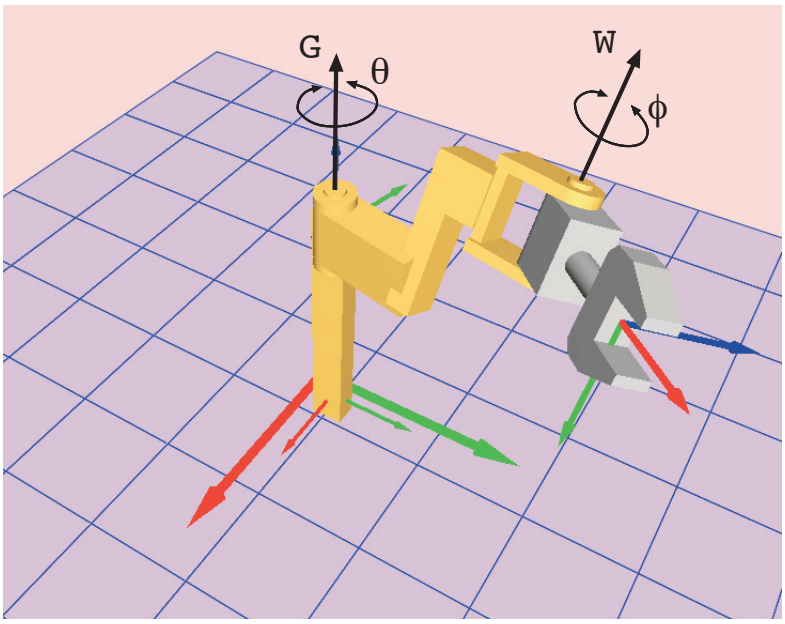

Figure 5.3: The spatial RR robot

A brief history

The design of spatial RR chains has been a focus of research since the first attempts of generalizing Burmerster's theory by Roth [83]. Veldkamp [103] solved the problem for three instantaneous positions and showed that there exist two solutions. Later, Suh [95] used screw theory to show that, in all cases, the finite position synthesis yields two solutions that form a Bennett linkage. Finally, Tsai and Roth, [98], [99] used the equivalent screw triangle to solve the finite position synthesis and presented an algebraic solution that yielded two RR linkages. The problem has been recently studied in different ways ([52], [73]).

The design equations

The dual quaternion kinematics equations are written as

$$\hat{Q}_{RR} = \hat{G}(\theta, 0)\hat{W}(\phi, 0), \tag{5.14}$$

where the dual point Q^0 and the dual vector Q are

$$\begin{aligned} Q^0 &= \cos\frac{\theta}{2}\cos\frac{\phi}{2} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}\mathsf{G}\cdot\mathsf{W}, \\ \mathsf{Q} &= \sin\frac{\theta}{2}\cos\frac{\phi}{2}\mathsf{G} + \cos\frac{\theta}{2}\sin\frac{\phi}{2}\mathsf{W} + \sin\frac{\theta}{2}\sin\frac{\phi}{2}(\mathsf{G}\times\mathsf{W}). \end{aligned} \tag{5.15}$$

We can define up to $n_{max} = 3$ complete positions, while $n_R = 5$ positions. Hence this is not a orientation-limited chain and we solve for three arbitrary task positions.

To create the design equations we equate Eq.(5.14) to each of the task dual quaternions $\hat{P}$, that is,

$$\hat{Q}_{RR}(\theta^i, \phi^i) - \hat{P}^i = \vec{0}, \quad i = 2, 3, \tag{5.16}$$

to obtain the set of design equations. If $\mathsf{G} = \mathbf{g} + \epsilon\mathbf{g}^0 = (g_x, g_y, g_z) + \epsilon(g_x^0, g_y^0, g_z^0)$ and $\mathsf{W} = \mathbf{w} + \epsilon\mathbf{w}^0 = (w_x, w_y, w_z) + \epsilon(w_x^0, w_y^0, w_z^0)$, they expand to

$$\begin{Bmatrix} \cos\frac{\theta}{2}\sin\frac{\phi}{2}\mathbf{w} + \sin\frac{\theta}{2}\cos\frac{\phi}{2}\mathbf{g} + \sin\frac{\theta}{2}\sin\frac{\phi}{2}\mathbf{g}\times\mathbf{w} \\ \cos\frac{\theta}{2}\cos\frac{\phi}{2} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}\mathbf{g}\cdot\mathbf{w} \end{Bmatrix}$$
$$+\epsilon\begin{Bmatrix} \cos\frac{\theta}{2}\sin\frac{\phi}{2}\mathbf{w}^0 + \sin\frac{\theta}{2}\cos\frac{\phi}{2}\mathbf{g}^0 + \sin\frac{\theta}{2}\sin\frac{\phi}{2}(\mathbf{g}\times\mathbf{w}^0 + \mathbf{g}^0\times\mathbf{w}) \\ -\sin\frac{\theta}{2}\sin\frac{\phi}{2}(\mathbf{g}\cdot\mathbf{w}^0 + \mathbf{g}^0\cdot\mathbf{w}) \end{Bmatrix} = \begin{Bmatrix} \mathbf{p} + \epsilon\mathbf{p}^0 \\ p_w + \epsilon p_w^0 \end{Bmatrix}. \tag{5.17}$$

Solving the design equations

We eliminate the joint parameters θ and ϕ as described in section 3.8. In this case we cannot solve separately for the rotations; the directions of G and W are not fully specified by the rotations but also by the translations.

The values for the angles of Eq.(3.41),

$$\begin{aligned}
\cos\frac{\theta}{2}\sin\frac{\phi}{2} &= \frac{\mathbf{p}\cdot(-\mathbf{g}\times(\mathbf{g}\times\mathbf{w}))}{(\mathbf{g}\times\mathbf{w})\cdot(\mathbf{g}\times\mathbf{w})} \\
\sin\frac{\theta}{2}\cos\frac{\phi}{2} &= \frac{\mathbf{p}\cdot((-\mathbf{g}\times\mathbf{w})\times\mathbf{w})}{(\mathbf{g}\times\mathbf{w})\cdot(\mathbf{g}\times\mathbf{w})} \\
\sin\frac{\theta}{2}\sin\frac{\phi}{2} &= \frac{\mathbf{p}\cdot(\mathbf{g}\times\mathbf{w})}{(\mathbf{g}\times\mathbf{w})\cdot(\mathbf{g}\times\mathbf{w})} \\
\cos\frac{\theta}{2}\cos\frac{\phi}{2} &= \frac{\mathbf{p}\cdot(\mathbf{g}\times\mathbf{w})(\mathbf{g}\cdot\mathbf{w})}{(\mathbf{g}\times\mathbf{w})\cdot(\mathbf{g}\times\mathbf{w})} + p_w
\end{aligned} \tag{5.18}$$

are used to obtain the reduced design equation, Eq.(4.19),

$$\begin{aligned}
\mathcal{R}: \quad & g_x((-p_y^2-p_z^2)w_x + (p_xp_y - p_wp_z)w_y + (p_wp_y + p_xp_z)w_z) + \\
& g_y((p_xp_y + p_wp_z)w_x + (-p_x^2-p_z^2)w_y + (p_yp_z - p_wp_x)w_z) + \\
& g_z((p_xp_z - p_wp_y)w_x + (p_wp_x + p_yp_z)w_y + (-p_x^2-p_y^2)w_z) = 0.
\end{aligned} \tag{5.19}$$

We substitute the values in Eq. (5.18) in the four dual components of the dual quaternion of Eq. (5.17) to obtain four equations that are free of joint variables. three of them are independent; one of them can be written as a polynomial combination of the other three, as shown in Chapter 4.

The set of reduced design equations contains the angle condition in eq. (5.19) plus

the three first moment components of Eq.(5.17),

$$\begin{aligned}\mathcal{M}: \quad & (\mathbf{p}\cdot(-\mathbf{g}\times(\mathbf{g}\times\mathbf{w}))\mathbf{w}^0+(\mathbf{p}\cdot((-\mathbf{g}\times\mathbf{w})\times\mathbf{w}))\mathbf{g}^0+\\ & (\mathbf{p}\cdot(\mathbf{g}\times\mathbf{w}))(\mathbf{g}\times\mathbf{w}^0+\mathbf{g}^0\times\mathbf{w})=((\mathbf{g}\times\mathbf{w})\cdot(\mathbf{g}\times\mathbf{w}))\mathbf{p}^0, \qquad (5.20)\end{aligned}$$

and the Plücker conditions for the axes. This set of reduced design equations,

$$\begin{aligned}&\{\mathcal{R},\mathcal{M}_x,\mathcal{M}_y,\mathcal{M}_z\}^i, \quad i=2,3,\\ &\mathbf{g}\cdot\mathbf{g}=1,\mathbf{g}\cdot\mathbf{g}^0=0,\\ &\mathbf{w}\cdot\mathbf{w}=1,\mathbf{w}\cdot\mathbf{w}^0=0. \qquad (5.21)\end{aligned}$$

consists of 12 equations in 12 unknowns. We do not present an algebraic elimination here, as it is already known, see [99]. A closed algebraic solution that considers the line geometry of the workspace is presented in Chapter 6. However, the equations are simple enough for a symbolic program to solve them.

Example

The solutions presented in Table 5.4 and Figure 5.4 for the example of Table 5.3 have been generated using the Mathematica solver. Only the two real solutions are presented here; for the complete set of solutions see Appendix A.

5.3 Synthesis of Three-degree-of-freedom Robots

Most of the three-degree-of-freedom robots have been solved recently. In this section we include all basic combinations of revolute and prismatic joints and also those

Table 5.3: Task positions for an RR chain

Position	*Axis*	*Rotation*	*Translation*
position 1	$(1.0, 0.0, 0.0) + \epsilon(0.0, 0.0, 0.0)$	0^o	0
position 2	$(-0.43, -0.75, 0.49) + \epsilon(0.26, -1.37, -1.85)$	113.0^o	2.59
position 3	$(0.03, -0.80, -0.60) + \epsilon(1.68, 1.21, -1.51)$	142.6^o	-1.24

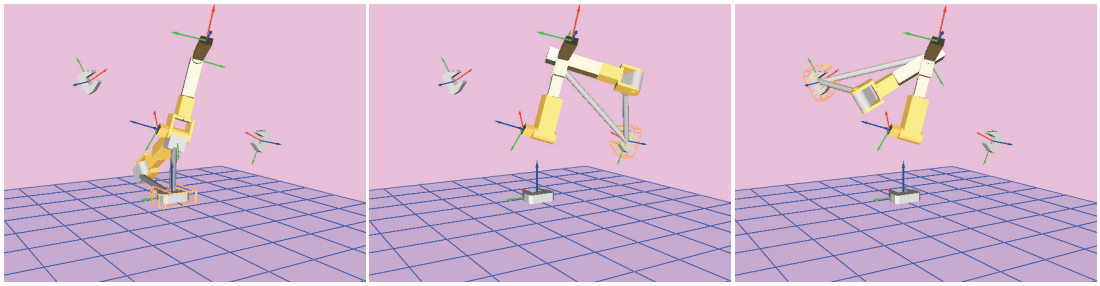

Figure 5.4: Two spatial RR robots reaching three positions

special cases that are of interest, that is, some robots with universal (T) or cylindrical (C) joints.

5.3.1 Spatial RPP robot

The spatial RPP robot is a three-degree-of-freedom robot. It consists of one fixed revolute joint $\mathsf{G}(\theta)$ and two prismatic joints that provide translations d and b along directions **h** and **u** , see Figure 5.5.

A brief history

The finite-position synthesis of chains consisting of three links with two prismatic joints was considered by Tsai [97] to be equivalent to the corresponding dyad with one prismatic joint, and this is essentially true when considering only complete task

Table 5.4: The two real solutions for the RR chain

Joint Axis	*Direction*	*Moment*
G	$(0.14, 0.94, 0.30)$	$(-1.32, -0.33, 1.68)$
W	$(0.59, 0.03, 0.81)$	$(-0.81, -2.56, 0.69)$
G	$(-0.88, 0.45, 0.17)$	$(-1.57, -2.76, -0.73)$
W	$(0.14, 0.94, 0.30)$	$(-1.32, -0.33, 1.69)$

positions. No other information about previous solutions could be found, but this chain is within the possibilities of geometric constraint synthesis.

The design equations

The dual quaternion kinematics equations,

$$\hat{Q}_{RPP} = \hat{G}(\theta, 0)\hat{H}(0, d)\hat{U}(0, b), \tag{5.22}$$

are expanded as $\hat{Q}_{RPP} = Q^0 + \mathsf{Q}$, where the point is

$$Q^0 = \cos\frac{\theta}{2} - \epsilon(\sin\frac{\theta}{2}(\frac{d}{2}\mathsf{G}\cdot\mathsf{H} + \frac{b}{2}\mathsf{G}\cdot\mathsf{U})), \tag{5.23}$$

and the vector

$$\mathsf{Q} = \sin\frac{\theta}{2}\mathsf{G} + \epsilon(\cos\frac{\theta}{2}(\frac{d}{2}\mathsf{H} + \frac{b}{2}\mathsf{U}) + \sin\frac{\theta}{2}(\frac{d}{2}\mathsf{G}\times\mathsf{H} + \frac{b}{2}\mathsf{G}\times\mathsf{U})). \tag{5.24}$$

Using the counting formula in Eq.(3.17) with $r = 1$, we obtain $n_R = 2$. Applying the results of Chapter 4, we substitute the two consecutive prismatic directions by the common normal $\mathbf{n}$ to the plane that they define, to obtain $n_{max} = 3$. We can solve for at most two complete task positions. Notice that the design for the PRP robot

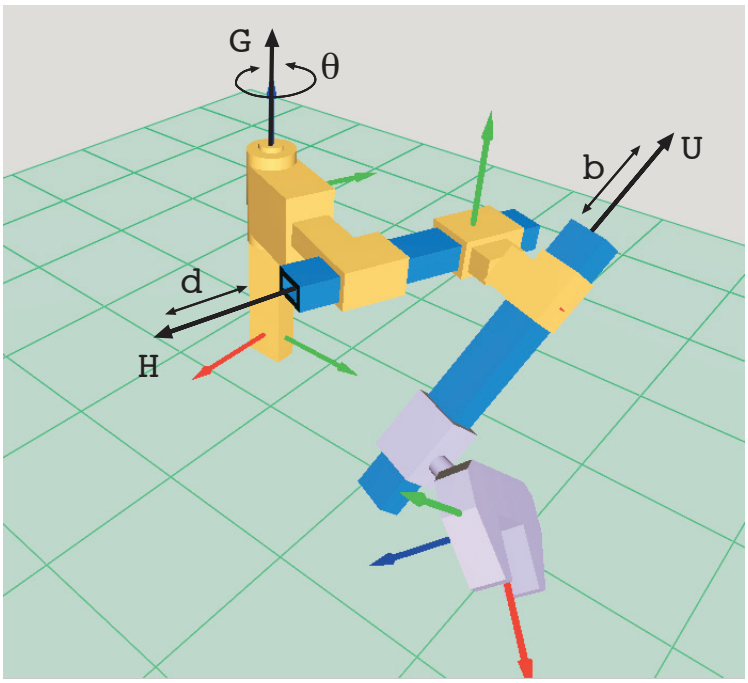

Figure 5.5: The spatial RPP robot

yields a different counting because the prismatic axes are separated by the revolute joint.

Once one direction and the plane are fixed, the second direction in the plane can be chosen arbitarly with the only condition of not being parallel to the first. Computing the maximum number of translations using Eq.(3.19) yields $n_T = 5$. We solve for two complete task positions plus three additional arbitrary translations that must have an orientation given by a rotation angle around $\mathbf{g}$. The initial set of parameterized

design equations is

$$\begin{aligned}
&\hat{G}(\theta^2,0)\hat{H}(0,d^2)\hat{U}(0,b^2) = \hat{P}^2,\\
&(\hat{G}(\theta^i,0)\hat{H}(0,d^i)\hat{U}(0,b^i))^\circ = \hat{P}^{i\circ}, \quad i=3,4,5,\\
&\mathbf{n}\cdot\mathbf{n}=1, \quad \mathbf{g}\cdot\mathbf{g}=1, \quad \mathbf{g}\cdot\mathbf{g}^0=0.
\end{aligned} \tag{5.25}$$

These are 18 equation in 18 unknowns, 9 of them corresponding to the structural variables G and $\mathbf{n}$. This set of equations can be reduced to obtain a closed algebraic solution.

Solving the design equations

We solve for the direction of G as in Section 5.2.1, to obtain as a result that $\mathbf{g}$ must be equal to the direction of the screw axis of the relative displacement $\hat{P}^2$. We also obtain the inverse kinematics for the angle θ

Substitute the expression of θ in the set of equations $(\hat{G}(\theta^i,0)\hat{H}(0,d^i)\hat{U}(0,b^i))^\circ = \hat{P}^{i\circ}, \quad i=2,\ldots,5$, and discard the dual scalar part. We solve linearly for d and b in the system,

$$\begin{bmatrix}
\vdots & \vdots & \vdots \\
\frac{1}{2}(p_w\mathbf{h}-\mathbf{h}\times\mathbf{p}) & \frac{1}{2}(p_w\mathbf{u}-\mathbf{u}\times\mathbf{p}) & \mathbf{g}^\circ\sqrt{\mathbf{p}\cdot\mathbf{p}}-\mathbf{p}^\circ \\
\vdots & \vdots & \vdots
\end{bmatrix}
\begin{Bmatrix} d \\ b \\ 1 \end{Bmatrix} =
\begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}, \tag{5.26}$$

where we denote $\mathbf{p}$ and $\mathbf{p}^\circ$ the dual vector of the task dual quaternion.

Impose the condition that the determinant of the matrix in Eq.(5.26) must be zero, and solve linearly for d, b choosing two of the equations. The determinant of

the matrix can be expressed, if we make $\mathbf{n} = \mathbf{h} \times \mathbf{u}$, as

$$\mathbf{n} \cdot (p_w^2(\sqrt{\mathbf{p} \cdot \mathbf{p}}\mathbf{g}^\circ - \mathbf{p}^\circ) + p_w(\sqrt{\mathbf{p} \cdot \mathbf{p}}\mathbf{g}^\circ - \mathbf{p}^\circ) \times \mathbf{p} + ((\sqrt{\mathbf{p} \cdot \mathbf{p}}\mathbf{g}^\circ - \mathbf{p}^\circ) \cdot \mathbf{p})\mathbf{p}) = 0. \tag{5.27}$$

We make $n_z = 1$ and eliminate g_z^0 using the Plücker condition. The bilinear system of four equations in four unknowns is solved to obtain at most six complex solutions.

Numerical example

We present an example in which we want to design the RPP robot to reach the positions presented in Table 5.5. The first two are arbitrary and the last three have arbitrary translations with rotations belonging to the workspace of the chain.

Table 5.5: Task positions for an RPP chain

Position	*Axis*	*Rotation*	*Translation*
position 1	$(1.0, 0.0, 0.0) + \epsilon(0.0, 0.0, 0.0)$	0^o	0
position 2	$(0.98, -0.14, 0.16) + \epsilon(-0.51, -1.81, 1.51)$	1.75	2.21
position 3	$(0.98, -0.14, 0.16) + \epsilon(0.12, 0.02, -0.70)$	4.28	3.97
position 4	$(0.98, -0.14, 0.16) + \epsilon(0.28, 1.70, -0.25)$	0.96	0.51
position 5	$(0.98, -0.14, 0.16) + \epsilon(-0.06, 0.18, 0.52)$	5.43	3.06

We obtain six complex roots, which are listed in Appendix A –we pick $\mathbf{h} = \mathbf{n} \times \mathbf{x}$. Two of the roots are real but one corresponds to the degenerate case $\mathbf{n} = \mathbf{g}$. The only real solution is presented, as the joint axes in the first position,in Table 5.6. Figure 5.6 shows the robot while reaching each of the positions.

Table 5.6: The solution for the RPP chain

Joint Axis	*Direction*	*Moment*
G	$(0.98, -0.14, 0.16)$	$(0.25, 1.45, -0.29)$
$\mathbf{h}$	$(0.0, 0.25, -0.97)$	
$\mathbf{u}$	$(-0.96, -0.26, -0.07)$	

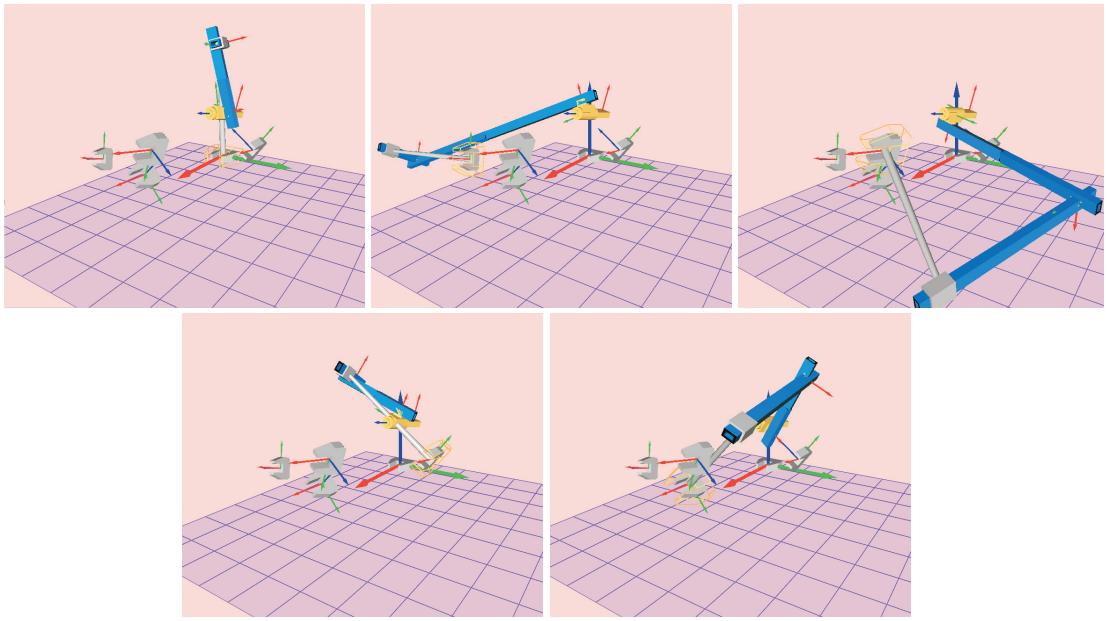

Figure 5.6: The solution RPP robot reaching two complete positions plus three translations

5.3.2 Spatial RRP robot

There are three possible assemblies for robots consisting of two revolute joints and one prismatic joint: RRP, RPR and PRR. From the kinematic synthesis point of view, all three topologies belong to the same class: the only difference are the signs in the anti-commutative part of the design equations. Here we state and solve the design equations for the RRP robot.

The RRP robot is a three-degree-of-freedom robot. It consists of two revolute joints, a fixed axis G that allows rotation of angle θ about it and a moving axis W of rotation ϕ about it, followed by a prismatic joint of translation d along an arbitrary axis H with direction **h**, see Figure 5.7.

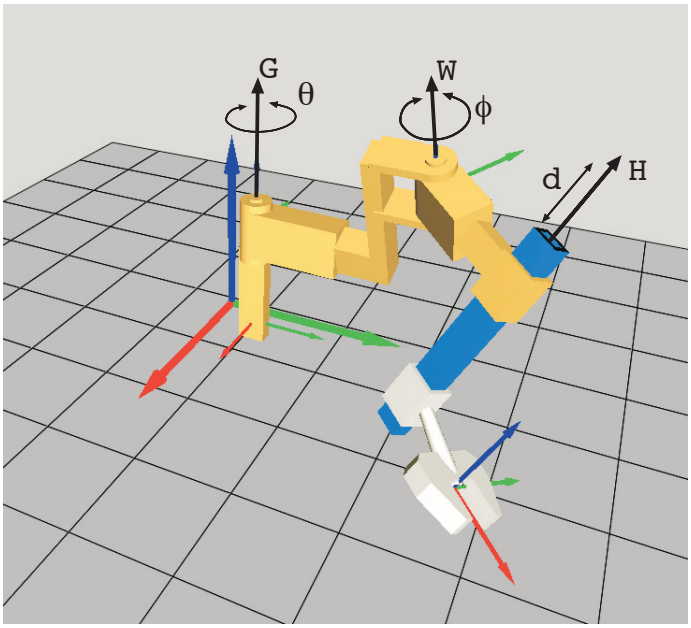

Figure 5.7: The spatial RRP robot

A brief history

The finite-position synthesis design equations for linkages consisting of two revolute and one prismatic joints were stated by Tsai [97] using the equivalent screw triangle. The PRR robot was stated and partially solved by Sandor [90] for three task positions, using the free parameters to optimize the performance of the mechanism. Recently, solutions were obtained almost simultaneously using dual quaternion synthesis [71] for solving the RPR robot, and using matrix kinematics equations [47] for the PRR

robot. In this last case, the authors were able to reduce the design equations to a univariate polynomial of degree 30, out of which 12 roots are solutions to the design problem.

The design equations

The dual quaternion representation for the relative displacements,

$$\hat{Q}_{RRP} = \hat{G}(\theta, 0)\hat{W}(\phi, 0)\hat{H}(0, d), \tag{5.28}$$

is expanded as $\hat{Q}_{RRP} = Q^0 + \mathsf{Q}$, where the point is

$$\begin{aligned} Q^0 = &\cos\frac{\theta}{2}\cos\frac{\phi}{2} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}\mathsf{G}\cdot\mathsf{W} - \epsilon(\frac{d}{2}\sin\frac{\theta}{2}\cos\frac{\phi}{2}\mathsf{G}\cdot\mathsf{H} + \\ &\frac{d}{2}\cos\frac{\theta}{2}\sin\frac{\phi}{2}\mathsf{W}\cdot\mathsf{H} + \frac{d}{2}\sin\frac{\theta}{2}\sin\frac{\phi}{2}(\mathsf{G}\times\mathsf{W})\cdot\mathsf{H}), \end{aligned} \tag{5.29}$$

and the vector

$$\begin{aligned} \mathsf{Q} = &\sin\frac{\theta}{2}\cos\frac{\phi}{2}\mathsf{G} + \cos\frac{\theta}{2}\sin\frac{\phi}{2}\mathsf{W} + \sin\frac{\theta}{2}\sin\frac{\phi}{2}(\mathsf{G}\times\mathsf{W}) + \\ &\epsilon\frac{d}{2}(\cos\frac{\theta}{2}\cos\frac{\phi}{2}\mathsf{H} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}(\mathsf{G}\cdot\mathsf{W})\mathsf{H} + \sin\frac{\theta}{2}\cos\frac{\phi}{2}(\mathsf{G}\times\mathsf{H}) + \\ &\cos\frac{\theta}{2}\sin\frac{\phi}{2}(\mathsf{W}\times\mathsf{H}) + \sin\frac{\theta}{2}\sin\frac{\phi}{2}(\mathsf{G}\times\mathsf{W})\times\mathsf{H}), \end{aligned} \tag{5.30}$$

and equated to the goal positions,

$$\hat{G}(\theta^i, 0)\hat{W}(\phi^i, 0)\hat{H}(0, d^i) = \hat{P}^i, \quad i = 2, 3, 4. \tag{5.31}$$

We can define up to $n_{max} = 4 + \frac{1}{3}$ positions, while $n_R = 5$. These are four full positions plus two of the six parameters that define a fifth position. We can include the fifth partially specified position or we can add one extra constraint on the joint axes. If we specify the constraint on the directions of the revolute joint, we obtain $n_R = 4$ and we can solve separately for the direction of the revolute joints. This simplifies the

solution process and allows us to obtain a simple closed algebraic solution. However, any other constraint on any of the structural variables can be used instead.

Solving the design equations

We eliminate the three joint variables following the procedure explained in Section 3.8.2, with the block matrices of Eq.(3.51) and Eq.(3.52) being

$$\begin{bmatrix} A \\ \dots \\ B \end{bmatrix} = \begin{bmatrix} \mathbf{g} & \mathbf{w} & \mathbf{g}\times\mathbf{w} & \vec{0} \\ 0 & 0 & -\mathbf{g}\cdot\mathbf{w} & 1 \\ \dots & \dots & \dots & \dots \\ \mathbf{g}^0 & \mathbf{w}^0 & \mathbf{g}^0\times\mathbf{w}+\mathbf{g}\times\mathbf{w}^0 & \vec{0} \\ 0 & 0 & -(\mathbf{g}^0\cdot\mathbf{w}+\mathbf{g}\cdot\mathbf{w}^0) & 0 \end{bmatrix}, \tag{5.32}$$

and

$$[C] = \begin{bmatrix} \mathbf{w}\times\mathbf{h} & \mathbf{g}\times\mathbf{h} & \mathbf{h} & (\mathbf{g}\times\mathbf{w})\times\mathbf{h}-(\mathbf{g}\cdot\mathbf{w})\mathbf{h} \\ -\mathbf{w}\cdot\mathbf{h} & -\mathbf{g}\cdot\mathbf{h} & 0 & -(\mathbf{g}\times\mathbf{w})\cdot\mathbf{h} \end{bmatrix}. \tag{5.33}$$

We invert the block matrix using Eq.(3.54). In this case, the inverses of the $[A]$, $[C]$ matrices are easy to compute,

$$[A]^{-1} = \frac{1}{(\mathbf{g}\times\mathbf{w})^2} \begin{bmatrix} \mathbf{w}\times(\mathbf{g}\times\mathbf{w})^T & 0 \\ -\mathbf{g}\times(\mathbf{g}\times\mathbf{w})^T & 0 \\ \mathbf{g}\times\mathbf{w}^T & 0 \\ (\mathbf{g}\cdot\mathbf{w})\mathbf{g}\times\mathbf{w}^T & (\mathbf{g}\times\mathbf{w})^2 \end{bmatrix} \tag{5.34}$$

and,

$$[C]^{-1} =$$

$$\frac{1}{(\mathbf{g}\times\mathbf{w})^2}\begin{bmatrix} -(\mathbf{g}\times(\mathbf{g}\times\mathbf{w}))\times\mathbf{h}^T & \mathbf{h}\cdot(\mathbf{g}\times(\mathbf{g}\times\mathbf{w})) \\ -(\mathbf{w}\times(\mathbf{w}\times\mathbf{g}))\times\mathbf{h}^T & \mathbf{h}\cdot(\mathbf{w}\times(\mathbf{w}\times\mathbf{g})) \\ (\mathbf{g}\cdot\mathbf{w})(\mathbf{g}\times\mathbf{w})\times\mathbf{h}^T+(\mathbf{g}\times\mathbf{w})^2\mathbf{h}^T & -(\mathbf{g}\cdot\mathbf{w})\mathbf{h}\cdot(\mathbf{g}\times\mathbf{w}) \\ (\mathbf{g}\times\mathbf{w})\times\mathbf{h}^T & -\mathbf{h}\cdot(\mathbf{g}\times\mathbf{w}) \end{bmatrix}, \tag{5.35}$$

and the values of the joint variables are given by solving the linear system as in Eq.(3.55) and Eq.(3.56). The expressions obtained for the inverse kinematics are substituted in Eq.(3.57) and Eq.(3.58) to create the three design equations $\mathcal{R}, \mathcal{M}_1, \mathcal{M}_2$. Notice that $\mathcal{R}$,

$$\mathcal{R}: \quad p_w\mathbf{g}\cdot\mathbf{p}\times\mathbf{w}+\mathbf{g}\cdot(\mathbf{p}\times(\mathbf{p}\times\mathbf{w}))=0, \tag{5.36}$$

where we require that $\mathbf{g}\times\mathbf{w}\neq 0$, which restricts the pair of revolute joints from generating a pure planar movement, is exactly the same as Eq.(5.19). This is true for any spatial robot with two revolute joints.

The final set of reduced design equations,

$$\begin{aligned} &\{\mathcal{R},\mathcal{M}_1,\mathcal{M}_2\}^{\mathrm{i}}, \quad \mathrm{i}=2,3,4, \\ &g_x^2+g_y^2+g_z^2=1, \quad w_x^2+w_y^2+w_z^2=1, \quad h_x^2+h_y^2+h_z^2=1, \\ &g_xg_x^0+g_yg_y^0+g_zg_z^0=0, \quad w_xw_x^0+w_yw_y^0+w_zw_z^0=0, \\ &c(\mathbf{g}), \end{aligned} \tag{5.37}$$

contains 15 equations in the 15 parameters of the joint axes. The last equation, $c(\mathbf{g})$, is any constraint on the joint direction $\mathbf{g}$.

This set of reduced design equations can be further simplified to a univariate polynomial using resultant techniques. For the following simplification, we use the condition $g_x = 0$.

As we indicated, we can solve separately for the directions $\mathbf{g}$ and $\mathbf{w}$ using the three equations $\mathcal{R}^i$ and the extra condition $g_x = 0$. We make $g_z = 1, w_z = 1$ to obtain four equations in four unknowns. The procedure that follows is the standard way to solve for two directions and can be found for instance in [58].

Assemble the polynomials $\mathcal{R}^i$ into the matrix equation,

$$\begin{bmatrix} a_{11}g_y + b_{11} & a_{12}g_y + b_{12} & a_{13}g_y + b_{13} \\ a_{21}g_y + b_{21} & a_{22}g_y + b_{22} & a_{23}g_y + b_{23} \\ a_{31}g_y + b_{31} & a_{32}g_y + b_{32} & a_{33}g_y + b_{33} \end{bmatrix} \begin{Bmatrix} w_x \\ w_y \\ 1 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}. \tag{5.38}$$

The terms a_{ij} and b_{ij} are constants determined by the task positions and the normalizing conditions. This matrix equation has a solution only if the rank of the coefficient matrix is less than 3. To achieve this we set the determinant of the matrix to zero. The result is a cubic polynomial in g_y,

$$\mathcal{D}: \quad d_0 + d_1 g_y + d_2 g_y^2 + d_3 g_y^3 = 0. \tag{5.39}$$

The three roots for g_y are used to solve linearly for $\mathbf{w}$ in Eq.(5.38). Thus, we obtain as many as three solutions for the directions $\mathbf{g}$ and $\mathbf{w}$ in the RRP chain. Notice that this result applies to the RPR and PRR chains as well, because the equations are the same for all three chains.

Substitute the solutions in the equations $\mathcal{M}_1^i, \mathcal{M}_2^i$, which are linear in the components $\mathbf{h} = (h_x, h_y, h_z)$, $\mathbf{g}^0 = (g_{x0}, g_{y0}, g_{z0})$ and $\mathbf{w}^0 = (w_{x0}, w_{y0}, w_{z0})$, to obtain six

equations of the form,

$$\begin{aligned}\mathcal{M}: \quad & h_x(\mathbf{A}_x \cdot \mathbf{g}^0 + \mathbf{B}_x \cdot \mathbf{w}^0 + c_x) + h_y(\mathbf{A}_y \cdot \mathbf{g}^0 + \mathbf{B}_y \cdot \mathbf{w}^0 + c_y) \\ & + h_z(\mathbf{A}_z \cdot \mathbf{g}^0 + \mathbf{B}_z \cdot \mathbf{w}^0 + c_z) = 0. \end{aligned} \tag{5.40}$$

The coefficient vectors $\mathbf{A}$, $\mathbf{B}$ and c are constants determined by the specified task positions and the values of $\mathbf{g}$ and $\mathbf{w}$.

Out of the nine parameters, we can eliminate three by making $h_z = 1$ and using the Plücker conditions to eliminate g_{y0}, w_{y0}. We obtain a system of six equations in six unknowns, that we write as the 6×5 linear system,

$$\begin{bmatrix} a_{11}h_x + b_{11}h_y + c_{11} & a_{12}h_x + b_{12}h_y + c_{12} & \dots & a_{15}h_x + b_{15}h_y + c_{15} \\ a_{21}h_x + b_{21}h_y + c_{21} & a_{22}h_x + b_{22}h_y + c_{22} & \dots & a_{25}h_x + b_{25}h_y + c_{25} \\ \vdots & \vdots & \dots & \vdots \\ a_{61}h_x + b_{61}h_y + c_{61} & a_{62}h_x + b_{62}h_y + c_{62} & \dots & a_{65}h_x + b_{65}h_y + c_{45} \end{bmatrix} \begin{Bmatrix} g_{x0} \\ g_{z0} \\ w_{x0} \\ w_{z0} \\ 1 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{Bmatrix}. \tag{5.41}$$

This equation can be solved if we force the six 5×5 minors to be zero, so the system has rank four. They yield six polynomials of degree five in h_x and h_y. We can then collect by the powers of h_x to construct the 6×6 matrix equation

$$\begin{bmatrix} d_{10} & d_{11}(h_y) & d_{12}(h_y) & d_{13}(h_y) & d_{14}(h_y) & d_{15}(h_y) \\ d_{20} & d_{21}(h_y) & d_{22}(h_y) & d_{23}(h_y) & d_{24}(h_y) & d_{25}(h_y) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ d_{60} & d_{61}(h_y) & d_{62}(h_y) & d_{63}(h_y) & d_{64}(h_y) & d_{65}(h_y) \end{bmatrix} \begin{Bmatrix} h_x^5 \\ h_x^4 \\ \vdots \\ h_x \\ 1 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{Bmatrix}. \tag{5.42}$$

We impose $\det[D(h_y)] = 0$ to obtain a univariate polynomial of degree 15. The solutions of this polynomial for h_y allow us to compute the associated values h_x in

(5.42). These values are substituted into (5.41) to determine corresponding values for $\mathbf{g}^0$, and $\mathbf{w}^0$, thus, completing the solution of the design equations for the RRP chain.

Obtaining the univariate polynomial with a software such as Mathematica or Maple is easily done with the above procedure; however, for implementing the synthesis in a programming language, we use a less tedious procedure.

Manocha [49] provides a convenient way to find the values y such that $\det[D(y)] = 0$ by computing the eigenvalues of a matrix constructed from $[D(y)]$. The first step is to expand $[D(y)]$ as a matrix polynomial in y, that is

$$[D(y)] = [M_0] + [M_1]y + [M_2]y^2 + [M_3]y^3 + [M_4]y^4 + [M_5]y^5. \tag{5.43}$$

Notice that the structure of the coefficients $d_{ij}(y)$ forces the first i columns of the 6×6 matrices $[M_i]$ to be zero. This means only $[M_0]$ is non-singular.

We now transform $[D(y)]$ into a monic polynomial. However, because $[M_0]$ is the only invertable coefficient matrix, we must transform coordinates so $z = 1/y$, and define

$$[D(z)] = [M_0]z^5 + [M_1]z^4 + [M_2]z^3 + [M_3]z^2 + [M_4]z + [M_5]. \tag{5.44}$$

Multiply by $[M_0]^{-1}$ to obtain

$$[\bar{D}(z)] = [I]z^5 + [\bar{M}_1]z^4 + [\bar{M}_2]z^3 + [\bar{M}_3]z^2 + [\bar{M}_4]z + [\bar{M}_5], \tag{5.45}$$

where $[I]$ is the 6×6 identity matrix and $[\bar{M}_i] = [M_0]^{-1}[M_i]$.

The roots of $\det[\bar{D}(z)] = 0$ are obtained as the eigenvalues of the 30×30 matrix

$$[C] = \begin{bmatrix} 0 & [I] & 0 & 0 & 0 \\ 0 & 0 & [I] & 0 & 0 \\ 0 & 0 & 0 & [I] & 0 \\ 0 & 0 & 0 & 0 & [I] \\ -[\bar{M}_5] & -[\bar{M}_4] & -[\bar{M}_3] & -[\bar{M}_2] & -[\bar{M}_1] \end{bmatrix}. \tag{5.46}$$

This result and a general fomulation that does not rely on a monic structure for the matrix polynomial is discussed in detail in [50].

We solved this problem both using the resultant polynomial obtained from $\det[D(y)]$ and by computing the 15 non-zero eigenvalues of $[C]$ using Mathematica. The computations in both cases took less than 1 second and were identical. In completing the solution, we found that only four of the 15 roots yielded physical RRP chains. This combines with the three roots of the directions equations to yield 12 solutions.

Example

We present an example in which we want to design the RRP robot to reach the following four positions:

Table 5.7: Task positions for an RRP chain

Position	*Axis*	*Rotation*	*Translation*
position 1	$(1.0, 0.0, 0.0) + \epsilon(0.0, 0.0, 0.0)$	0^o	0
position 2	$(0.51, 0.73, -0.45) + \epsilon(-0.60, 1.58, 1.86)$	262.0^o	-0.08
position 3	$(-0.68, -0.55, -0.49) + \epsilon(0.91, 0.27, -1.54)$	123.1^o	0.48
position 4	$(-0.26, 0.87, 0.42) + \epsilon(0.09, 0.32, -0.62)$	27.9^o	-2.32

In this case we obtain twelve real roots, which are listed in Appendix A. One of

the obtained solutions is presented, as the joint axes in the first position, in Table 5.8. Figure 5.8 shows the robot while reaching each of the positions.

Table 5.8: The solution for the RRP chain

Joint Axis	*Direction*	*Moment*
G	$(0, -0.90, 0.44)$	$(-1.60, -0.60, -1.20)$
W	$(0.30, -0.83, -0.47)$	$(0.25, 0.27, -0.32)$
H	$(-0.35, 0.72, 0.59)$	$(-0.79, -0.75, 0.44])$

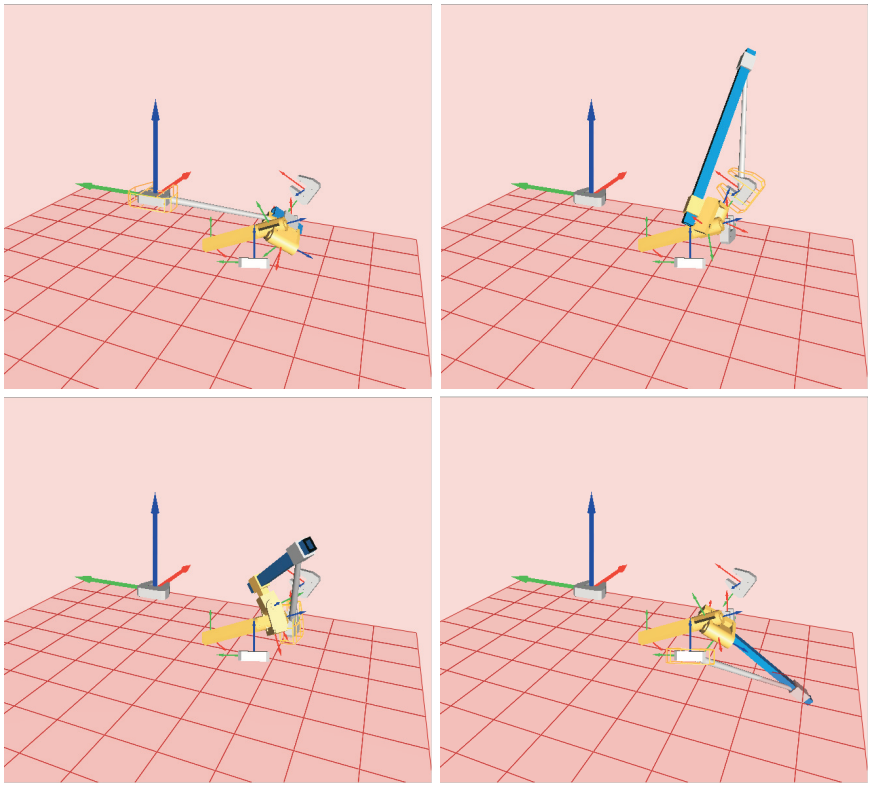

Figure 5.8: A solution RRP robot reaching four positions

5.3.3 The TP robot

The spatial TP robot can be seen as a special case of the RRP robot solved in Section 5.3.2, in which both revolute joints are forced to be perpendicular in direction and concurrent. It consists of a universal joint T with rotations θ_1 and θ_2 about perpendicular directions $\mathbf{g}_1$, $\mathbf{g}_2$ that intersect at a point $\mathbf{c}$, and a prismatic joint $\mathbf{h}$ that allows translation d along it, see Figure 5.9.

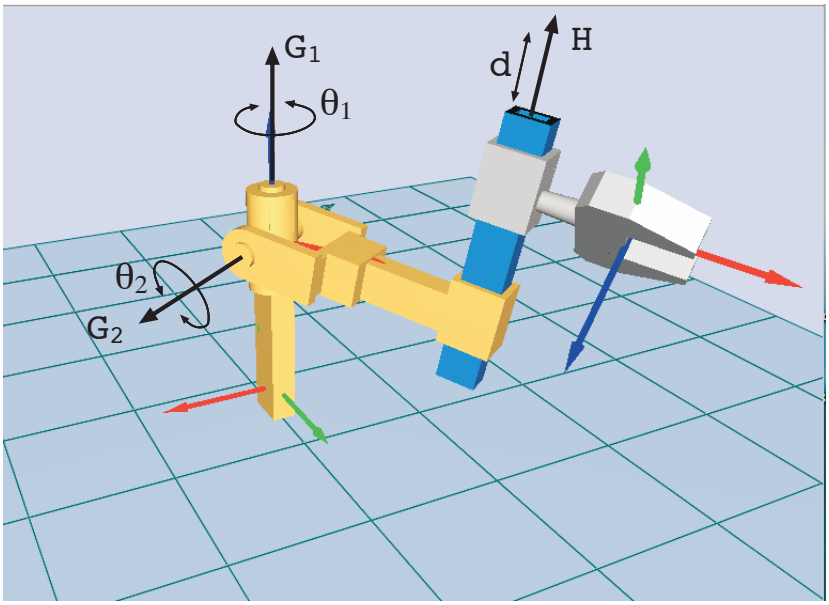

Figure 5.9: The TP robot

The design equations

The dual quaternion representation for the relative displacements,

$$\hat{Q}_{TP} = \hat{G}_1(\theta_1, 0)\hat{G}_2(\theta_2, 0))\hat{H}(0, d), \tag{5.47}$$

is expanded as $\hat{Q}_{TP} = Q^0 + \mathsf{Q}$, where the point is

$$\begin{aligned} Q^0 = &\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2} - \epsilon(\frac{d}{2}\sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}\mathsf{G}_1\cdot\mathsf{H} + \\ &\frac{d}{2}\cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}\mathsf{G}_2\cdot\mathsf{H} + \frac{d}{2}\sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}(\mathsf{G}_1\times\mathsf{G}_2)\cdot\mathsf{H}), \end{aligned} \tag{5.48}$$

and the vector

$$\begin{aligned} \mathsf{Q} = &\sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}\mathsf{G}_1 + \cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}\mathsf{G}_2 + \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}(\mathsf{G}_1\times\mathsf{G}_2) + \\ &\epsilon\frac{d}{2}(\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2}\mathsf{H} + \sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2}(\mathsf{G}_1\times\mathsf{H}) + \\ &\cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2}(\mathsf{G}_2\times\mathsf{H}) + \sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2}(\mathsf{G}_1\times\mathsf{G}_2)\times\mathsf{H}), \end{aligned} \tag{5.49}$$

and equated to the goal positions. In this case we obtain a count of $n_{max} = 3 + \frac{2}{3}$ by adding the two constraints $\mathbf{g}_1 \cdot \mathbf{g}_2 = 0$ and $\mathbf{g}_1 \cdot \mathbf{g}_2^0 + \mathbf{g}_1^0 \cdot \mathbf{g}_2 = 0$ to the counting for the RRP chain. For the orientations we obtain $n_R = 4$. If we add one structural constraint to either $\mathbf{g}_1$ or $\mathbf{g}_2$ and another one to $\mathbf{h}$ or to the intersection point $\mathbf{c}$, we obtain an equal count $n_R = 3$ and $n_{max} = 3$, and the design equations

$$\hat{G}_1(\theta_1^i, 0)\hat{G}_2(\theta_2^i, 0)\hat{H}(0, d^i) = \hat{P}^i, \quad i = 2, 3, \tag{5.50}$$

can be solved separately for the rotations.

Solving the design equations

We use identical methodology as in the RRP chain for eliminating the joint variables, with the advantage that now the submatrices are orthogonal and hence it is easier to calculate their inverses. Once we obtain the set of reduced design equations, we solve first for the directions of the rotation axes and use this result to solve for the rest of variables, following the same procedure as in the previous case. We can obtain up to 12 TP solutions.

Example

For this example we use the extra constraints $g_x = 0$ and $h_z = 0$ to solve for the three task positions of Table 5.9.

Table 5.9: Task positions for a TP chain

Position	*Axis*	*Rotation*	*Translation*
position 1	$(1.0, 0.0, 0.0) + \epsilon(0.0, 0.0, 0.0)$	0^o	0
position 2	$(-0.39, 0.10, 0.92), +\epsilon(-1.73, -3.03, -0.40)$	59.5^o	1.79
position 3	$(-0.38, 0.48, -0.79) + \epsilon(0.72, 0.24, -0.20)$	152.9^o	-2.61

We obtain 6 real solutions that are presented in Appendix A. Notice that there are only two different solutions for the structural variables that affect the translation only. Figure 5.10 and Table 5.10 show solution 2 reaching the task positions.

Table 5.10: The solutions for the TP chain

Joint Axis	*Direction*	*Moment*
G_1	$(0, -0.67, 0.74)$	$(5.01, -1.11, -1.01)$
G_2	$(-0.17, 0.73, 0.66)$	$(-6.83, -2.45, 0.92)$
H	$(0, 0.66, 0.75)$	$(-6.36, 0.36, -0.31)$

5.3.4 Spatial RRR robot

The spatial RRR robot is a three-degree-of-freedom robot consisting of three revolute axes which are not parallel nor concurrent: the fixed axis G of rotation θ and the rotations ϕ and ψ about the moving axes W and F. See Figure 5.11.

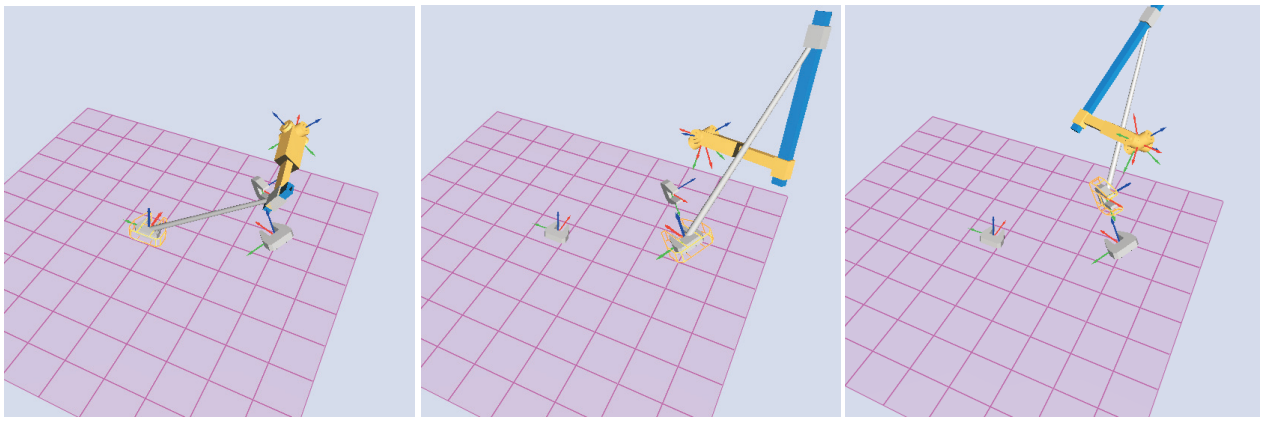

Figure 5.10: A solution for the TP robot reaching three positions

A brief history

Design equations for the RRR robot were stated in [89] using the loop equation method, and partially solved using optimization for four task positions. Lee and Mavroidis [46] presented first a partial solution for three task positions using homotopy continuation, and posteriorly [45] they solved the complete problem, for five precision positions, using interval analysis, and obtained 26 roots, 20 of them outside of the domain. In both cases they formulated the design equations using the kinematics equations in matrix form and solved for the Denavit-Hartenberg parameters.

The design equations

The dual quaternion representation for the relative displacements of the chain is given by

$$\hat{Q}_{RRR} = \hat{G}(\theta, 0)\hat{W}(\phi, 0)\hat{F}(\psi, 0). \tag{5.51}$$

When applying the dual quaternion product we obtain the expression $\hat{Q}_{RRR} =$

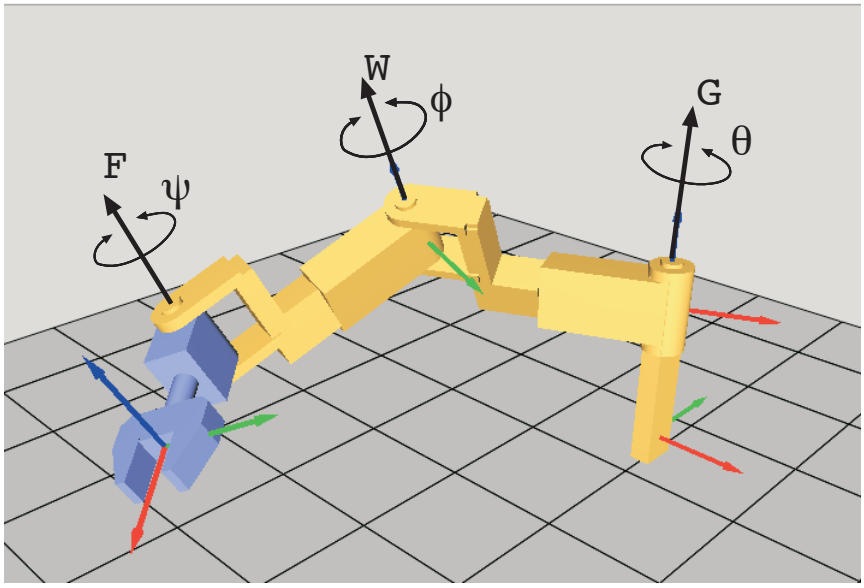

Figure 5.11: The spatial RRR robot

$Q^0 + \mathsf{Q}$, where the dual point Q^0 and the dual vector Q are

$$
\begin{aligned}
Q^0 &= \cos\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2}\mathsf{G}\cdot\mathsf{W} - \cos\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}\mathsf{W}\cdot\mathsf{F} - \\
&\quad \sin\frac{\theta}{2}\cos\frac{\phi}{2}\sin\frac{\psi}{2}\mathsf{G}\cdot\mathsf{F} - \sin\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}(\mathsf{G}\times\mathsf{W})\cdot\mathsf{F}, \\
\mathsf{Q} &= \sin\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2}\mathsf{G} + \cos\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2}\mathsf{W} + \cos\frac{\theta}{2}\cos\frac{\phi}{2}\sin\frac{\psi}{2}\mathsf{F} + \\
&\quad \sin\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2}(\mathsf{G}\times\mathsf{W}) + \sin\frac{\theta}{2}\cos\frac{\phi}{2}\sin\frac{\psi}{2}(\mathsf{G}\times\mathsf{F}) + \cos\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}(\mathsf{W}\times\mathsf{F}) - \\
&\quad \sin\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}(\mathsf{G}\cdot\mathsf{W})\mathsf{F} + \sin\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}(\mathsf{G}\times\mathsf{W})\times\mathsf{F}.
\end{aligned}
\tag{5.52}
$$

We notice that the RRR robot can reach any orientation, and using Eq.(3.16) we obtain $n_{max} = 5$ positions. The parameterized design equations for $n = 5$ positions,

$$
\hat{Q}_{RRR}(\theta^i, \phi^i, \psi^i) - \hat{P}^i = \vec{0}, \quad i = 2, \ldots, 5, \tag{5.53}
$$

are solved to obtain a finite number of solutions.

Solving the design equations

Following the methodology described in Section 3.8.1, we define the eight-dimensional vector containing the joint variables,

$$\hat{V}(\theta,\phi,\psi) = \begin{Bmatrix} \cos\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2} \\ \sin\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2} \\ \cos\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2} \\ \cos\frac{\theta}{2}\cos\frac{\phi}{2}\sin\frac{\psi}{2} \\ \sin\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2} \\ \sin\frac{\theta}{2}\cos\frac{\phi}{2}\sin\frac{\psi}{2} \\ \cos\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2} \\ \sin\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2} \end{Bmatrix}, \tag{5.54}$$

and create the linear system

$$\begin{bmatrix} A & \vdots & B \\ \dots & \vdots & \dots \\ C & \vdots & D \end{bmatrix} \hat{V}(\theta,\phi,\psi) = \hat{P}. \tag{5.55}$$

where the matrices are

$$\begin{bmatrix} A \\ \dots \\ C \end{bmatrix} = \begin{bmatrix} \vec{0} & \mathbf{g} & \mathbf{w} & \mathbf{f} \\ 1 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots \\ \vec{0} & \mathbf{g}^0 & \mathbf{w}^0 & \mathbf{f}^0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.56}$$

$$\begin{bmatrix} B \\ \dots \\ D \end{bmatrix} = \begin{bmatrix} \mathbf{g}\times\mathbf{w} & \mathbf{g}\times\mathbf{f} & \mathbf{w}\times\mathbf{f} & (\mathbf{g}\times\mathbf{w})\times\mathbf{f} - \mathbf{g}\cdot\mathbf{w}\mathbf{f} \\ -\mathbf{g}\cdot\mathbf{w} & -\mathbf{g}\cdot\mathbf{f} & -\mathbf{w}\cdot\mathbf{f} & -(\mathbf{g}\times\mathbf{w})\cdot\mathbf{f} \\ \dots & \dots & \dots & \dots \\ (\mathbf{g}\times\mathbf{w})^0 & (\mathbf{g}\times\mathbf{f})^0 & (\mathbf{w}\times\mathbf{f})^0 & ((\mathbf{g}\times\mathbf{w})\times\mathbf{f})^0 - (\mathbf{g}\cdot\mathbf{w})^0\mathbf{f} - (\mathbf{g}\cdot\mathbf{w})\mathbf{f}^0 \\ -(\mathbf{g}\cdot\mathbf{w})^0 & -(\mathbf{g}\cdot\mathbf{f})^0 & -(\mathbf{w}\cdot\mathbf{f})^0 & -((\mathbf{g}\times\mathbf{w})^0)\cdot\mathbf{f} - (\mathbf{g}\times\mathbf{w})\cdot\mathbf{f}^0 \end{bmatrix} \tag{5.57}$$

and $(\mathbf{g}\times\mathbf{w})^0 = \mathbf{g}\times\mathbf{w}^0 + \mathbf{g}^0\times\mathbf{w}$, the dual part of the cross product, and same notation is used for the rest of the terms.

We use Cramer's rule to solve for the joint variables. To compute the determinant, we use Schur complement applied to the A, B, C, D block matrices.

The reduced design equations are obtained from the three independent relations among components of the vector $\hat{V}$,

$$\begin{aligned} \mathcal{R}_1 \quad & \frac{\cos\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2}}{\sin\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2}} = \frac{\cos\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}}{\sin\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}}, \\ \mathcal{R}_2 \quad & \frac{\cos\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2}}{\cos\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2}} = \frac{\sin\frac{\theta}{2}\cos\frac{\phi}{2}\sin\frac{\psi}{2}}{\sin\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}}, \\ \mathcal{R}_3 \quad & \frac{\cos\frac{\theta}{2}\cos\frac{\phi}{2}\cos\frac{\psi}{2}}{\cos\frac{\theta}{2}\cos\frac{\phi}{2}\sin\frac{\psi}{2}} = \frac{\sin\frac{\theta}{2}\sin\frac{\phi}{2}\cos\frac{\psi}{2}}{\sin\frac{\theta}{2}\sin\frac{\phi}{2}\sin\frac{\psi}{2}}. \end{aligned} \tag{5.58}$$

The set of reduced design equations,

$$\begin{aligned} & \{\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3\}^{\mathrm{i}}, \quad \mathrm{i} = 2, \dots, 5, \\ & \mathbf{g}\cdot\mathbf{g} = 1, \mathbf{w}\cdot\mathbf{w} = 1, \mathbf{f}\cdot\mathbf{f} = 1, \\ & \mathbf{g}\cdot\mathbf{g}^0 = 0, \mathbf{w}\cdot\mathbf{w}^0 = 0, \mathbf{f}\cdot\mathbf{f}^0 = 0, \end{aligned} \tag{5.59}$$

consists of 18 equations in 18 unknowns corresponding to the three joint axes G, W and F. The equations are solved numerically.

Numerical Example

We solve both the parameterized and the reduced equations. We use the Newton-Raphson solver *fsolve* in Matlab with Levenberg-Marquardt strategy and random initial conditions.

Table 5.11: Task positions for an RRR chain

Position	*Axis*	*Rotation*	*Translation*
position 1	$(1.0, 0.0, 0.0) + \epsilon(0.0, 0.0, 0.0)$	0^o	0
position 2	$(0.98, -0.14, 0.16), +\epsilon(-0.51, -1.81, 1.51)$	100.4^o	2.21
position 3	$(0.28, -0.46, -0.84) + \epsilon(-0.01, -2.18, 1.20)$	134.4^o	-1.20
position 4	$(0.44, -0.31, 0.84) + \epsilon(-2.61, -1.41, 0.86)$	161.4^o	-0.68
position 5	$(-0.08, -0.01, -0.99) + \epsilon(1.09, -0.43, -0.09)$	162.0^o	1.85

For the actual example, we run the parameterized equations around 6000 times, each run consisting of a maximum of 4000 evaluations. It took the computer, a Power PC with processor G4 at 500MHz, around 5 seconds per run when the initial value converged to a solution; otherwise, it took about 12 seconds to compute the 4000 evaluations. The total time was of 18 hours. Out of the 6000 runs, it converged to a root in 1190 of them. Notice that roughly one of each six initial conditions converged to a root; however, many were repeated due to the random initial conditions and only 133 were different. Out of those, we obtain 19 real solutions for the RRR robot.

The same task positions were used to solve, also numerically, the reduced design equations. We use the same numerical solver and strategy.

The task positions for this example are presented in Table 5.11, and one of the 19

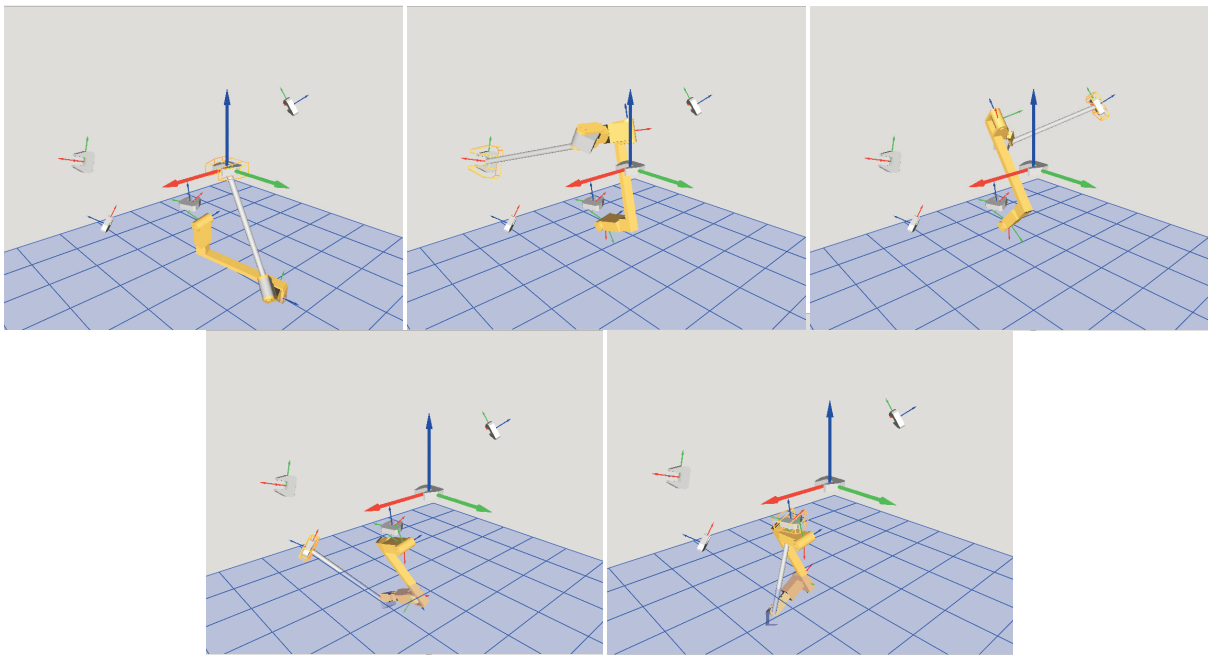

Figure 5.12: A spatial RRR robot reaching five positions

solutions is shown in Figure 5.12 and Table 5.12. The rest of solutions can be found in Appendix A.

Table 5.12: One solution for the RRR chain

Joint Axis	*Direction*	*Moment*
G	$(0, -0.67, 0.74)$	$(-0.06, -0.90, -0.82)$
W	$(-0.17, 0.73, 0.66)$	$(-4.19, -1.33, 0.38)$
F	$(0, -0.49, 0.87)$	$(-1.34, 0.66, 0.37)$

5.4 Synthesis of Four-degree-of-freedom Robots

The group of four-degree-of-freedom robots presents a greater variety and some of them have not been solved yet. This group includes robots with an S-joint, whose

design equations are easy to state just by defining the geometric constraints of the chain, see [12] and more recently [27], but present some particularities when defining them using the dual quaternion formulation, see Chapter 4. The main three groups of kinematic chains have two (RRPP), three (RRRP), and four (RRRR) revolute joints; the rest are particular cases.

5.4.1 Spatial RPRP robot

The RPRP robot is a case of orientation-limited robot. Its design equations are solved both in parameterized form and in reduced form, for five complete positions and four additional translations. In the following section, four constraints are defined on the prismatic joints to define an RPC robot. The reduced design equations for the RPC robot are created and solved to obtain closed algebraic solutions.

The spatial RPRP robot is a four-degree-of-freedom robot. The fixed axis $\mathsf{G} = \mathbf{g} + \epsilon\mathbf{g}^0$ allows rotation of angle θ about it. This is followed along the chain by a translation d along an arbitrary direction $\mathbf{h}$, a rotation of angle ϕ about an arbitrary axis $\mathsf{W} = \mathbf{w} + \epsilon\mathbf{w}^0$, and a translation b along an arbitrary direction $\mathbf{u}$, see Figure 5.13.

A brief history

The RPRP robot was first solved in [73] using the dual quaternion methodology. For the general case no closed algebraic solution was presented, even though it is possible to solve the problem using resultant techniques. The CC robot can be considered as

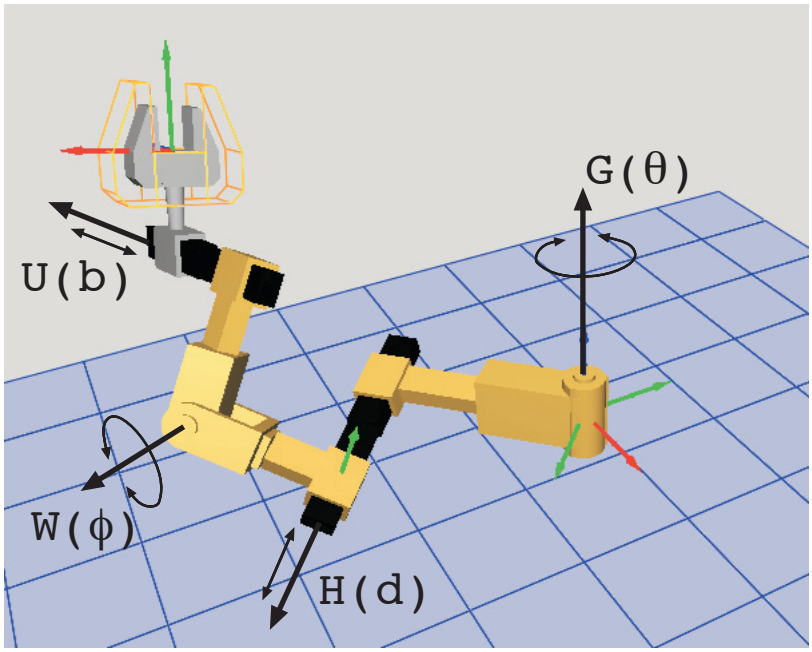

Figure 5.13: The RPRP robot

an RPRP robot in which there are constraints defining the prismatic joints. It is one of the dyads that were successfully solved early by imposing geometric constraints, see [96], and [58] for a complete algebraic solution. From the point of view of the dual quaternion synthesis, the CC robot is alike other RPRP robots with defined prismatic joints. In [73], we solved the perpendicular RPC robot (an RPC robot with its first prismatic joint perpendicular to the revolute joints), and obtained a closed algebraic solution that yields 6 complex solutions.

The design equations

The dual quaternion kinematics equations are obtained by composing the dual quaternions that represent each joint axis,

$$\hat{Q}_{RPRP}(\theta, d, \phi, b) = \hat{G}(\theta, 0)\hat{H}(0, d)\hat{W}(\phi, 0)\hat{U}(0, b), \tag{5.60}$$

which expand to the dual scalar and the dual vector, $\hat{Q}_{RPC} = \hat{Q}_0 + \mathsf{Q}$,

$$\begin{aligned}\hat{Q}_0 = {} & c\frac{\theta}{2}c\frac{\phi}{2} - \mathbf{g}\cdot\mathbf{w}s\frac{\theta}{2}s\frac{\phi}{2} + \epsilon(-(\frac{d}{2}\mathbf{g}\cdot\mathbf{h} + \frac{b}{2}\mathbf{g}\cdot\mathbf{u})s\frac{\theta}{2}c\frac{\phi}{2} - \\ & (\frac{b}{2}\mathbf{w}\cdot\mathbf{u} + \frac{d}{2}\mathbf{h}\cdot\mathbf{w})c\frac{\theta}{2}s\frac{\phi}{2} - (\frac{d}{2}(\mathbf{g}\times\mathbf{h})\cdot\mathbf{w} + \frac{b}{2}(\mathbf{g}\times\mathbf{w})\cdot\mathbf{u} + \mathbf{g}_0\cdot\mathbf{w} + \mathbf{g}\cdot\mathbf{w}_0)s\frac{\theta}{2}s\frac{\phi}{2}),\end{aligned} \tag{5.61}$$

$$\begin{aligned}\mathsf{Q} = {} & \mathbf{g}s\frac{\theta}{2}c\frac{\phi}{2} + \mathbf{w}c\frac{\theta}{2}s\frac{\phi}{2} + \mathbf{g}\times\mathbf{w}s\frac{\theta}{2}s\frac{\phi}{2} + \epsilon((\mathbf{g}_0 + \frac{b}{2}\mathbf{g}\times\mathbf{u} + \\ & \frac{d}{2}\mathbf{g}\times\mathbf{h})s\frac{\theta}{2}c\frac{\phi}{2} + (\mathbf{w}_0 + \frac{d}{2}\mathbf{h}\times\mathbf{w} + \frac{b}{2}\mathbf{w}\times\mathbf{u})c\frac{\theta}{2}s\frac{\phi}{2} + (\frac{b}{2}\mathbf{u} + \frac{d}{2}\mathbf{h})c\frac{\theta}{2}c\frac{\phi}{2} + \\ & (\mathbf{g}_0\times\mathbf{w} + \mathbf{g}\times\mathbf{w}_0 + \frac{b}{2}((\mathbf{g}\times\mathbf{w})\times\mathbf{u} - (\mathbf{g}\cdot\mathbf{w})\mathbf{u}) + \frac{d}{2}((\mathbf{g}\times\mathbf{h})\times\mathbf{w} - (\mathbf{g}\cdot\mathbf{h})\mathbf{w})s\frac{\theta}{2}s\frac{\phi}{2}),\end{aligned} \tag{5.62}$$

where c and s stand for cosine and sine respectively.

We use the formulas in Eqs.(3.16), (3.17) and (3.19) with $r = 2$, $t = 2$, to count the maximum number of goal dual transformations that we can specify. We obtain $n_{max} = 7$ and $n_R = 5$; the maximum number of complete positions we can define is five, but in addition $n_T = 9$ and we can specify four more arbitrary translations whose orientations must belong to the workspace of the RPRP robot.

We create the design equations for the complete positions by equating the expres-

sions in Eq.(5.60) to the goal dual quaternions,

$$\hat{Q}_{RPRP}(\theta^i, d^i, \phi^i, b^i) = \hat{P}^i, \quad i = 2, 3, 4, 5, \tag{5.63}$$

and the design equations for the partially specified positions by equating the dual components of Eq.(5.60), Q^0_{RPRP}, to the dual components of the four extra dual quaternions,

$$\hat{Q}^0_{RPRP}(\theta^i, d^i, \phi^i, b^i) = \hat{P}^i_0, \quad i = 6, 7, 8, 9. \tag{5.64}$$

Here we need to remark that the dual components in Eq.(5.64) are defined for a chosen orientation in the workspace, and considering that, we can solve to define the robot. If we want to do the opposite, that is, define a task translation and adjust the rotation to whatever is needed, then we can define as many extra translations as we want, because the robot is not constrained if we consider translations only.

Solving the design equations

We solve separately for the directions of the revolute joint axes $\mathbf{g}$ and $\mathbf{w}$ using the real part of each dual quaternion equality in Eq.(5.63), applying the elimination method shown in [54] to obtain six complex solutions.

We now use the workspace of orientations of the robot for each of the real solutions to specify orientations for the rest of the task positions. We can pick any orientation belonging to the workspace to construct the four additional dual quaternions with arbitrary task translations.

Table 5.13: Real solutions for the directons of the revolute joints of the RPRP robot.

Joint Axis	*Direction*
$\mathbf{g}$	$(-0.48, -0.78, 0.39)$
$\mathbf{w}$	$(0.33, -0.23, 0.91)$
$\mathbf{g}$	$(0.04, 0.05, 0.99)$
$\mathbf{w}$	$(0.70, 0.48, 0.53)$

The dual design equations, once we substitute the values of $\mathbf{g}$ and $\mathbf{w}$, contain dual parts only, that is,

$$Q^{0i} = \begin{Bmatrix} q_{x0} \\ q_{y0} \\ q_{z0} \end{Bmatrix}^i = \begin{Bmatrix} p_{x0} \\ p_{y0} \\ p_{z0} \end{Bmatrix}^i, \quad i = 2, \ldots 9,$$

$$\mathbf{h} \cdot \mathbf{h} = 1, \quad \mathbf{u} \cdot \mathbf{u} = 1,$$

$$\mathbf{g} \cdot \mathbf{g}^0 = 0, \quad \mathbf{w} \cdot \mathbf{w}^0 = 0. \tag{5.65}$$

We solve the dual part of the design equations numerically.

Example

We define five complete task dual quaternions randomly. They are labeled 1 to 5 in Table 5.14 and Figure 5.14.

Solve algebraically for the directions of the revolute joints $\mathbf{g}$ and $\mathbf{w}$ to obtain two real solutions only, showed in Table 5.13 . See Appendix A for the complete set of solutions.

We can now define the workspace of orientations of the robot for each of the real

Table 5.14: The five complete plus four translational goal positions for an RPRP robot.

Position	*Axis*	*Rot.*	*Trans.*
1	$(1.0, 0.0, 0.0; 0.0, 0.0, 0.0)$	0	0
2	$(0.98, -0.14, 0.16; -0.51, -1.81, 1.51)$	1.75	2.21
3	$(0.28, -0.46, -0.84; -0.01, -2.18, 1.20)$	2.34	-1.20
4	$(0.44, -0.31, 0.84; -2.61, -1.41, 0.86)$	2.82	-0.68
5	$(-0.08, -0.01, -0.99; 1.09, -0.43, -0.09)$	2.83	1.85
6	$(0.35, 0.69, 0.63; 0.44, -0.44, 0.23)$	2.92	5.72
7	$(-0.61, -0.64, -0.46; -0.51, -0.98, 2.04)$	2.16	4.17
8	$(-0.09, 0.08, 0.99; -1.62, -0.30, -0.12)$	4.71	-2.48
9	$(0.82, 0.37, 0.44; 1.22, -1.34, -1.12)$	1.74	-0.88

solutions. For this example we used solution 2 of Table 5.13. We pick orientations belonging to the workspace to construct the four additional dual quaternions with arbitrary translations. These are shown in Table 5.14.

If we choose to work with the dual reduced design equations, we need to solve a set of eight multi-linear equations in $\mathbf{h}$, $\mathbf{u}$, $\mathbf{g}^0$ and $\mathbf{w}^0$ of total degree 6561. Because of the multi-linearity, we can reduce the bound for the number of roots to a multi-linear Bezout number of 70. We solve these equations using polynomial homotopy continuation [60]. For this purpose we use the softare PHC developed by Verschelde [104]. We use a random linear product start system with 1120 start solutions. It took the program 44 minutes in a Power PC at 700 MHz to find the solutions for each of the two sets of equations. (One set for each real solution of $\mathbf{g}$, $\mathbf{w}$). We obtained

Table 5.15: An RPRP robot that reaches 5 complete positions plus 4 translations.

Joint Axis	*Direction*	*Moment*
G	$(0.04, 0.05, 0.99)$	$(1.43, -0.09, -0.06)$
$\mathbf{h}$	$(-0.33, -0.45, 0.83)$	
W	$(0.70, 0.48, 0.53)$	$(1.85, -1.40, -1.16)$
$\mathbf{u}$	$(0.30, 0.84, 0.44)$	

a total of 152 solutions for the first case, out of which only 21 are regular and, of those, 7 are real roots. For the second set, we obtained 153 solutions, with again 21 regular solutions and 5 real roots. The real results are presented in Appendix A. In Table 5.15 and Figure 5.14 we present solution 2 of the second set reaching the nine positions.

We also solved the design equations numerically by using the *fsolve* function on Matlab, with a Levenberg-Marquardt algorithm and using random initial conditions. For the dual parameterized equations we performed 1000 runs of 4000 iterations per run. Each run took approximately 14 seconds, with a total time of about four hours on a Mac G4 at 733 MHz. We obtained one solution. For the reduced equations, we use the same numerical solver also for 1000 runs of 3000 iterations each. One iteration took approximately 21 seconds, and the total time was about six hours. We obtained 3 different solutions.

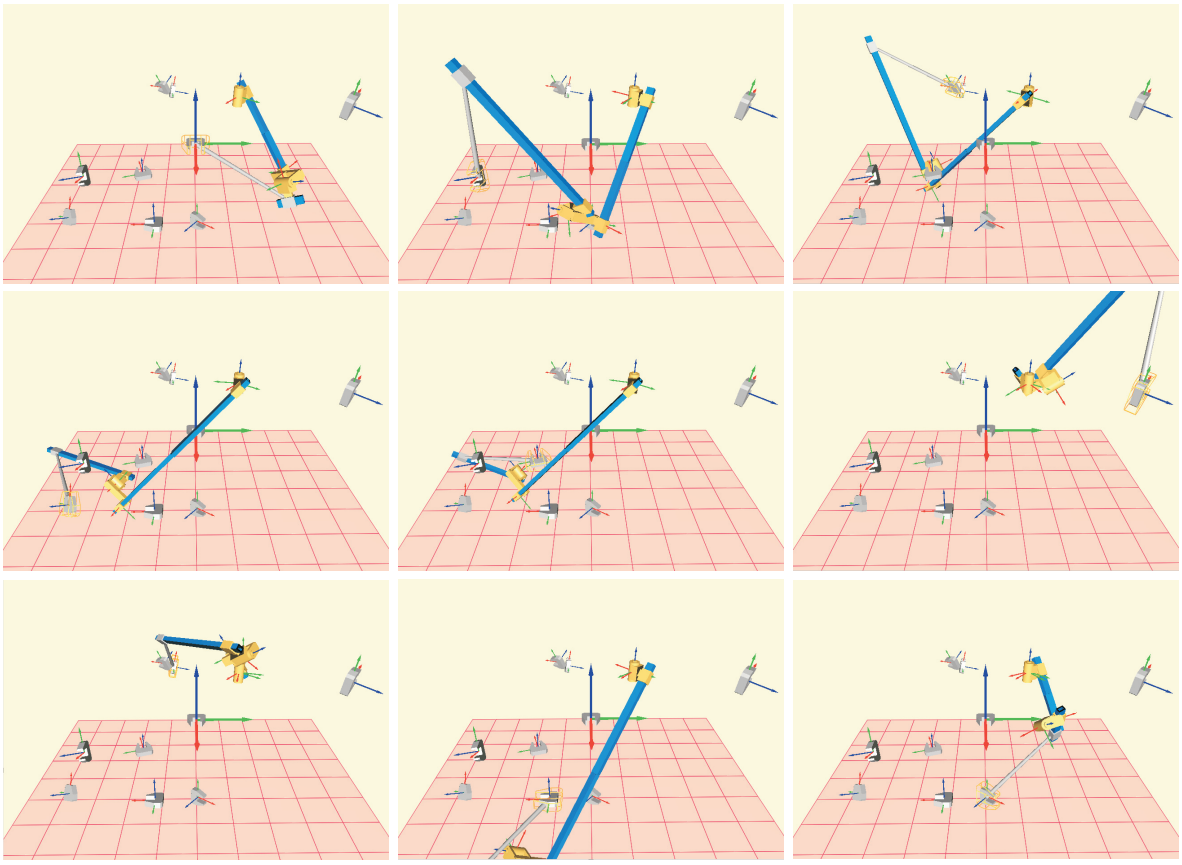

Figure 5.14: A solution for the RPRP robot reaching the task positions

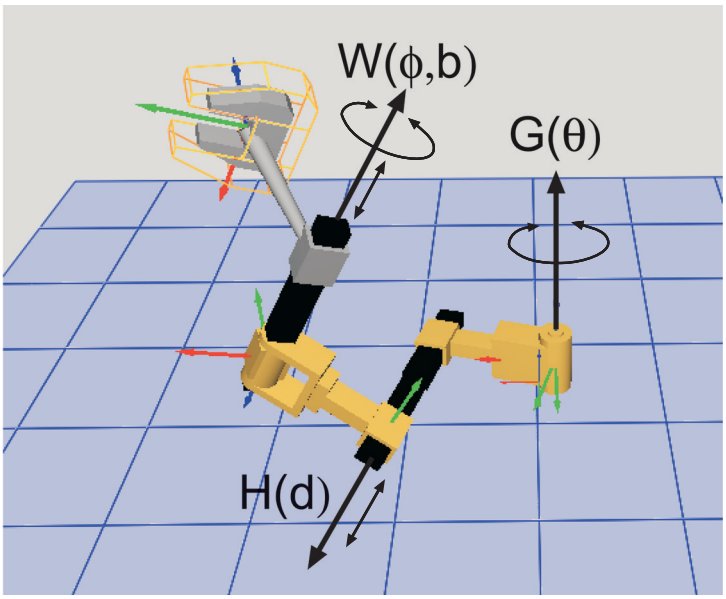

Figure 5.15: The RPC robot

5.4.2 The RPC robot

The RPC robot is a special case of the spatial RPRP robot presented in Section 5.4.1. If we impose as constraints that the last prismatic direction must be equal to the direction of the previous revolute joint, we obtain the RPC robot of Figure 5.15. Notice that the P and R joints that create the cylindrical joint need not be coincident.

According to the counting from previous section, we need to add four constraints in order to be able to solve for five complete positions. We define the constraints

$$\mathbf{g} \cdot \mathbf{h} = 0 \quad , \quad \mathbf{w} \cdot \mathbf{h} = 0.$$
$$\mathbf{w} \times \mathbf{u} = \vec{0}. \tag{5.66}$$

that make the first prismatic joint perpendicular to both revolute joints, and the last pair R-P joints into a cylindrical joint.

A brief history

Sandor [90] used the loop closure equations to state the design equations for the RPC robot, and partially solved the problem for 3 task positions, fixing some design variables and using objective functions to improve the performance. The RPC robot was completely solved in [73], together with the general RPRP robot, using the dual quaternion methodology. A closed algebraic solution was found by solving separately the orientation part from the rest.

The design equations

The dual quaternion kinematics equations,

$$\hat{Q}_{RPC}(\theta, d, \phi, b) = \hat{G}(\theta, 0)\hat{H}(0, d)\hat{W}(\phi, b), \tag{5.67}$$

are expanded and simplified by making $\mathbf{u} = \mathbf{w}$ and $\mathbf{g} \cdot \mathbf{h} = 0, \mathbf{w} \cdot \mathbf{h} = 0$ in Eq.(5.61) and Eq.(5.62). If we compute the number of complete positions that we can design for, with $r = 2$, $t = 2$, but considering the constraints in Eq. (5.66), we obtain $n_{max} = 5$ complete spatial goal positions, same value that we obtain for n_R. The five task positions are equated to the kinematics equations to obtain parameterized set of design equations,

$$\hat{Q}_{RPC}(\theta^i, d^i, \phi^i, b^i) = \hat{P}^i, \quad i = 2, 3, 4, 5. \tag{5.68}$$

Solving the design equations

We reduce the system by solving linearly for θ, ϕ, d as in Eqs.(3.51) and (3.52). We invert the matrix as in Eq.(3.54), where the submatrices are

$$\left[A^{-1}\right] = \frac{1}{(\mathbf{g}\times\mathbf{w})\cdot(\mathbf{g}\times\mathbf{w})} \begin{bmatrix} \mathbf{g}-(\mathbf{g}\cdot\mathbf{w})\mathbf{w} & 0 \\ \mathbf{w}-(\mathbf{g}\cdot\mathbf{w})\mathbf{g} & 0 \\ \mathbf{g}\times\mathbf{w} & 0 \\ (\mathbf{g}\cdot\mathbf{w})\mathbf{g}\times\mathbf{w} & (\mathbf{g}\times\mathbf{w})\cdot(\mathbf{g}\times\mathbf{w}) \end{bmatrix} \tag{5.69}$$

and, using the constraints we defined,

$$\left[C^{-1}\right] = \frac{1}{\mathbf{h}\cdot(\mathbf{g}\times\mathbf{w})} \begin{bmatrix} -\mathbf{g} & 0 \\ -\mathbf{w} & 0 \\ (\mathbf{h}\cdot(\mathbf{g}\times\mathbf{w}))\mathbf{h} & -\mathbf{g}\cdot\mathbf{w} \\ \vec{0} & 1 \end{bmatrix}. \tag{5.70}$$

The solution for the rotational components is the same for any other orientation-limited 2R robot, see for instance Eq.(5.18).

The solution for the translational components corresponding to the dual part,

$$\begin{aligned}
\frac{d}{2}\cos\frac{\theta}{2}\sin\frac{\phi}{2} &= \frac{b}{2}\left(\frac{(p_w\mathbf{g}\cdot\mathbf{w}-\mathbf{g}\times\mathbf{w}\cdot\mathbf{p})}{\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}}\right)+ \\
&\quad \frac{\mathbf{g}\cdot\mathbf{w}^0(\mathbf{w}\cdot\mathbf{p}-\mathbf{g}\cdot\mathbf{w}\mathbf{g}\cdot\mathbf{p})-\mathbf{g}^0\cdot\mathbf{g}\times\mathbf{w}\mathbf{p}\cdot\mathbf{g}\times\mathbf{w}-\mathbf{g}\cdot\mathbf{p}^0(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})}{\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})} \\
\frac{d}{2}\sin\frac{\theta}{2}\cos\frac{\phi}{2} &= \frac{b}{2}\left(\frac{p_w}{\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}}\right)+ \\
&\quad \frac{\mathbf{g}^0\cdot\mathbf{w}(\mathbf{g}\cdot\mathbf{p}-\mathbf{g}\cdot\mathbf{w}\mathbf{w}\cdot\mathbf{p})-\mathbf{w}^0\cdot\mathbf{g}\times\mathbf{w}\mathbf{p}\cdot\mathbf{g}\times\mathbf{w}-\mathbf{w}\cdot\mathbf{p}^0(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})}{\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})} \\
\frac{d}{2}\cos\frac{\theta}{2}\cos\frac{\phi}{2} &= \frac{b}{2}\left(\frac{\mathbf{g}\cdot\mathbf{p}\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}}{(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})}\right)+\frac{\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}(\mathbf{g}\cdot\mathbf{p}(\mathbf{g}\cdot\mathbf{w}\mathbf{h}\cdot\mathbf{w}^0-\mathbf{h}\cdot\mathbf{g}^0)}{\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})}+ \\
&\quad \frac{\mathbf{w}\cdot\mathbf{p}(\mathbf{g}\cdot\mathbf{w}\mathbf{h}\cdot\mathbf{g}^0-\mathbf{h}\cdot\mathbf{w}^0)+(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})(\mathbf{h}\cdot\mathbf{p}^0-\mathbf{g}\cdot\mathbf{w}p_{w0})}{\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})} \\
\frac{d}{2}\sin\frac{\theta}{2}\sin\frac{\phi}{2} &= \frac{b}{2}\left(\frac{\mathbf{w}\cdot\mathbf{p}}{\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}}\right)+\frac{\mathbf{p}\cdot\mathbf{g}\times\mathbf{w}(\mathbf{g}^0\cdot\mathbf{w}+\mathbf{g}\cdot\mathbf{w}^0)+p_{w0}(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})}{\mathbf{h}\cdot\mathbf{g}\times\mathbf{w}(\mathbf{g}\times\mathbf{w}\cdot\mathbf{g}\times\mathbf{w})},
\end{aligned} \tag{5.71}$$

are a function of the joint variable b.

To create the reduced design equations, we impose the conditions of Eqs.(3.57) and (3.58). The first one gives the direction equation $\mathcal{R}$ in Eq.(4.19), and the other two become two linear equations in b.

If we denote them $\mathcal{L}_1 : A_1\frac{b}{2}+B_1=0$, $\mathcal{L}_2 : A_2\frac{b}{2}+B_2=0$, we create the second-step reduced design equation by equating both solutions for b,

$$\mathcal{M} \quad : \quad \frac{B_1}{A_1}=\frac{B_2}{A_2}. \tag{5.72}$$

These two reduced design equations, plus the set of Plücker and extra constraints,

form the final set of 15 reduced equations in 15 parameters,

$$\{\mathcal{R}, \mathcal{M}\}^i, \quad i = 1, \ldots 4,$$

$$\mathbf{g} \cdot \mathbf{g} = 1, \quad \mathbf{g} \cdot \mathbf{g}_0 = 0,$$

$$\mathbf{w} \cdot \mathbf{w} = 1, \quad \mathbf{w} \cdot \mathbf{w}_0 = 0,$$

$$\mathbf{h} \cdot \mathbf{h} = 1, \quad \mathbf{h} \cdot \mathbf{g} = 0, \quad \mathbf{h} \cdot \mathbf{w} = 0. \tag{5.73}$$

To obtain a closed solution, we solve separately for the set of orientation equations $\{\mathcal{R}\}^i$ as presented in 5.3.2. That procedure gives a maximum of six complex roots. Once we have $\mathbf{g}$ and $\mathbf{w}$, the prismatic direction $\mathbf{h}$ is completely specified by the two constrains in Eq.(5.66). The translation equations $\{\mathcal{M}\}^i$ are linear in the moment components of the revolute joints, $\mathbf{g}^0$ and $\mathbf{w}^0$. Due to this linearity, there exist a maximum of six complex solutions for the five-position synthesis of the RPC robot with the given constraints.

For the design example, the goal displacements shown on Table 5.16 have been randomly generated. The algebraic solution of the reduced design equations yields six solutions, out of which two are real, see Appendix A. Figure 5.16 shows one of the real solutions obtained while reaching positions 1 to 5.

For comparison purposes, we solved also the parameterized design equations numerically using the *fsolve* Levenberg-Marquardt algorithm in Matlab. We did 100 runs starting with random initial values, each of the runs consisting of 3000 iterations. Each run took about 9 seconds. Overall the process took about 15 minutes on a Mac G4 at 733 MHz. We obtain the two real solutions repeated 7 and 6 times.

Table 5.16: The task positions for an RPC robot.

Axis	*Rot.*	*Trans.*
$(1.0, 0.0, 0.0; 0.0, 0.0, 0.0)$	0	0
$(-0.15, 0.36, 0.92; -0.98, -1.93, 0.60)$	1.49	-1.05
$(-0.64, -0.53, 0.55; 0.53, 1.17, 1.76)$	0.66	1.19
$(-0.43, -0.47, 0.77; 2.35, -0.17, 1.20)$	2.94	-0.29
$(0.48, 0.21, 0.85; 1.52, -0.33, -0.78)$	0.67	1.09

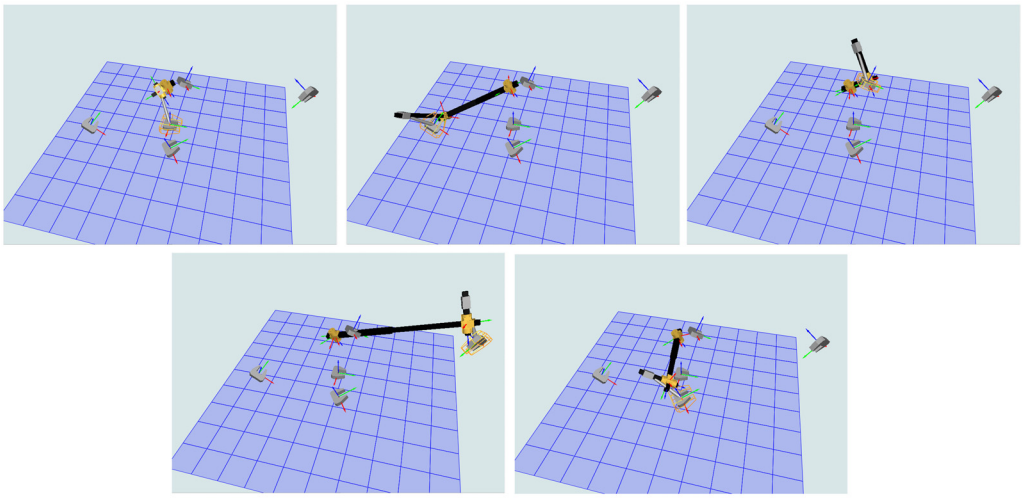

Figure 5.16: One RPC solution reaching positions 1,2,3,4 and 5.

5.4.3 Spatial PRRR robot

The spatial PRRR is a four-degree-of-freedom robot consisting of a prismatic joint H and three revolute joints G, W, F, which are arbitrarly located in space, see Figure 5.17.

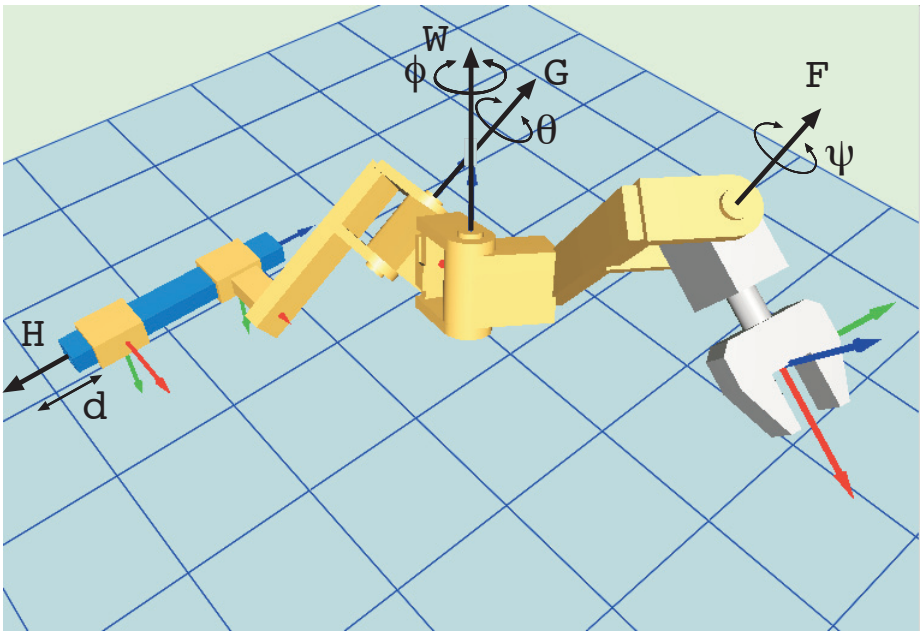

Figure 5.17: The spatial PRRR robot

A brief history

The general PRRR robot or any of the topologies formed by permuting its axes (RPRR, RRPR, RRRP) has not been solved so far except in this work. A particular case, when the first revolute and the prismatic joint directions are the same, creating a CRR robot, was solved in [74]. Another particular case, when two of the revolute joints intersect and form a T joint, was solved in [72]. A well-known case is obtained when all three revolute joints intersect at a point, creating a spherical joint. The PS robot has been solved stating the geometric constraint equations.

The design equations

The dual quaternion kinematics equations for the PRRR robot,

$$\hat{Q}_{PRRR}(d,\theta,\phi,\psi) = \hat{H}(0,d)\hat{G}(\theta,0)\hat{W}(\phi,0)\hat{F}(\psi,0), \tag{5.74}$$

or, if we expand them,

$$\begin{aligned} Q^0 = {} & c\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2} - s\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}\mathsf{G}\cdot\mathsf{W} - c\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}\mathsf{W}\cdot\mathsf{F} - s\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}\mathsf{G}\cdot\mathsf{F} - s\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}\mathsf{G}\times\mathsf{W}\cdot\mathsf{F} \\ & - \epsilon\frac{d}{2}\big(s\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2}\mathsf{H}\cdot\mathsf{G} + c\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}\mathsf{H}\cdot\mathsf{W} + c\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}\mathsf{H}\cdot\mathsf{F} + s\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}\mathsf{H}\cdot\mathsf{G}\times\mathsf{W} + \\ & s\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}\mathsf{H}\cdot\mathsf{G}\times\mathsf{F} + c\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}\mathsf{H}\cdot\mathsf{W}\times\mathsf{F} + s\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}(\mathsf{H}\cdot(\mathsf{G}\times\mathsf{W})\times\mathsf{F} - \mathsf{G}\cdot\mathsf{W}\mathsf{H}\cdot\mathsf{F})\big), \end{aligned} \tag{5.75}$$

$$\begin{aligned} \mathsf{Q} = {} & s\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2}\mathsf{G} + c\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}\mathsf{W} + c\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}\mathsf{F} + s\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}\mathsf{G}\times\mathsf{W} + s\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}\mathsf{G}\times\mathsf{F} + \\ & c\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}\mathsf{W}\times\mathsf{F} + s\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}((\mathsf{G}\times\mathsf{W})\times\mathsf{F} - (\mathsf{G}\cdot\mathsf{W})\mathsf{F}) + \epsilon\frac{d}{2}\big(c\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2}\mathsf{H} + \\ & s\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2}\mathsf{H}\times\mathsf{G} + c\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}\mathsf{H}\times\mathsf{W} + c\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}\mathsf{H}\times\mathsf{F} + s\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}(\mathsf{H}\times(\mathsf{G}\times\mathsf{W}) - \mathsf{G}\cdot\mathsf{W}\mathsf{H}) \\ & + s\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}(\mathsf{H}\times(\mathsf{G}\times\mathsf{F}) - \mathsf{G}\cdot\mathsf{F}\mathsf{H}) + c\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}(\mathsf{H}\times(\mathsf{W}\times\mathsf{F}) - \mathsf{W}\cdot\mathsf{F}\mathsf{H}) + \\ & s\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}(\mathsf{H}\times((\mathsf{G}\times\mathsf{W})\times\mathsf{F}) - \mathsf{G}\cdot\mathsf{W}\mathsf{H}\times\mathsf{F} - \mathsf{G}\times\mathsf{W}\cdot\mathsf{F}\mathsf{H})\big), \end{aligned} \tag{5.76}$$

are equated to the task dual quaternions to create the design equations. The PRRR robot, with $r = 3$ and $t = 1$, can reach any orientation, and the count gives $n_{max} = 8$ complete task positions. The set of parameterized design equations,

$$\hat{Q}_{PRRR}(d^i,\theta^i,\phi^i,\psi^i) = \hat{P}^i, \quad i = 2,\ldots,8, \tag{5.77}$$

consists of 49 equations in 49 unknowns, out of which 21 are structural parameters and 28 are values of the joint variables to reach each of the task positions.

Solving the design equations

Table 5.17: The task positions for a PRRR robot.

Axis	*Rot.*	*Trans.*
$(1.0, 0.0, 0.0; 0.0, 0.0, 0.0)$	0	0
$(0.66, -0.62, 0.43; 1.36, -0.77, -3.21)$	2.89	-2.04
$(-0.65, -0.67, 0.33; -0.37, 0.12, -0.51)$	2.33	2.90
$(0.50, 0.68, -0.53; -2.05, 1.06, -0.57)$	2.31	0.37
$(-0.49, 0.46, 0.74; 1.01, -0.06, 0.71)$	1.58	0.15
$(-0.56, 0.39, -0.73; -1.48, 0.87, 1.58)$	2.79	0.43
$(0.75, -0.51, 0.43; -0.68, -0.95, 0.06)$	0.75	0.83
$(-0.60, 0.78, 0.15; -1.15, -1.07, 0.96)$	0.68	-0.19

The set of design equations in Eq.(5.77) has been solved in the parameterized form only. A possible reduction scheme would be like the one presented in Section 5.4.5, eliminating first two revolute and one prismatic joint variable and the last revolute joint variable in a second step. However, the equations so obtained are of high degree and their numerical solution becomes more costly than the solution of the parameterized set, as shown in Section 5.4.1.

For the parameterized solution, we performed 2500 runs with random initial values, each run of 4000 iterations. Each iteration took about 14 seconds on a G4 at 733 MHz, which accounts to a total time of 9.8 hours. We obtained 91 real solutions; we present the first ten in Appendix A, sorted by increasing total link length, defined as the sum of the Denavit-Hartenberg parameters a_{ij} and d_i. In Figure 5.18 we can see

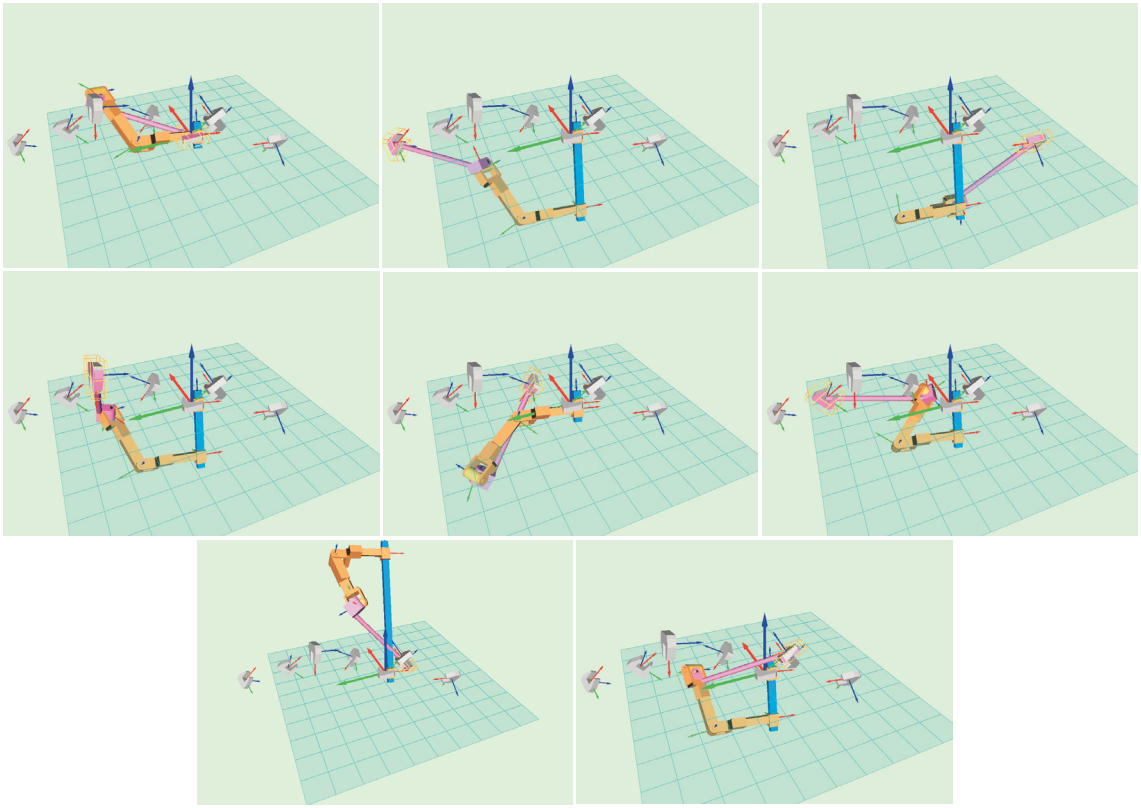

Figure 5.18: One PRRR solution reaching the eight task positions

the first solution reaching the eight task positions.

5.4.4 The CRR robot

The CRR robot is a special case of the PRRR serial chain in which the direction of the axes of the initial revolute and prismatic joints are parallel. Let the fixed axis $\mathsf{G} = \mathbf{g} + \epsilon\mathbf{g}^0$ allow a rotation by θ and be connected to an axis $\mathsf{H} = \mathbf{h} + \epsilon\mathbf{h}^0$ that allows the translation d, and require the directions $\mathbf{h}$ and $\mathbf{g}$ to be parallel. The third joint is a rotation of angle ϕ about an axis $\mathsf{W} = \mathbf{w} + \epsilon\mathbf{w}^0$, and the fourth joint is a rotation ψ about an axis $\mathsf{F} = \mathbf{f} + \epsilon\mathbf{f}^0$, see Figure 5.19.

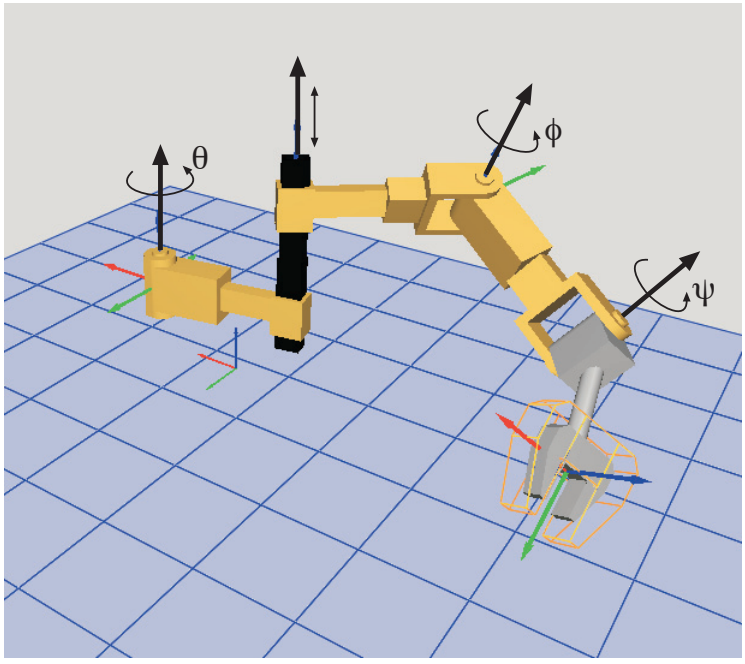

Figure 5.19: The CRR robot

A brief history

A solution for the CRR robot has been presented for the first time in [74], using the dual quaternion formulation and solving the equations numerically.

The design equations

The dual quaternion kinematics equations for the CRR robot can be expressed in a compact way as

$$\hat{Q}_{CRR}(\theta, d, \phi, \psi) = \hat{G}(\theta, d)\hat{W}(\phi, 0)\hat{F}(\psi, 0), \tag{5.78}$$

because if $\mathbf{h} = \mathbf{g}$ in Eq.(5.74), then $\hat{H}(0, d)\hat{G}(\theta, 0) = \hat{G}(\theta, d)$. The expansion of the kinematics equations can be obtained from Eq.(5.76) adding this condition.

To create the design equations, we consider $r = 3$, $t = 1$ and the two extra conditions that make $\mathbf{g} = \mathbf{h}$. We obtain $n_{max} = 7$ task positions. The design equations are

$$\hat{Q}_{CRR}(\theta^i, d^i, \phi^i, \psi^i) = \hat{P}^i, \quad i = 2, \ldots 7. \tag{5.79}$$

It is a set of 42 equations in 42 unknowns, out of which 24 define the values of the joint variables.

Solving the design equations

Solve linearly for three joint variables as explained in section 3.8.2. The parameterized design equations in Eq.(5.79) can be written as a linear transformation for the vector

of Eq.(3.47) with joint variables θ, ϕ and d. We solve for $\hat{V}(\theta, \phi, d)$ as demonstrated in Eqs.(3.55) and (3.56), but now the solutions are functions of the joint variable ψ.

The subspaces of solutions, defined by the relations Eq.(3.57) and Eq.(3.58), have the structure

$$A_i \cos^2 \frac{\psi}{2} + B_i \cos \frac{\psi}{2} \sin \frac{\psi}{2} + C_i \sin^2 \frac{\psi}{2} = 0, \quad i = 1, 2, 3. \tag{5.80}$$

We solve linearly for $\cos^2 \frac{\psi}{2}$ and $\sin^2 \frac{\psi}{2}$ using two 2×2 systems. Equating both results for each variable, we get two reduced design equations,

$$\mathcal{M}_1 : \quad \frac{C1(-A_3B_2C_1 + A_2B_3C_1 + A_3B_1C_2 - A_1B_3C_2 - A_2B_1C_3 + A_1B_2C_3)}{(A_2C_1 - A_1C_2)(A_3C_1 - A_1C_3)} = 0,$$

$$\mathcal{M}_2 : \quad \frac{A_1(A_3B_2C_1 - A_2B_3C_1 - A_3B_1C_2 + A_1B_3C_2 + A_2B_1C_3 - A_1B_2C_3)}{(A_2C_1 - A_1C_2)(A_3C_1 - A_1C_3)} = 0. \tag{5.81}$$

These two reduced design equations, plus the set of Plücker constraints, form the final set of 18 equations in 18 parameters,

$$\begin{aligned} &\{\mathcal{M}_1, \mathcal{M}_2\}^i, \quad i = 1, \ldots 6, \\ &\mathbf{g} \cdot \mathbf{g} = 1, \quad \mathbf{w} \cdot \mathbf{w} = 1, \quad \mathbf{f} \cdot \mathbf{f} = 1, \\ &\mathbf{g} \cdot \mathbf{g}_0 = 0, \quad \mathbf{w} \cdot \mathbf{w}_0 = 0, \quad \mathbf{f} \cdot \mathbf{f}_0 = 0. \end{aligned} \tag{5.82}$$

Numerical Example

The goal positions for the following example have been randomly generated and are specified as screw axis, rotation and translation in Table 5.18.

Table 5.18: The task positions for a CRR robot

Pos.	*Axis*	*Rot.*	*Transl.*
1	$(1.0, 0.0, 0.0) + \epsilon(0.0, 0.0, 0.0)$	0	0
2	$(0.66, -0.62, 0.43) + \epsilon(1.36, -0.77, -3.21)$	2.89	-2.04
3	$(-0.65, -0.68, 0.33) + \epsilon(-0.37, 0.11, -0.51)$	2.33	2.90
4	$(0.50, 0.68, -0.53) + \epsilon(-2.05, 1.06, -0.57)$	2.31	0.37
5	$(-0.49, 0.46, 0.74) + \epsilon(1.01, -0.06, 0.71)$	1.58	0.15
6	$(-0.57, 0.39, -0.73) + \epsilon(-1.48, 0.87, 1.58)$	2.79	0.43
7	$(0.75, -0.51, 0.43) + \epsilon(-0.68, -0.95, 0.06)$	0.75	0.83

We solved the parameterized design equations using Matlab's *fsolve* on a MacG4 at 733MHz. Starting with random initial conditions, we obtained 52 different real solutions from 1000 runs of 3000 iterations each. Each run took approximately 15 seconds; in whole, the process took about 4 hours.

The solution of the reduced design equations yielded eight real solutions using also random initial conditions and 1000 runs of 3000 iterations each. The increased complexity of the equations required 14 hours to complete the process. Thus, numerical solution of the original parameterized equations seems more efficient. However, he reduced equations can be further simplified.

We present numerical data of 10 of the solutions in Appendix A, for those chains with shortest total link length, defined as the sum of the Denavit-Hartenberg parameters $a_{12} + d_2 + a_{23}$. Figure 5.20 presents solution 11 reaching the seven positions.

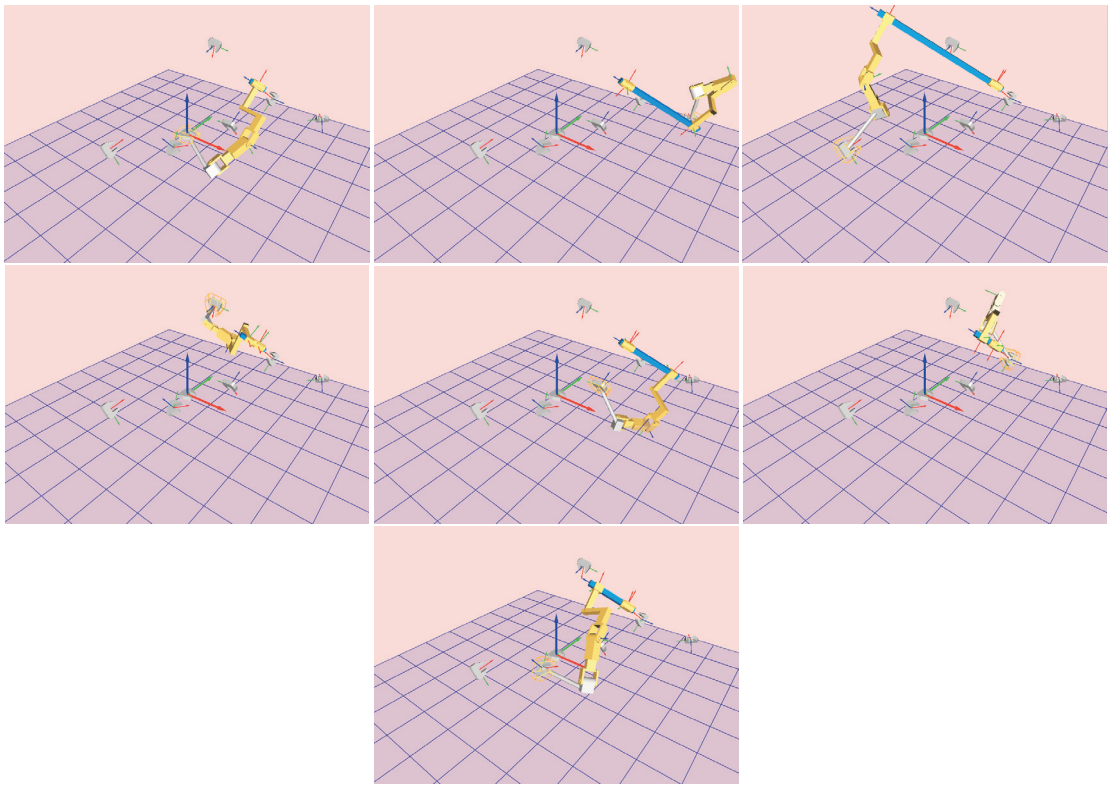

Figure 5.20: A solution for the CRR robot reaching the seven goal positions

5.4.5 The TPR robot

The TPR serial robot is a four-degree-of-freedom robot that can be seen as a special case of the RRPR robot, for which the base joint T consists of two revolute joints of perpendicular axes that intersect on a point. The fixed axis G_1 allows rotation of angle θ_1 about it. Located at 90^o and intersecting G_1 is the revolute axis G_2, which allows rotation of angle θ_2. We denote by $\mathbf{c}$ the intersection point of the two rotation axes G_1 and G_2.

The T joint is followed along the chain by a translation d along an arbitrary direction $\mathbf{h}$ and finally a rotation of angle ϕ about an arbitrary axis W, see Figure

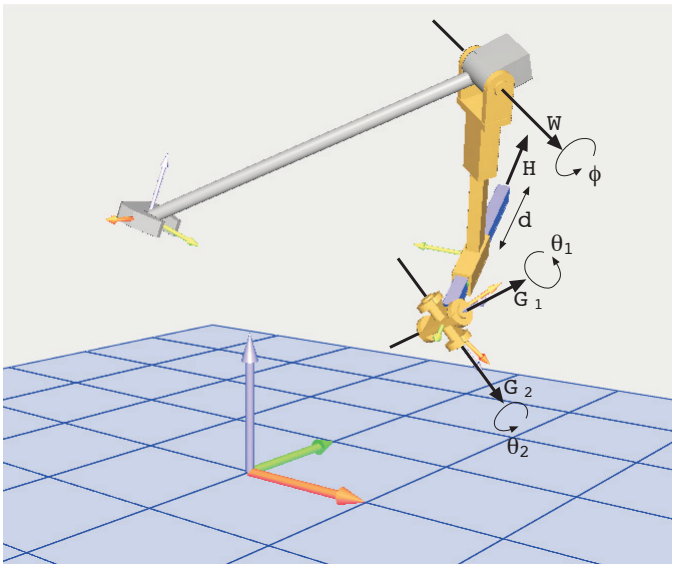

Figure 5.21: The spatial TPR robot

5.21.

A brief history

The first solution for the finite position synthesis of the TPR robot was presented in [75], where it was solved numerically using the dual quaternion formulation.

The design equations

The dual quaternion representation for the relative displacements of the chain is given by

$$\hat{Q}_{TPR} = \hat{G}_1(\theta_1, 0)\hat{G}_2(\theta_2, 0)\hat{H}(0, d)\hat{W}(\phi, 0). \tag{5.83}$$

When applying the dual quaternion product we obtain the expression $\hat{Q}_{TPR} =$

$Q^0 + \mathsf{Q}$, where the point is

$$
\begin{aligned}
Q^0 = {} & c\frac{\theta_1}{2}c\frac{\theta_2}{2}c\frac{\phi}{2} - s\frac{\theta_1}{2}c\frac{\theta_2}{2}s\frac{\phi}{2}\mathsf{G}_1\cdot\mathsf{W} - c\frac{\theta_1}{2}s\frac{\theta_2}{2}s\frac{\phi}{2}\mathsf{G}_2\cdot\mathsf{W} - \\
& s\frac{\theta_1}{2}s\frac{\theta_2}{2}s\frac{\phi}{2}(\mathsf{G}_1\times\mathsf{G}_2)\cdot\mathsf{W} - \epsilon\Big(\frac{d}{2}s\frac{\theta_1}{2}c\frac{\theta_2}{2}c\frac{\phi}{2}\mathsf{G}_1\cdot\mathsf{H} + \frac{d}{2}c\frac{\theta_1}{2}s\frac{\theta_2}{2}c\frac{\phi}{2}\mathsf{G}_2\cdot\mathsf{H} \\
& + \frac{d}{2}c\frac{\theta_1}{2}c\frac{\theta_2}{2}s\frac{\phi}{2}\mathsf{H}\cdot\mathsf{W} + \frac{d}{2}s\frac{\theta_1}{2}s\frac{\theta_2}{2}c\frac{\phi}{2}(\mathsf{G}_1\times\mathsf{G}_2)\cdot\mathsf{H} + \frac{d}{2}s\frac{\theta_1}{2}c\frac{\theta_2}{2}s\frac{\phi}{2}(\mathsf{G}_1\times\mathsf{H})\cdot\mathsf{W} \\
& + \frac{d}{2}c\frac{\theta_1}{2}s\frac{\theta_2}{2}s\frac{\phi}{2}(\mathsf{G}_2\times\mathsf{H})\cdot\mathsf{W} + \frac{d}{2}s\frac{\theta_1}{2}s\frac{\theta_2}{2}s\frac{\phi}{2}((\mathsf{G}_1\times\mathsf{G}_2)\times\mathsf{H})\cdot\mathsf{W}\Big),
\end{aligned}
\tag{5.84}
$$

and the dual vector

$$
\begin{aligned}
\mathsf{Q} = {} & s\frac{\theta_1}{2}c\frac{\theta_2}{2}c\frac{\phi}{2}\mathsf{G}_1 + c\frac{\theta_1}{2}s\frac{\theta_2}{2}c\frac{\phi}{2}\mathsf{G}_2 + c\frac{\theta_1}{2}c\frac{\theta_2}{2}s\frac{\phi}{2}\mathsf{W} + s\frac{\theta_1}{2}s\frac{\theta_2}{2}c\frac{\phi}{2}\mathsf{G}_1\times\mathsf{G}_2 + \\
& s\frac{\theta_1}{2}c\frac{\theta_2}{2}s\frac{\phi}{2}\mathsf{G}_1\times\mathsf{W} + c\frac{\theta_1}{2}s\frac{\theta_2}{2}s\frac{\phi}{2}\mathsf{G}_2\times\mathsf{W} + s\frac{\theta_1}{2}s\frac{\theta_2}{2}s\frac{\phi}{2}(\mathsf{G}_1\times\mathsf{G}_2)\times\mathsf{W} + \\
& \epsilon\Big(\frac{d}{2}c\frac{\theta_1}{2}c\frac{\theta_2}{2}c\frac{\phi}{2}\mathsf{H} + \frac{d}{2}s\frac{\theta_1}{2}c\frac{\theta_2}{2}s\frac{\phi}{2}((\mathsf{G}_1\times\mathsf{H})\times\mathsf{W} - (\mathsf{G}_1\cdot\mathsf{H})\mathsf{W}) + \\
& \frac{d}{2}c\frac{\theta_1}{2}s\frac{\theta_2}{2}s\frac{\phi}{2}((\mathsf{G}_2\times\mathsf{H})\times\mathsf{W} - (\mathsf{G}_2\cdot\mathsf{H})\mathsf{W}) + \frac{d}{2}s\frac{\theta_1}{2}c\frac{\theta_2}{2}c\frac{\phi}{2}\mathsf{G}_1\times\mathsf{H} + \\
& \frac{d}{2}c\frac{\theta_1}{2}s\frac{\theta_2}{2}c\frac{\phi}{2}\mathsf{G}_2\times\mathsf{H} + \frac{d}{2}s\frac{\theta_1}{2}s\frac{\theta_2}{2}c\frac{\phi}{2}(\mathsf{G}_1\times\mathsf{G}_2)\times\mathsf{H} + \frac{d}{2}c\frac{\theta_1}{2}c\frac{\theta_2}{2}s\frac{\phi}{2}\mathsf{H}\times\mathsf{W} + \\
& \frac{d}{2}s\frac{\theta_1}{2}s\frac{\theta_2}{2}s\frac{\phi}{2}(((\mathsf{G}_1\times\mathsf{G}_2)\times\mathsf{H})\times\mathsf{W} - ((\mathsf{G}_1\times\mathsf{G}_2)\cdot\mathsf{H})\mathsf{W})\Big).
\end{aligned}
\tag{5.85}
$$

The T-joint axis is formulated so that the coordinates of the intersection point $\mathbf{c}$ appear explicitely,

$$
\begin{aligned}
\mathsf{G1} &= (g_{1x}, g_{1y}, g_{1z}) + \epsilon((c_x, c_y, c_z)\times(g_{1x}, g_{1y}, g_{1z})) \\
\mathsf{G2} &= (g_{2x}, g_{2y}, g_{2z}) + \epsilon((c_x, c_y, c_z)\times(g_{2x}, g_{2y}, g_{2z}))
\end{aligned}
\tag{5.86}
$$

Use Eq.(3.16) with $r = 3$ revolute joints, $t = 1$ prismatic joint, and the two extra constraints that make G_1 and G_2 a T joint, to compute $n_{max} = 7$ positions. Equate

Eqs.(5.83) to the task dual quaternions,

$$\hat{Q}_{TPR}(\theta_1^i, \theta_2^i, d^i, \phi^i) - \hat{P}^i = \vec{0}, \quad i = 2, \ldots, 7, \tag{5.87}$$

to obtain the set of design equations. It consists of 42 equations in 42 unknowns, out of which 24 correspond to the inverse kinematics for the joint variables.

Solving the design equations

We reduced the design equations for the TPR robot in two different ways. Here we present the results obtained when eliminating two rotation angles first and the translation and the third angle in two successive steps.

Every direction will be reached by moving the rotation axes accordingly to the third rotation parameter as appears in the solution of the linear system,

$$[R(\phi)]V(\theta_1, \theta_2) = P \tag{5.88}$$

with the matrix

$$[R(\phi)] = \begin{bmatrix} G_1W(\phi) & G_2W(\phi) & G_1G_2W(\phi) & W(\phi) \end{bmatrix}, \tag{5.89}$$

whose columns are the quaternions,

$$\begin{aligned}
G_1W(\phi) &= -\sin\frac{\phi}{2}\mathbf{g}_1 \cdot \mathbf{w} + \cos\frac{\phi}{2}\mathbf{g}_1 + \sin\frac{\phi}{2}\mathbf{g}_1 \times \mathbf{w} \\
G_2W(\phi) &= -\sin\frac{\phi}{2}\mathbf{g}_2 \cdot \mathbf{w} + \cos\frac{\phi}{2}\mathbf{g}_2 + \sin\frac{\phi}{2}\mathbf{g}_2 \times \mathbf{w} \\
G_1G_2W(\phi) &= -\cos\frac{\phi}{2}\mathbf{g}_1 \cdot \mathbf{g}_2 - \sin\frac{\phi}{2}(\mathbf{g}_1 \times \mathbf{g}_2) \cdot \mathbf{w} + \\
&\qquad \cos\frac{\phi}{2}\mathbf{g}_1 \times \mathbf{g}_2 + \sin\frac{\phi}{2}((\mathbf{g}_1 \times \mathbf{g}_2) \times \mathbf{w} - (\mathbf{g}_1 \cdot \mathbf{g}_2)\mathbf{w}) \\
W(\phi) &= \cos\frac{\phi}{2} + \sin\frac{\phi}{2}\mathbf{w}
\end{aligned} \tag{5.90}$$

The matrix $[R(\phi)]$ is orthogonal when we solve for variables whose axes are perpendicular. Solve for the T-joint angles by transposing the matrix $[R(\phi)]$ to obtain

$$\begin{aligned}
\sin\frac{\theta_1}{2}\cos\frac{\theta_2}{2} &= \cos\frac{\phi}{2}\mathbf{g}_1\cdot\mathbf{p} + \sin\frac{\phi}{2}((\mathbf{g}_1\times\mathbf{w})\cdot\mathbf{p} - p_w\mathbf{g}_1\cdot\mathbf{w}) \\
\cos\frac{\theta_1}{2}\sin\frac{\theta_2}{2} &= \cos\frac{\phi}{2}\mathbf{g}_2\cdot\mathbf{p} + \sin\frac{\phi}{2}((\mathbf{g}_2\times\mathbf{w})\cdot\mathbf{p} - p_w\mathbf{g}_2\cdot\mathbf{w}) \\
\sin\frac{\theta_1}{2}\sin\frac{\theta_2}{2} &= \cos\frac{\phi}{2}(\mathbf{g}_1\times\mathbf{g}_2)\cdot\mathbf{p} + \sin\frac{\phi}{2}(((\mathbf{g}_1\times\mathbf{g}_2)\times\mathbf{w})\cdot\mathbf{p} - p_w(\mathbf{g}_1\times\mathbf{g}_2)\cdot\mathbf{w}) \\
\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2} &= \cos\frac{\phi}{2}p_w + \sin\frac{\phi}{2}\mathbf{w}\cdot\mathbf{p}
\end{aligned} \tag{5.91}$$

When we impose the constraint of Eq.(3.42) on the T-joint angles, we obtain the condition for the angle ϕ,

$$\begin{aligned}
&\Big((\mathbf{g}_1\cdot\mathbf{p})(\mathbf{g}_2\cdot\mathbf{p}) - p_w(\mathbf{g}_1\times\mathbf{g}_2)\cdot\mathbf{p}\Big)\cos^2\frac{\phi}{2} + \\
&\Big((\mathbf{g}_1\times\mathbf{g}_2)\cdot\mathbf{w}(1-2\mathbf{p}\cdot\mathbf{p}) - 2p_w(\mathbf{g}_1\cdot\mathbf{w})(\mathbf{g}_2\cdot\mathbf{p}) - 2(\mathbf{g}_1\cdot\mathbf{p})(\mathbf{p}\times\mathbf{w})\cdot\mathbf{g}_2\Big)\cos\frac{\phi}{2}\sin\frac{\phi}{2} + \\
&\Big(((\mathbf{g}_1\times\mathbf{g}_2)\cdot\mathbf{p} + 2(\mathbf{g}_1\cdot(\mathbf{p}\times\mathbf{w}))(\mathbf{g}_2\cdot\mathbf{w}))p_w + (\mathbf{g}_2\cdot(\mathbf{p}\times\mathbf{w}))(\mathbf{g}_1\cdot(\mathbf{p}\times\mathbf{w})) + \\
&(\mathbf{g}_1\times\mathbf{g}_2)\cdot(\mathbf{p}\times\mathbf{w})(\mathbf{p}\cdot\mathbf{w}) + (1-\mathbf{p}\cdot\mathbf{p})(\mathbf{g}_1\cdot\mathbf{w})(\mathbf{g}_2\cdot\mathbf{w})\Big)\sin^2\frac{\phi}{2} = 0,
\end{aligned} \tag{5.92}$$

which we denote as

$$A_0\cos^2\frac{\phi}{2} + B_0\sin^2\frac{\phi}{2} + C_0\cos\frac{\phi}{2}\sin\frac{\phi}{2} = 0. \tag{5.93}$$

To eliminate the translation d and the rotation ϕ, the solutions for the angles θ_1, θ_2 are substituted in the three moment equations of the dual quaternion. They are linear in the joint translation d and quadratic in the joint rotation ϕ, and have the structure

$$(A_{1i}d + A_{0i})\cos^2\frac{\phi}{2} + (B_{1i}d + B_{0i})\sin^2\frac{\phi}{2} + (C_{1i}d + C_{0i})\cos\frac{\phi}{2}\sin\frac{\phi}{2} + D_{0i} = 0,$$
$$i = 1, 2, 3. \tag{5.94}$$

To eliminate the joint variable ϕ we add to these the condition of Eq.(5.93), to create the homogeneous system

$$\begin{bmatrix} A_{11}d + A_{01} & B_{11}d + B_{01} & C_{11}d + C_{01} & D_{01} \\ A_{12}d + A_{02} & B_{12}d + B_{02} & C_{12}d + C_{02} & D_{02} \\ A_{13}d + A_{03} & B_{13}d + B_{03} & C_{13}d + C_{03} & D_{03} \\ A_0 & B_0 & C_0 & 0 \end{bmatrix} \begin{Bmatrix} \cos^2\frac{\phi}{2} \\ \sin^2\frac{\phi}{2} \\ \cos\frac{\phi}{2}\sin\frac{\phi}{2} \\ 1 \end{Bmatrix} = \vec{0} \tag{5.95}$$

Impose that the determinant must be equal to zero to have solutions. The subspace of solutions from the matrix corresponding to the first three rows are a function of the prismatic joint variable d. The relation between these three solutions,

$$\begin{aligned} &\cos^2\frac{\phi}{2} + \sin^2\frac{\phi}{2} = 1 \\ &(\cos^2\frac{\phi}{2})(\sin^2\frac{\phi}{2}) = (\cos\frac{\phi}{2}\sin\frac{\phi}{2})^2, \end{aligned} \tag{5.96}$$

lead to two equations, a cubic and a quartic equations in d. These two conditions, together with the quadratic equation from the determinant of the matrix of eq.(5.95),

$$K_{4i}d^4 + K_{3i}d^3 + K_{2i}d^2 + K_{1i}d + K_{0i} = 0, \quad i = 1, 2, 3 \tag{5.97}$$

or in matrix form,

$$\begin{bmatrix} 0 & 0 & K_{21} & K_{11} & K_{01} \\ 0 & K_{32} & K_{22} & K_{12} & K_{02} \\ K_{43} & K_{33} & K_{23} & K_{13} & K_{03} \end{bmatrix} \begin{Bmatrix} d^4 \\ d^3 \\ d^2 \\ d \\ 1 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \tag{5.98}$$

are used to eliminate the parameter d, to obtain two final reduced design equations $\{\mathcal{M}_1, \mathcal{M}_2\}$ per goal dual quaternion.

The final set of design equations consists of twelve reduced equations and six Plücker constraints,

$$\begin{aligned}
&\{\mathcal{M}_1, \mathcal{M}_2\}^i, \quad i = 2, \ldots, 7, \\
&\mathbf{g}_1 \cdot \mathbf{g}_1 = 1, \quad \mathbf{g}_2 \cdot \mathbf{g}_2 = 1, \quad \mathbf{w} \cdot \mathbf{w} = 1, \quad \mathbf{h} \cdot \mathbf{h} = 1, \\
&\mathbf{g}_1 \cdot \mathbf{g}_2 = 0, \quad \mathbf{w} \cdot \mathbf{w}^0 = 0,
\end{aligned} \tag{5.99}$$

that are solved for the 18 unknowns corresponding to the four joint axes.

Numerical example

Table 5.19: The task positions for a TPR robot

Pos.	*Real part*	*Dual part*
pos. 1	$(1.0, 0.0, 0.0)$	$(0.0, 0.0, 0.0)$
pos. 2	$(-0.21, 0.07, -0.10, 0.97)$	$(-0.46, 0.36, -0.20, -0.15)$
pos. 3	$(-0.11, 0.21, 0.88, 0.41)$	$(0.36, 2.43, -0.81, 0.62)$
pos. 4	$(0.58, -0.62, -0.49, -0.20)$	$(-2.14, -2.35, 0.61, -0.37)$
pos. 5	$(-0.19, 0.14, -0.55, 0.80)$	$(-1.33, -0.48, 0.51, 0.11)$
pos. 6	$(-0.28, 0.24, -0.00, 0.93)$	$(-0.00, -0.42, 0.43, 0.11)$
pos. 7	$(0.00, -0.01, -0.03, 1.00)$	$(0.07, -0.17, -0.65, -0.02)$

We solved both the parameterized and the reduced design equations. For the parameterized solution, we used the *fsolve* algorithm in Matlab on a G4 at 733 MHz. We performed runs of 10.000 iterations each, that took about 50 seconds each. After 880 runs we obtained 6 different solutions. The reduced equations were less suited

for the Newton-Raphson algorithm and we were able to obtain at most two solutions after many trials.

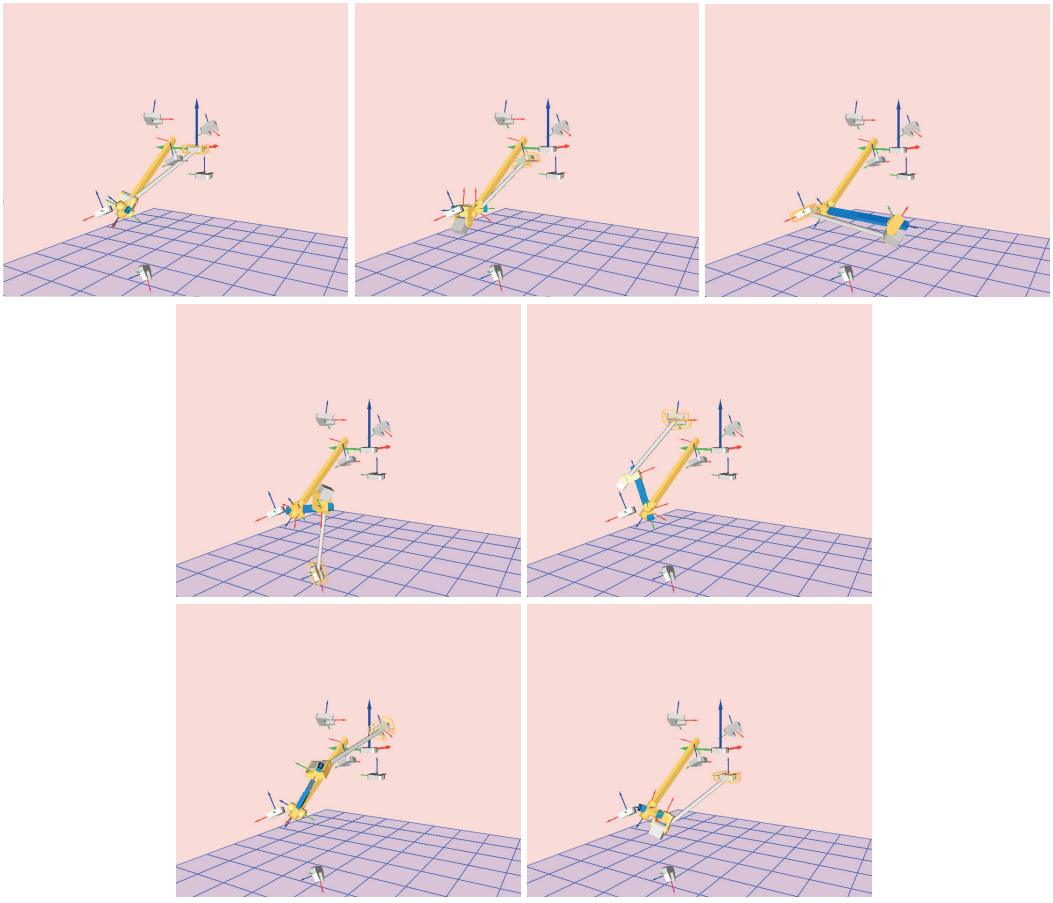

Figure 5.22: A TPR robot reaching the seven task positions

Table 5.19 presents the task positions and Figure 5.22 and Table 5.20 show one of the solutions; the complete set can be found in Appendix A.

5.4.6 Spatial RRRR robot

The spatial RRRR is the most general four-degree-of-freedom robot, consisting of four revolute joints G, W, F, and U which are arbitrarily located in space, see Figure

Table 5.20: The joint axes for the first TPR solution

Joint Axis	*Direction*	*Moment*
$\mathbf{g}_1$	$(-0.72, 0.13, -0.68)$	
$\mathbf{g}_2$	$(-0.54, 0.51, 0.67)$	
$\mathbf{c}$	$(-3.07, 2.02, -2.58)$	
$\mathbf{h}$	$(0.45, 0.89, -0.11)$	
W	$(0.29, -0.88, -0.36)$	$(-2.66, -1.61, 1.79)$

5.23.

A brief history

The synthesis equations for the 4R robot were stated and partially solved in [89] using the loop equations. They used optimization to solve for up to four task positions, fixing the extra design parameters. No complete solution has been published for the 4R robot, to our knowledge. There are results for the particular case obtained if we make three of the revolute joints into an S joint, see [12].

The design equations

The kinematics equations in dual quaternion form,

$$\hat{Q}_{4R}(\theta, \phi, \psi, \xi) = \hat{G}(\theta, 0)\hat{W}(\phi, 0)\hat{F}(\psi, 0)\hat{U}(\xi, 0), \tag{5.100}$$

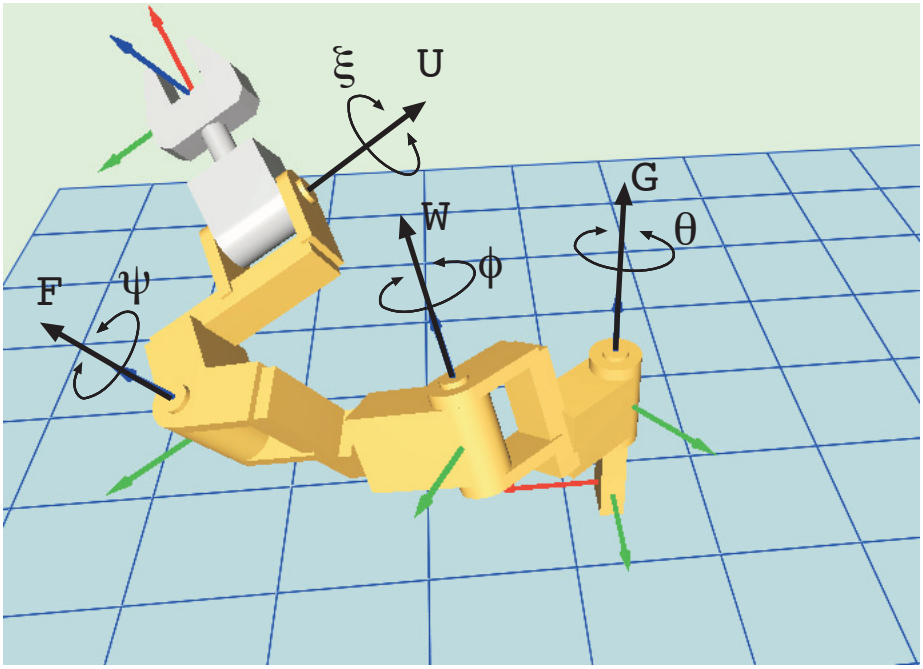

Figure 5.23: The spatial RRRR robot

are expanded to

$$
\begin{aligned}
Q^0 = {} & c\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2}c\frac{\xi}{2} - s\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}c\frac{\xi}{2}\mathsf{G}\cdot\mathsf{W} - c\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}c\frac{\xi}{2}\mathsf{W}\cdot\mathsf{F} - s\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}c\frac{\xi}{2}\mathsf{G}\cdot\mathsf{F} - \\
& s\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2}s\frac{\xi}{2}\mathsf{G}\cdot\mathsf{U} - c\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}s\frac{\xi}{2}\mathsf{W}\cdot\mathsf{U} - c\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}s\frac{\xi}{2}\mathsf{F}\cdot\mathsf{U} - \\
& s\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}c\frac{\xi}{2}(\mathsf{G}\times\mathsf{W})\cdot\mathsf{F} - s\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}s\frac{\xi}{2}(\mathsf{G}\times\mathsf{W})\cdot\mathsf{U} - s\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}s\frac{\xi}{2}(\mathsf{G}\times\mathsf{F})\cdot\mathsf{U} - \\
& c\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}s\frac{\xi}{2}(\mathsf{W}\times\mathsf{F})\cdot\mathsf{U} - s\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}s\frac{\xi}{2}((\mathsf{G}\times\mathsf{W})\times\mathsf{F}\cdot\mathsf{U} - \mathsf{G}\cdot\mathsf{WF}\cdot\mathsf{U}), \qquad (5.101)
\end{aligned}
$$

and

$$\begin{aligned}
\mathsf{Q} = {} & s\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2}c\frac{\xi}{2}\mathsf{G} + c\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}c\frac{\xi}{2}\mathsf{W} + c\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}c\frac{\xi}{2}\mathsf{F} + c\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2}s\frac{\xi}{2}\mathsf{U} + \\
& s\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}c\frac{\xi}{2}\mathsf{G}\times\mathsf{W} + s\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}c\frac{\xi}{2}\mathsf{G}\times\mathsf{F} + c\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}c\frac{\xi}{2}\mathsf{W}\times\mathsf{F} + \\
& s\frac{\theta}{2}c\frac{\phi}{2}c\frac{\psi}{2}s\frac{\xi}{2}\mathsf{G}\times\mathsf{U} + c\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}s\frac{\xi}{2}\mathsf{W}\times\mathsf{U} + c\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}s\frac{\xi}{2}\mathsf{F}\times\mathsf{U} + \\
& s\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}c\frac{\xi}{2}((\mathsf{G}\times\mathsf{W})\times\mathsf{F} - (\mathsf{G}\cdot\mathsf{W})\mathsf{F}) + s\frac{\theta}{2}s\frac{\phi}{2}c\frac{\psi}{2}s\frac{\xi}{2}((\mathsf{G}\times\mathsf{W})\times\mathsf{U} - (\mathsf{G}\cdot\mathsf{W})\mathsf{U}) + \\
& s\frac{\theta}{2}c\frac{\phi}{2}s\frac{\psi}{2}s\frac{\xi}{2}((\mathsf{G}\times\mathsf{F})\times\mathsf{U} - (\mathsf{G}\cdot\mathsf{F})\mathsf{U}) + c\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}s\frac{\xi}{2}((\mathsf{W}\times\mathsf{F})\times\mathsf{U} - (\mathsf{W}\cdot\mathsf{F})\mathsf{U}) + \\
& s\frac{\theta}{2}s\frac{\phi}{2}s\frac{\psi}{2}s\frac{\xi}{2}(((\mathsf{G}\times\mathsf{W})\times\mathsf{F})\times\mathsf{U} - (\mathsf{G}\cdot\mathsf{W})\mathsf{F}\times\mathsf{U} - (\mathsf{G}\times\mathsf{W})\cdot\mathsf{FU}), \qquad (5.102)
\end{aligned}$$

where s and c stand for sin and cos, respectively.

The maximum number of task positions to define is computed using Eq.(3.16). With $r = 4$ revolute joints, we obtain $n_{max} = 9$ task positions to synthesize a finite number of 4R robots.

To create the design equations, we identify the kinematics equations with the task dual quaternions,

$$\hat{Q}_{4R}(\theta^i, \phi^i, \psi^i, \xi^i) = \hat{P}^i, \quad i = 2, \ldots, 9. \qquad (5.103)$$

This set of parameterized equations consists of 56 equations in 56 unknowns, out of which 32 correspond to joint variables and 24 define the structure of the robot.

Solving the design equations

We solved the parameterized design equations for the nine random dual quaternions presented in Table 5.21, using Matlab's *fsolve* algorithm on a 500Mhz G4 processor.

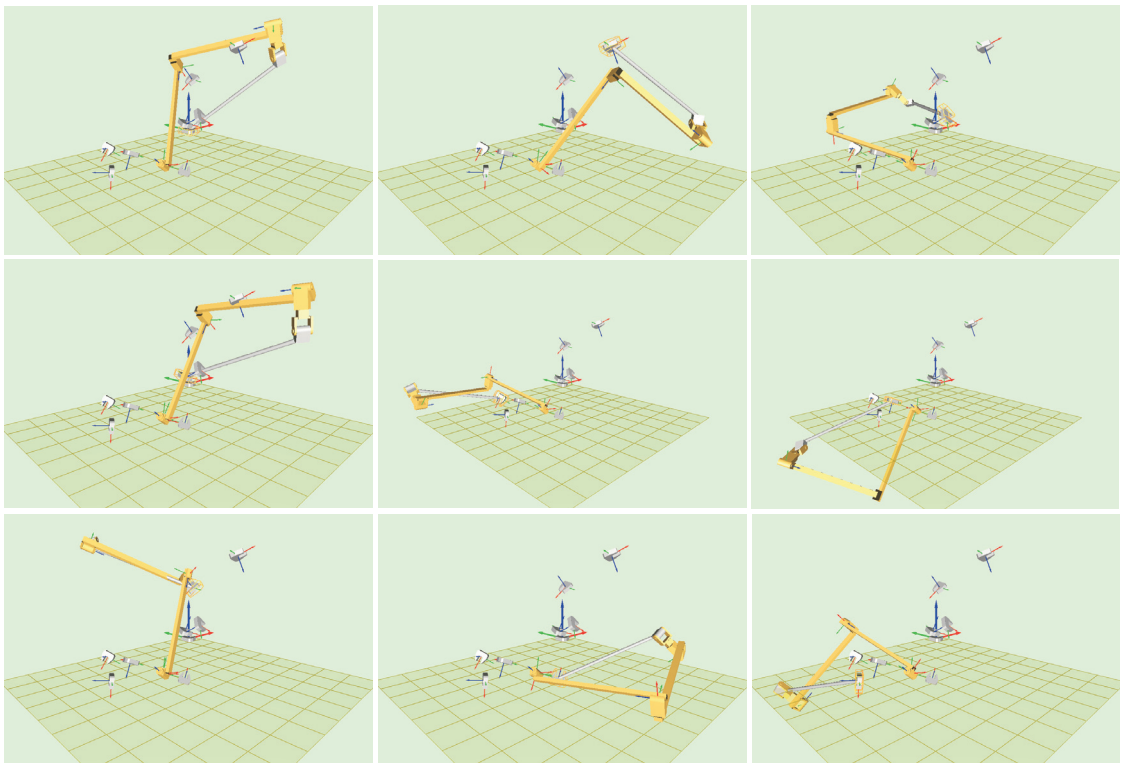

Figure 5.24: An RRRR solution reaching nine task positions

We performed 5000 runs of 6000 iterations each, to obtain 11 solutions, see Figure 5.24 and data in Appendix A. Each run took around 36 seconds, and the total time was of approximately 48 hours.

5.5 Synthesis of Five-degree-of-freedom Robots

Robots with five degrees of freedom present the greatest complexity from the synthesis point of view. Only those robots with S-joints have been solved, see [12] and more recently [27]. For these, the geometric constraints are easy to identify. The main three sets of kinematic chains considered here have three (RRRPP), four (RRRRP), and

Axis	*Rot.*	*Trans.*
$(1.0, 0.0, 0.0; 0.0, 0.0, 0.0)$	0	0
$(0.79, -0.60, 0.12; 0.99, 1.78, 2.22)$	2.67	2.67
$(-0.95, 0.18, 0.26; -0.48, -1.10, -0.98)$	2.28	1.37
$(0.50, 0.56, -0.66; -0.82, -0.26, -0.84)$	0.47	-1.80
$(0.50, 0.33, -0.80; -1.50, -1.65, -1.60)$	2.97	0.42
$(-0.69, -0.69, 0.20; -0.33, 0.78, 1.52)$	2.90	2.59
$(-0.37, -0.03, 0.93; 0.03, -0.67, -0.00)$	2.94	2.99
$(0.52, -0.51, 0.69; 0.22, -1.62, -1.36)$	0.88	-2.09
$(-0.64, 0.20, 0.74; -0.15, 2.48, -0.81)$	2.85	2.72

Table 5.21: The task positions for an RRRR robot

five (RRRRR) revolute joints. It is not difficult to create the dual quaternion design equations for five-degree-of-freedom robots, but the parameterized solution becomes very costly because of the high number of joint variables.

5.5.1 The PPS robot

The PPS robot is a five-degree-of-freedom robot. The fixed prismatic joint of direction $\mathbf{h}$ is followed along the chain by another prismatic direction $\mathbf{f}$. This is linked to a spherical joint S that we can model as three revolute joints G_1, G_2 and G_3 at perpendicular directions and intersecting on a point, or as an arbitrary direction S located also on the same point, see Figure 5.25.

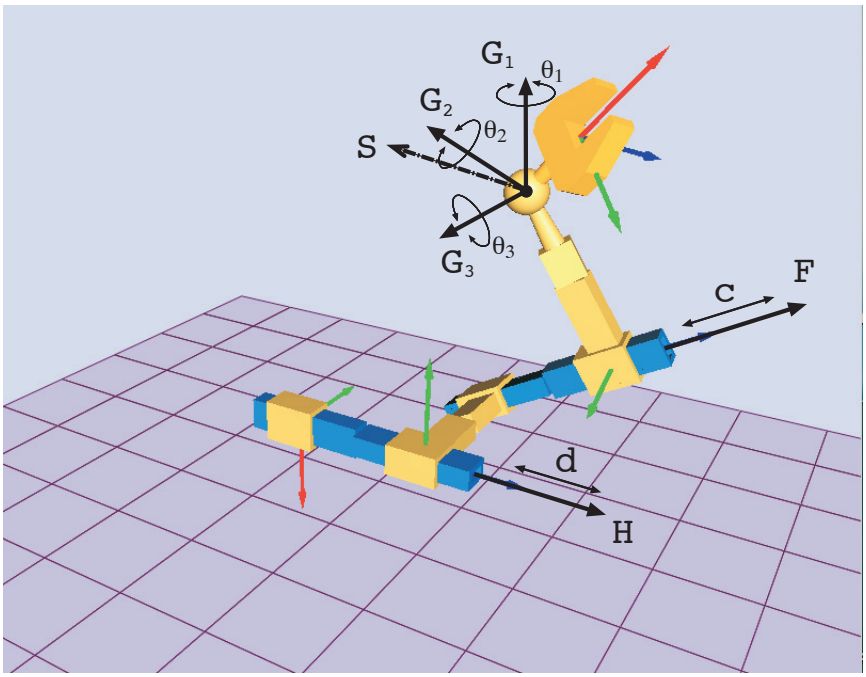

Figure 5.25: The PPS robot

A brief history

As most of the three-link robots that contain a spherical joint, the design equations for the PPS robot were stated by Chen and Roth [12] using geometric constraints. Later, Nielsen and Roth [64] solved those equations using Sylvester resultants to obtain ten solutions. Recently, the solutions of this same group of robots have been studied by Su [27].

The design equations

The spherical joint can be modeled as three perpendicular axes at a point, or also as an axis of different direction for each task position. Both formulations lead to the same results, as it has been discussed in Chapter 4. For uniformity reasons, we use

here three axes to model the spherical joint. We have the kinematics equations

$$\hat{Q}_{PPS} = \hat{H}(0,d)\hat{F}(0,c)\hat{G}_1(\theta_1,0)\hat{G}_2(\theta_2,0)\hat{G}_3(\theta_3,0). \tag{5.104}$$

The dual quaternion product yields

$$\hat{Q}_{PPS} = \begin{Bmatrix} \alpha_1\mathbf{g}_1 + \alpha_2\mathbf{g}_2 + \alpha_3\mathbf{g}_3 \\ \alpha_4 \end{Bmatrix} + \epsilon \begin{Bmatrix} \alpha_1(\mathbf{g}_1^0 + (\frac{d}{2}\mathbf{h} + \frac{c}{2}\mathbf{f}) \times \mathbf{g}_1) + \alpha_2(\mathbf{g}_2^0 + (\frac{d}{2}\mathbf{h} + \frac{c}{2}\mathbf{f}) \times \mathbf{g}_2) + \alpha_3(\mathbf{g}_3^0 + (\frac{d}{2}\mathbf{h} + \frac{c}{2}\mathbf{f}) \times \mathbf{g}_3) + \alpha_4(\frac{d}{2}\mathbf{h} + \frac{c}{2}\mathbf{f}) \\ -\alpha_1(\frac{d}{2}\mathbf{h}\cdot\mathbf{g}_1 + \frac{c}{2}\mathbf{f}\cdot\mathbf{g}_1) - \alpha_2(\frac{d}{2}\mathbf{h}\cdot\mathbf{g}_2 + \frac{c}{2}\mathbf{f}\cdot\mathbf{g}_2) - \alpha_3(\frac{d}{2}\mathbf{h}\cdot\mathbf{g}_3 + \frac{c}{2}\mathbf{f}\cdot\mathbf{g}_3) \end{Bmatrix} \tag{5.105}$$

The values for α_i are presented in Eq.(4.41) and are a combination of the three rotations that define the spherical joint, see Chapter 4 for details.

If we use explicitly the intersection point $\mathbf{c}$ of the three revolute joints, by writing $\mathbf{g}_i^0 = \mathbf{c} \times \mathbf{g}_i$, the previous expression can be rewritten as

$$\hat{Q}_{PPS} = \begin{Bmatrix} \alpha_1\mathbf{g}_1 + \alpha_2\mathbf{g}_2 + \alpha_3\mathbf{g}_3 \\ \alpha_4 \end{Bmatrix} + \epsilon \begin{Bmatrix} (\mathbf{c} + \frac{d}{2}\mathbf{h} + \frac{c}{2}\mathbf{f}) \times (\alpha_1\mathbf{g}_1 + \alpha_2\mathbf{g}_2 + \alpha_3\mathbf{g}_3) + \alpha_4(\frac{d}{2}\mathbf{h} + \frac{c}{2}\mathbf{f}) \\ -(\frac{d}{2}\mathbf{h} + \frac{c}{2}\mathbf{f}) \cdot (\alpha_1\mathbf{g}_1 + \alpha_2\mathbf{g}_2 + \alpha_3\mathbf{g}_3) \end{Bmatrix} \tag{5.106}$$

We apply Eq.(3.16) and Eq.(3.17) we need to consider the particularities of both the spherical joint and the plane defined by the prismatic joints. There is no limit in the orientations, and the counting gives a limit of $n_{max} = 6$ positions to obtain a finite number of solutions. We equate Eqs.(5.104) to the task dual quaternions,

$$\hat{Q}_{PPS}(d^i, c^i, \theta_1^i, \theta_2^i, \theta_3^i) - \hat{P}^i = \vec{0}, \quad i = 2, \ldots, 6, \tag{5.107}$$

to obtain the set of design equations. It consists of 32 equations in 32 unknowns, out of which 25 correspond to the inverse kinematics for the joint variables.

Solving the design equations

The elimination of the joint variables is done in two steps. We eliminate the revolute joint variables from the orientations. Linear solution of the prismatic joint variables leads to a reduced design equation. This can be solved algebraically.

The solution for the coefficients α_1, α_2, α_3, α_4 containing the revolute joint variables is computed linearly from the real part of Eq.(5.106) equated to the task goal quaternion. If we denote by $\mathbf{p}$ the real vector and p_w the real scalar of the task dual quaternion $\hat{P}$, we have

$$\alpha_1 = \mathbf{g}_1 \cdot \mathbf{p}, \quad \alpha_2 = \mathbf{g}_2 \cdot \mathbf{p}, \quad \alpha_3 = \mathbf{g}_3 \cdot \mathbf{p}, \quad \alpha_4 = p_w \tag{5.108}$$

The PPS chain does not have more revolute joints, hence the relation among the α coefficients does not impose any restriction on the directions of the joint axes $\mathbf{g}_1$, $\mathbf{g}_2$, $\mathbf{g}_3$ except for these to be perpendicular. We substitute these values in the dual part of the design equations. The set is linear in the prismatic variables d, c,

$$\begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \mathbf{v}_3 \end{bmatrix} \begin{Bmatrix} d \\ c \\ 1 \end{Bmatrix} = \vec{0} \tag{5.109}$$

where the column vectors are

$$\begin{aligned} \mathbf{v}_1 &= \frac{1}{2}(p_w\mathbf{h} + \mathbf{h} \times (\mathbf{g}_1 \cdot \mathbf{pg}_1 + \mathbf{g}_2 \cdot \mathbf{pg}_2 + \mathbf{g}_3 \cdot \mathbf{pg}_3)) \\ \mathbf{v}_2 &= \frac{1}{2}(p_w\mathbf{f} + \mathbf{f} \times (\mathbf{g}_1 \cdot \mathbf{pg}_1 + \mathbf{g}_2 \cdot \mathbf{pg}_2 + \mathbf{g}_3 \cdot \mathbf{pg}_3)) \\ \mathbf{v}_3 &= -\mathbf{p}^0 + \mathbf{c} \times (\mathbf{g}_1 \cdot \mathbf{pg}_1 + \mathbf{g}_2 \cdot \mathbf{pg}_2 + \mathbf{g}_3 \cdot \mathbf{pg}_3)) \end{aligned} \tag{5.110}$$

The reduced design equation is formulated as the determinant of the matrix in Eq.(5.109). In this determinant, the prismatic directions $\mathbf{h}$ and $\mathbf{f}$ appear as the cross product defining the normal direction to their plane, $\mathbf{n} = \mathbf{h} \times \mathbf{f}$. Using this and noticing that $\mathbf{g}_1 \cdot \mathbf{pg}_1 + \mathbf{g}_2 \cdot \mathbf{pg}_2 + \mathbf{g}_3 \cdot \mathbf{pg}_3 = \mathbf{p}$, the design equation simplifies to

$$\mathcal{R}: \quad (1 - \mathbf{p} \cdot \mathbf{p})(\mathbf{n} \cdot \mathbf{c} \times \mathbf{p} - \mathbf{n} \cdot \mathbf{p}^0) + p_w(\mathbf{n} \times \mathbf{p} \cdot \mathbf{c} \times \mathbf{p} - \mathbf{n} \cdot \mathbf{p} \times \mathbf{p}^0) - \mathbf{p} \cdot \mathbf{p}^0 \mathbf{n} \cdot \mathbf{p} = 0. \tag{5.111}$$

The final set of reduced design equations

$$\{\mathcal{R}\}^i, \quad i = 2, \ldots, 6, \quad \mathbf{n} \cdot \mathbf{n} = 1 \tag{5.112}$$

is solved either directly or transformed into a univariate polynomial using resultant methods, to obtain as many as ten complex solutions.

Numerical example

Table 5.22: The task positions for a PPS robot

Pos.	*Dir.*	*Mom.*	*Rot.*	*Trans.*
pos. 1	$(1.0, 0.0, 0.0)$	$(0.0, 0.0, 0.0)$	0	0
pos. 2	$(0.98, 0.02, -0.18)$	$(0.00, 1.16, 0.15)$	0.90	-0.53
pos. 3	$(0.41, 0.84, 0.35)$	$(1.23, -0.47, -0.29)$	2.07	-0.93
pos. 4	$(0.06, 0.53, 0.84)$	$(-0.62, 0.10, -0.02)$	5.62	2.89
pos. 5	$(0.54, -0.74, 0.40)$	$(0.74, 0.63, 0.16)$	2.89	-0.59
pos. 6	$(0.08, 0.33, -0.94)$	$(-1.03, -1.53, -0.63)$	3.02	2.44

We solved the reduced design equations of Table 5.22 to obtain 10 solutions; for

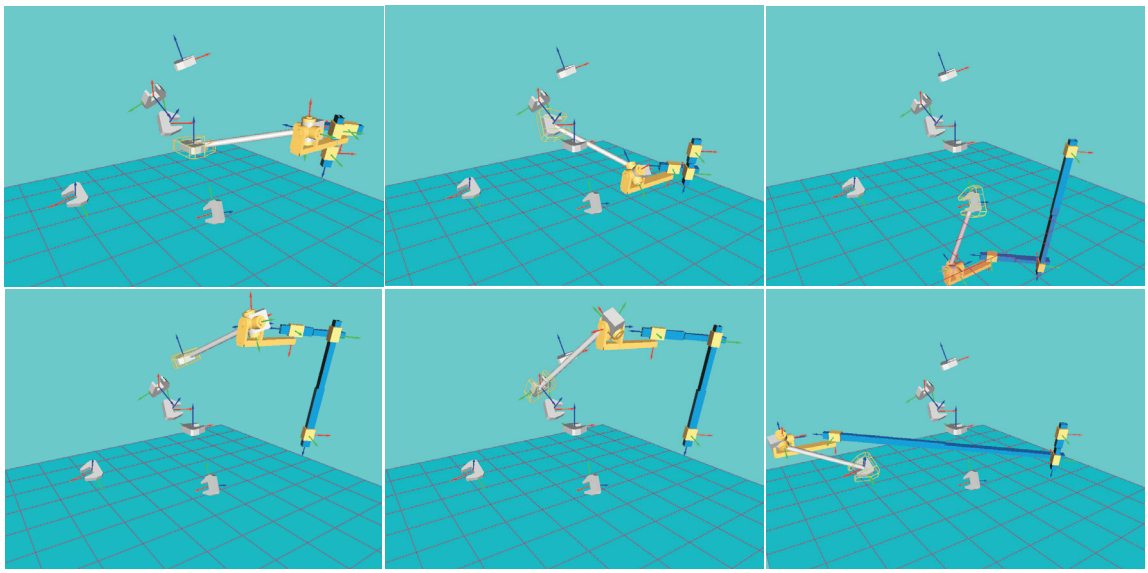

Figure 5.26: A PPS robot reaching the six task positions

this example, we have six real solutions. Figure 5.26 shows solution 4 while reaching the task positions. The complete set of solutions can be found in Appendix A.

5.5.2 The RCC robot

The RCC robot is a five-degree-of-freedom robot. The fixed revolute joint G allows rotation θ about it. It is followed along the chain by a cylindric joint $\mathsf{W}(\phi, d)$ and another cylindric joint $\mathsf{F}(\psi, b)$. See Figure 5.27.

A brief history

The equations for the RCC robot appear in the list of design equations that Tsai [97] was able to formulate using the equivalent screw triangle. He gives also the maximum number of design positions for this chain. To our knowledge, the design equations for the RCC robot have not been solved before.

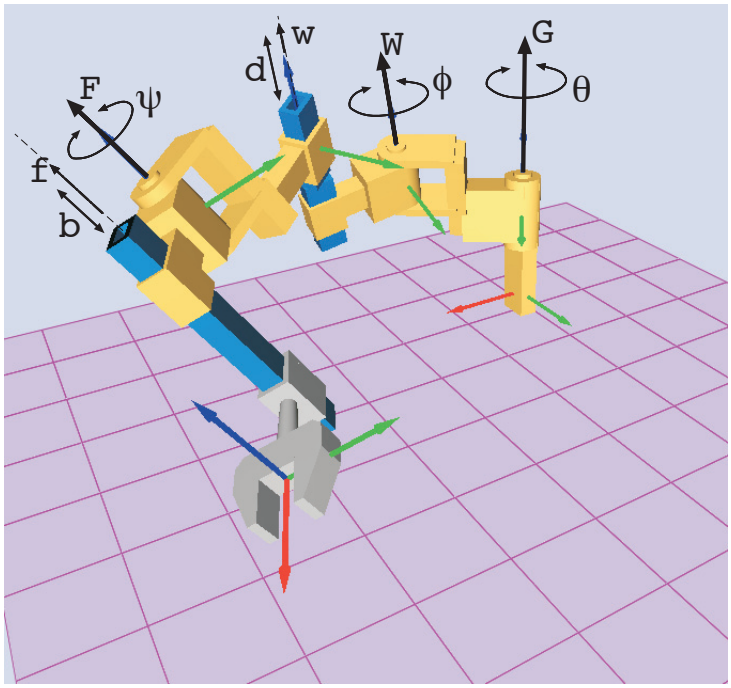

Figure 5.27: The RCC robot

The design equations

We have the kinematics equations

$$\hat{Q}_{RCC} = \hat{G}(\theta, 0)\hat{W}(\phi, d)\hat{F}(\psi, b). \tag{5.113}$$

The dual quaternion product yields

$$\begin{aligned}
Q^0 &= \cos\frac{\theta}{2}\cos\frac{\hat{\phi}}{2}\cos\frac{\hat{\psi}}{2} - \sin\frac{\theta}{2}\sin\frac{\hat{\phi}}{2}\cos\frac{\hat{\psi}}{2}\mathsf{G}\cdot\mathsf{W} - \cos\frac{\theta}{2}\sin\frac{\hat{\phi}}{2}\sin\frac{\hat{\psi}}{2}\mathsf{W}\cdot\mathsf{F} - \\
&\sin\frac{\theta}{2}\cos\frac{\hat{\phi}}{2}\sin\frac{\hat{\psi}}{2}\mathsf{G}\cdot\mathsf{F} - \sin\frac{\theta}{2}\sin\frac{\hat{\phi}}{2}\sin\frac{\hat{\psi}}{2}(\mathsf{G}\times\mathsf{W})\cdot\mathsf{F}, \\
\mathsf{Q} &= \sin\frac{\theta}{2}\cos\frac{\hat{\phi}}{2}\cos\frac{\hat{\psi}}{2}\mathsf{G} + \cos\frac{\theta}{2}\sin\frac{\hat{\phi}}{2}\cos\frac{\hat{\psi}}{2}\mathsf{W} + \cos\frac{\theta}{2}\cos\frac{\hat{\phi}}{2}\sin\frac{\hat{\psi}}{2}\mathsf{F} + \\
&\sin\frac{\theta}{2}\sin\frac{\hat{\phi}}{2}\cos\frac{\hat{\psi}}{2}(\mathsf{G}\times\mathsf{W}) + \sin\frac{\theta}{2}\cos\frac{\hat{\phi}}{2}\sin\frac{\hat{\psi}}{2}(\mathsf{G}\times\mathsf{F}) + \cos\frac{\theta}{2}\sin\frac{\hat{\phi}}{2}\sin\frac{\hat{\psi}}{2}(\mathsf{W}\times\mathsf{F}) - \\
&\sin\frac{\theta}{2}\sin\frac{\hat{\phi}}{2}\sin\frac{\hat{\psi}}{2}(\mathsf{G}\cdot\mathsf{W})\mathsf{F} + \sin\frac{\theta}{2}\sin\frac{\hat{\phi}}{2}\sin\frac{\hat{\psi}}{2}(\mathsf{G}\times\mathsf{W})\times\mathsf{F}.
\end{aligned} \tag{5.114}$$

Notice that the dual angles $\hat{\phi}$ and $\hat{\psi}$ contain the values of the translations d and b. In principle, this expression assumes that the cylindric joints W and F are such that the prismatic axis is coincident with the revolute axis in each of them. However, if we expand the product and separate the real and dual parts, we can see that only the direction of the prismatic joints appear, and hence we can use this as the general expression.

Applying Eq.(3.16) we obtain $n_{max} = 13$ positions to obtain a finite number of solutions. We equate Eqs.(5.113) to the task dual quaternions,

$$\hat{Q}_{RCC}(\theta^i, \phi^i, d^i, \psi^i, b^i) - \hat{P}^i = \vec{0}, \quad i = 2, \ldots, 13, \tag{5.115}$$

to obtain the set of parameterized design equations. It consists of 78 equations in 78 unknowns, out of which 18 are structural variables and 60 give the inverse kinematics for the joint variables.

Solving the design equations

We solved numerically the set of 72 independent parameterized design equations plus the 8 equations corresponding to Plücker constraints. We used the *fsolve* function in Matlab on a Power PC G4 at 500 MHz, with random initial conditions. After 900 runs of 10.000 iterations each, we obtained one real solution. Each complete run took 58 seconds, the total time was around 14.7 hours.

Table 5.23 contains the task positions and in Table 5.24 and Figure 5.28 we can see the obtained solution in some of the task positions.

Table 5.23: The task positions for a RCC robot

Pos.	*Dir.*	*Mom.*	*Rot.*	*Trans.*
pos. 1	$(1.0, 0.0, 0.0)$	$(0.0, 0.0, 0.0)$	0	0
pos. 2	$(0.98, 0.02, -0.18)$	$(0.00, 1.16, 0.15)$	0.90	-0.53
pos. 3	$(0.41, 0.84, 0.35)$	$(1.23, -0.47, -0.29)$	2.07	-0.93
pos. 4	$(0.52, -0.47, 0.71)$	$(-1.59, 0.41, 1.45)$	3.40	1.14
pos. 5	$(0.54, -0.74, 0.40)$	$(0.74, 0.63, 0.16)$	2.89	-0.59
pos. 6	$(0.08, 0.33, -0.94)$	$(-1.03, -1.53, -0.63)$	3.02	2.44
pos. 7	$(0.23, 0.86, -0.46)$	$(1.85, -0.65, -0.28)$	1.62	-0.05
pos. 8	$(0.60, 0.72, 0.34)$	$(0.50, -0.65, 0.49)$	1.02	1.00
pos. 9	$(-0.37, -0.93, -0.08)$	$(-2.33, 0.65, 3.00)$	2.58	0.11
pos. 10	$(0.98, -0.14, 0.16)$	$(-0.51, -1.81, 1.51)$	1.75	2.21
pos. 11	$(0.28, -0.46, -0.84)$	$(-0.01, -2.18, 1.20)$	2.34	-1.20
pos. 12	$(0.44, -0.31, 0.84)$	$(-2.61, -1.41, 0.86)$	2.82	-0.68
pos. 13	$(-0.08, -0.01, -0.996)$	$(1.09, -0.43, -0.09)$	2.83	1.85

Table 5.24: An RCC robot that reaches 13 positions

Joint Axis	*Direction*	*Moment*
G	$(-0.03, -0.97, -0.24)$	$(-7.29, -0.37, 2.50)$
W	$(0.15, 0.92, -0.36)$	$(0.99, -1.47, -3.29)$
F	$(-0.93, 0.33, -0.13)$	$(1.86, 2.76, -6.31)$

5.6 Summary

In this chapter, we presented the constrained robots that we have solved using the dual quaternion synthesis approach. For each case, we stated the design equations and

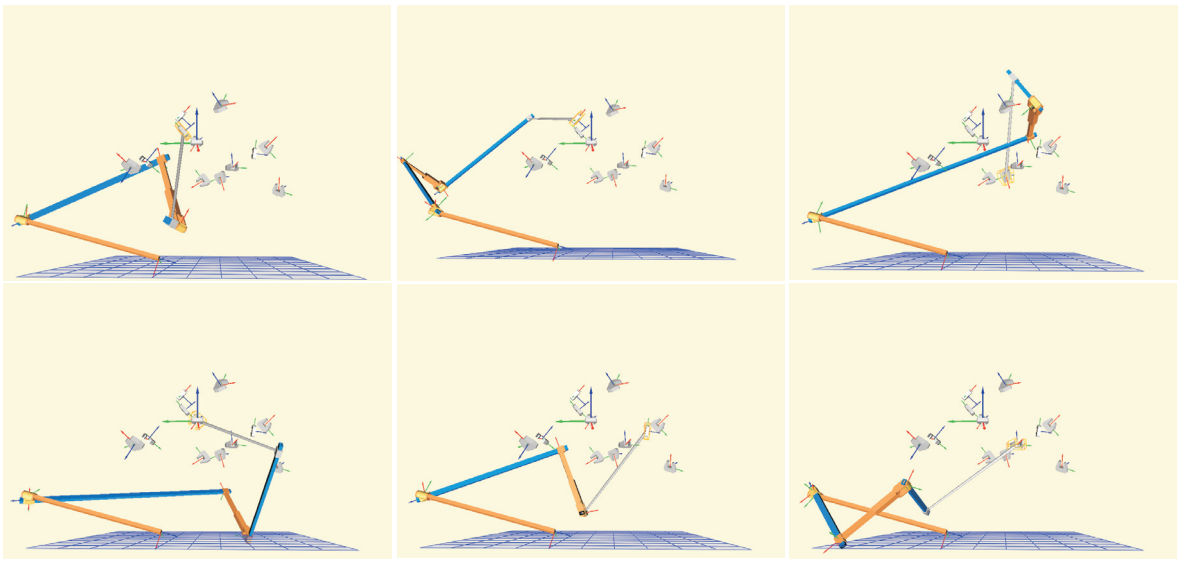

Figure 5.28: An RCC robot reaching positions 2, 5, 7, 8, 10 and 13

used different strategies to solve them. The chapter shows the growing complexity of the problem from the almost trivial case of the spatial RP chain to the most general robots with five degrees of freedom, for which we lack an efficient method to solve the equations. The different methods that we used to solve the design equations include resultants to reduce the problem to a univariate polynomial, the use of polynomial continuation algorithms, and also the standard Newton-Raphson methods to find a single solution for the problem. Both the closed algebraic solution and the polynomial homotopy continuation methods give the total number of solutions, while the set of solutions of the Newton-Raphson solver will be closer to the total number of solutions when we perform a high number of runs; the discrepance on the number of roots for different robots in this last case is also due to the fact that the Newton-Raphson algorithm only finds the real roots. This chapter illustrates the use of the theory

presented in chapters 3 and 4, and provides synthesis results for some robots that had not been synthesized before.

Chapter 6

A Simplification for the Spatial RR Robot

6.1 Overview

Three position synthesis of a spatial RR chain has been shown to yield two solutions that combine to form a spatial 4R linkage, known as Bennett's linkage. Initially, Veldkamp [103] obtained this result for three instantaneous positions. Suh [95] obtained numerical results that showed that the solution of the finite position synthesis problem also yielded two solutions that formed a Bennett linkage. Finally, Tsai and Roth [99] reduced 10 quadratic design equations to a single polynomial and showed that it always has two solutions.

Recent study of Bennett's linkage has focussed on the set of finite displacement screws that define the movement of the coupler [34]. The axes of these screws form a ruled surface known as a cylindroid.

In Chapter 5 we applied the dual quaternion methodology to the spatial RR chain to obtain a system of 12 equations in 12 unknowns (in reality, 8 equations in 8 unknowns if we use the Plücker conditions on the axes to eliminate four variables), that we solve numerically to obtain two real solutions. In this Chapter we present a methodology to reduce these equations to a univariate polynomial of third degree plus three linear equations. This methodology is based on transforming the problem to a coordinate system that captures the symmetry of the workspace of relative screw displacements of the two RR solutions, which combine to form a Bennett linkage.

This result might be generalized for simplifying the solutions of other constrained robots. However, the RR is among the simplest cases and it may be that the complexity in computing the symmetries in the workspace of more complicated cases makes the method non practical.

6.2 The Bennett Linkage

Bennett's linkage [8] moves with one degree of freedom, due to its special geometry. The mobility of a general 4R spatial linkage is obtained by applying Gruebler's criterion,

$$M = 6(n-1) - \sum_{k=1}^{m} p_k c_k = 6(4-1) - 4 \cdot 5 = -2. \tag{6.1}$$

Thus, in general, this assembly of links and joints forms a structure.

The twist angles and link lengths of the opposites sides of Bennett's linkage, (α, a)

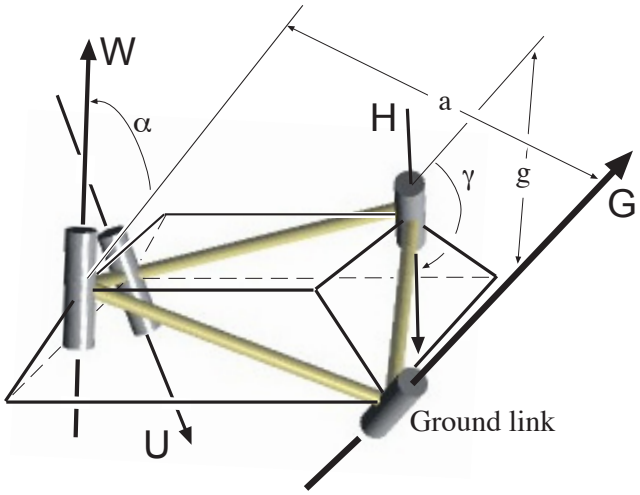

Figure 6.1: A Bennett linkage

and (γ, g), must be equal, see Figure 6.1. This, together with the condition

$$\frac{\sin\alpha}{a} = \frac{\sin\gamma}{g}, \tag{6.2}$$

ensures that the linkage moves with one degree of freedom.

6.3 The Workspace of the RR Chain

The kinematics equations for the spatial RR chain in Eq.(5.15) define the relative rotation and translation of the end-link as a screw axis S_{1i}, of dual angle $\hat{\psi}_{1i}$ for each value of the angles θ and ϕ in Figure 5.3. We use the definition of the pitch corresponding to screws representing finite displacements in [68]. By varying the joint angles θ and ϕ we obtain a set of screw axes that form a screw system of third order [33]. The workspace for the Bennett linkage is based on the same expression with the

constraint that now the joint angles are related by the expression

$$\tan\frac{\bar{\phi}}{2} = \frac{\sin\frac{\alpha+\gamma}{2}}{\sin\frac{\alpha-\gamma}{2}}\tan\frac{\bar{\theta}}{2} = K\tan\frac{\bar{\theta}}{2}, \tag{6.3}$$

where K is the constant obtained from the dimensions α and γ and $\bar{\theta}$ and $\bar{\phi}$ are absolute angles, see [37]. The condition of Eq.(6.3) translates into a condition on the relative angles $\theta = \theta_i - \theta_1$ with respect to the reference configuration,

$$\tan\frac{\phi}{2} = \frac{K\tan\frac{\theta}{2}(1+\tan^2\frac{\theta_1}{2})}{1+(K^2-1)\tan\frac{\theta_1}{2}\tan\frac{\theta}{2}+K^2(\tan^2\frac{\theta_1}{2})}. \tag{6.4}$$

We include the constraint of Eq.(6.4) in Eq.(5.15) to obtain

$$\begin{aligned}\frac{a\sin\alpha\sin\frac{\psi_{1i}}{2}}{a\sin\alpha\cos\frac{\psi_{1i}}{2}-\frac{t_{1i}}{2}\cos\alpha\sin\frac{\psi_{1i}}{2}}(1,\frac{\frac{t_{1i}}{2}}{\tan\frac{\psi_{1i}}{2}})S_{1i} = \\ = \tan\frac{\theta}{2}\big(\mathsf{G}+K_\theta\mathsf{W}^1+\tan\frac{\theta}{2}K_\theta\mathsf{G}\times\mathsf{W}^1\big),\end{aligned} \tag{6.5}$$

where

$$\begin{aligned}K_\theta &= \frac{K(1+K_1^2)}{1+K^2K_1^2+\tan\frac{\theta}{2}K_1(K^2-1)},\\ K_1 &= \tan\frac{\theta_1}{2}.\end{aligned} \tag{6.6}$$

The set of axes of the screws generated for different values of θ forms a *cylindroid*, Figure 6.2.

6.4 The Cylindroid

A cylindroid is a ruled surface that has a nodal line cutting all generators at right angles. The cylindroid appears as the axes of a real linear combination of two screws [37].

Figure 6.2: Top, side and angle views of the cylindroid.

We design the RR chain to reach the three spatial positions M_1, M_2 and M_3. The relative screw axes obtained from the design positions,

$$\mathsf{V}_a = \sin\frac{\hat{\psi}_{12}}{2}\mathsf{S}_{12}, \quad \mathsf{V}_b = \sin\frac{\hat{\psi}_{13}}{2}\mathsf{S}_{13}. \tag{6.7}$$

must lie on the cylindroid defined by Eq. (6.5), and in fact the real linear combination of these two screws generates this cylindroid.

The dual number $\sin\frac{\hat{\psi}_{1i}}{2} = (\sin\frac{\psi_{1i}}{2}, \frac{t_{1i}}{2}\cos\frac{\psi_{1i}}{2}) = \sin\frac{\psi_{1i}}{2}(1, P)$ contains the magnitude and pitch of the screw, where the pitch P is given by the expression

$$P_a = \frac{t_{12}}{2\tan\frac{\psi_{12}}{2}}, \quad P_b = \frac{t_{13}}{2\tan\frac{\psi_{13}}{2}}. \tag{6.8}$$

For reference see Parkin [68].

The two screws V_a and V_b, when independent, generate a cylindroid. We can express the cylindroid as a function of the parameters of these two screws; for the following calculations $\hat{\delta} = (\delta, d)$ is the dual angle between S_{12} and S_{13} which are the axes of V_a and V_b.

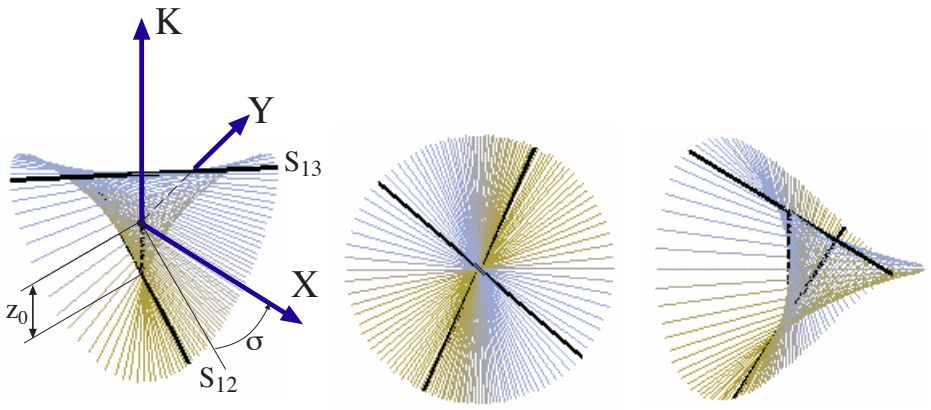

Figure 6.3: The principal axes as located from the initial screws S_{12} and S_{13}

6.4.1 The Principal Axes

The cylindroid defines a nodal line which is the common normal to all screw axes belonging to it. We can define a set of *principal axes* consisting of the nodal line and the only pair of lines of the cylindroid that intersect at a right angle. This occurs at the midpoint of the cylindroid along the nodal line [37, 69].

The expression of the cylindroid in the coordinate frame defined by the principal axes is considerably simpler. We locate the principal axes from the original relative screw V_a by using the dual angle $\hat{\sigma} = (\sigma, z_0)$.

The angle at which we locate the principal axis from V_a is σ, given by

$$\tan 2\sigma = \frac{-(P_b - P_a)\cot\delta + d}{(P_b - P_a) + d\cot\delta}. \tag{6.9}$$

This yields two angles separated by $\pi/2$. They define the directions of the principal axes X and Y of the cylindroid.

The offset z_0 measured from V_a to the principal axes is given by

$$z_0 = \frac{1}{2}(d - (P_b - P_a)\frac{\cos\delta}{\sin\delta}). \tag{6.10}$$

The principal axes of this cylindroid provide a convenient coordinate frame for the synthesis of RR chains.

6.5 Locating the Linkage

6.5.1 Bennett linkage coordinates

Yu [108] introduced a coordinate frame aimed to simplify the expression of the Bennett linkage. The joints of the Bennett linkage can be determined using a tetrahedron: each joint passes through one of the four vertices and its direction is perpendicular to the adjacent sides. The construction is showed in Figure 6.4.

Let $\mathbf{B}$, $\mathbf{P}^1$, $\mathbf{Q}$, $\mathbf{C}^1$ be the vertices of the tetrahedron. The edges are given by the difference of the vertices. The tetrahedron is oriented so that the K line forms the common normal to the lines defined by $\mathbf{B}-\mathbf{C}^1$ and $\mathbf{P}^1-\mathbf{Q}$, [34, 5]. Let $2a = |\mathbf{B}-\mathbf{C}^1|$ and $2b = |\mathbf{P}^1 - \mathbf{Q}|$, and let c and κ be the distance and angle between both edges along K.

We can completely describe the Bennett linkage in the principal axes frame using the four parameters of the tetrahedron a, b, c, κ. The coordinates of the vertices are

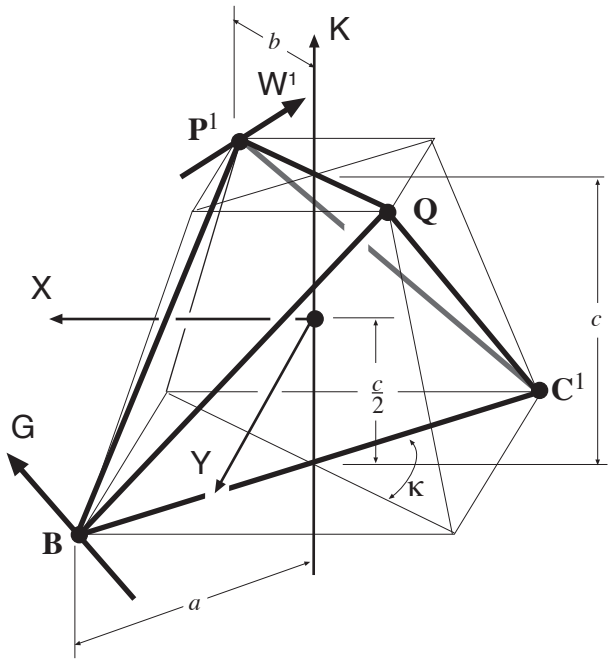

Figure 6.4: The tetrahedron that defines the Bennett linkage.

given by

$$\mathbf{B} = \begin{Bmatrix} a\cos\frac{\kappa}{2} \\ a\sin\frac{\kappa}{2} \\ \frac{-c}{2} \end{Bmatrix}, \quad \mathbf{P}^1 = \begin{Bmatrix} b\cos\frac{\kappa}{2} \\ -b\sin\frac{\kappa}{2} \\ \frac{c}{2} \end{Bmatrix},$$

$$\mathbf{Q} = \begin{Bmatrix} -b\cos\frac{\kappa}{2} \\ b\sin\frac{\kappa}{2} \\ \frac{c}{2} \end{Bmatrix}, \quad \mathbf{C}^1 = \begin{Bmatrix} -a\cos\frac{\kappa}{2} \\ -a\sin\frac{\kappa}{2} \\ \frac{-c}{2} \end{Bmatrix}. \tag{6.11}$$

To find the direction of the joint axes $\mathbf{g}$ and $\mathbf{w}^1$ we compute the cross product of the edges

$$\mathbf{g} = K_g(\mathbf{Q}-\mathbf{B}) \times (\mathbf{P}^1-\mathbf{B}) = K_g \begin{Bmatrix} 2bc\sin\frac{\kappa}{2} \\ 2bc\cos\frac{\kappa}{2} \\ 4ab\cos\frac{\kappa}{2}\sin\frac{\kappa}{2} \end{Bmatrix} \tag{6.12}$$

and

$$\mathbf{w}^1 = K_w(\mathbf{B}-\mathbf{P}^1)\times(\mathbf{C}^1-\mathbf{P}^1) = K_w\begin{Bmatrix} -2ac\sin\frac{\kappa}{2} \\ 2ac\cos\frac{\kappa}{2} \\ 4ab\cos\frac{\kappa}{2}\sin\frac{\kappa}{2}\end{Bmatrix}, \tag{6.13}$$

where the constants K_g and K_w normalize the vectors. The expressions of the screws $\mathsf{G} = (\mathbf{g}, \mathbf{B}\times\mathbf{g})^T$ and $\mathsf{W}^1 = (\mathbf{w}^1, \mathbf{P}^1\times\mathbf{w}^1)^T$ in the principal axis coordinates are

$$\mathsf{G} = K_g\begin{Bmatrix} 2bc\sin\frac{\kappa}{2} \\ 2bc\cos\frac{\kappa}{2} \\ 4ab\cos\frac{\kappa}{2}\sin\frac{\kappa}{2}\end{Bmatrix} + \epsilon K_g\begin{Bmatrix} b\cos\frac{\kappa}{2}(4a^2\sin^2\frac{\kappa}{2}+c^2) \\ -b\sin\frac{\kappa}{2}(4a^2\cos^2\frac{\kappa}{2}+c^2) \\ 2abc(\cos^2\frac{\kappa}{2}-\sin^2\frac{\kappa}{2})\end{Bmatrix} \tag{6.14}$$

and

$$\mathsf{W}^1 = K_w\begin{Bmatrix} -2ac\sin\frac{\kappa}{2} \\ 2ac\cos\frac{\kappa}{2} \\ 4ab\cos\frac{\kappa}{2}\sin\frac{\kappa}{2}\end{Bmatrix} + \epsilon K_w\begin{Bmatrix} -a\cos\frac{\kappa}{2}(4b^2\sin^2\frac{\kappa}{2}+c^2) \\ -a\sin\frac{\kappa}{2}(4b^2\cos^2\frac{\kappa}{2}+c^2) \\ 2abc(\cos^2\frac{\kappa}{2}-\sin^2\frac{\kappa}{2})\end{Bmatrix}. \tag{6.15}$$

Similarly, the coordinates of the second RR solution, which we denote by H and U^1, are given by the expressions

$$\begin{aligned}\mathsf{H} &= K_h(\mathbf{C}^1-\mathbf{Q})\times(\mathbf{B}-\mathbf{Q}) + \epsilon K_h\mathbf{Q}\times\big((\mathbf{C}^1-\mathbf{Q})\times(\mathbf{B}-\mathbf{Q})\big) \\ \mathsf{U}^1 &= K_u(\mathbf{P}^1-\mathbf{C}^1)\times(\mathbf{Q}-\mathbf{C}^1) + \epsilon K_u\mathbf{C}^1\times\big((\mathbf{P}^1-\mathbf{C}^1)\times(\mathbf{Q}-\mathbf{C}^1)\big).\end{aligned} \tag{6.16}$$

6.5.2 The tetrahedron and the cylindroid

The tetrahedron completely defines the Bennett linkage. In order to locate the tetrahedron with respect to the given relative positions,with screw axes S_{12} and S_{13}, we need to relate the tetrahedron to the principal axes of the cylindroid. In Appendix 7.2, we show that the principal axes $\{\mathsf{X}, \mathsf{Y}, \mathsf{K}\}$ are located in the midpoint of the tetra-

hedron and bisecting the angle κ [72], as seen in Figure 6.4. There we also include the coordinate transformation from the initial frame to the principal axes frame.

6.5.3 Transforming the design equations

The design equations for the RR chain defined in Section 5.2.2 are equivalent to those obtained from the geometric constraints imposed by the link connecting the moving and fixed axes as defined in [98, 96, 54], and also equivalent to the screw triangle formulation of Tsai [97], see Section 3.8.2.

By using the Bennett linkage coordinates to define G and W^1 as in Eq.(6.14) and Eq.(6.15), we reduce the number of design parameters in the design equations from ten to four. The result is four equations in the unknowns a, b, c and κ.

6.6 Solving the Design Equations

We now transform the task positions $[T_i]$ to the principal axis frame and determine the relative screw axes

$$\begin{aligned} \mathsf{S}_{12} &= \sin\frac{\hat{\psi}_{12}}{2}(\cos\hat{\delta}_1\mathsf{X} + \sin\hat{\delta}_1\mathsf{Y}) \\ \mathsf{S}_{13} &= \sin\frac{\hat{\psi}_{13}}{2}(\cos\hat{\delta}_2\mathsf{X} + \sin\hat{\delta}_2\mathsf{Y}), \end{aligned} \tag{6.17}$$

where $\hat{\psi}_{1i} = (\psi_{1i}, t_{1i})$ are the angle and translation of the screw displacements. The dual angles $\hat{\delta}_i$ are defined relative to the principal axes frame.

Substitute Eq.(6.17) in the equation created from the part corresponding to the

translation in Eq.(5.15) to obtain two linear equations in a and b,

$$\frac{t_{12}}{2} + (a-b)\cos\delta_1 \cos\frac{\kappa}{2} + (a+b)\sin\delta_1 \sin\frac{\kappa}{2} = 0$$
$$\frac{t_{13}}{2} + (a-b)\cos\delta_2 \cos\frac{\kappa}{2} + (a+b)\sin\delta_2 \sin\frac{\kappa}{2} = 0.$$

They can be solved to obtain

$$a = \frac{K_s}{2\sin\frac{\kappa}{2}} + \frac{K_d}{2\cos\frac{\kappa}{2}},$$
$$b = \frac{K_s}{2\sin\frac{\kappa}{2}} - \frac{K_d}{2\cos\frac{\kappa}{2}}. \tag{6.18}$$

The constants K_s and K_d combine information from the relative screws and are listed in Table 6.1.

Next we substitute Eq.(6.18) into the direction part of Eq.(5.15) and introduce the definition of $y = \tan\frac{\kappa}{2}$ to eliminate the sine and cosine functions of κ. The result is two rational equations in c and y,

$$\frac{\tan\frac{\psi_{12}}{2}(\frac{K_s^2}{K_d^2} - y^2) + c^2\frac{\tan\frac{\psi_{12}}{2}}{2K_d^2}(y^2(\cos 2\delta_1 - 1) + \cos 2\delta_1 + 1) - 2\frac{cy}{K_d}(\cos\delta_1 + \frac{K_s\sin\delta_1}{K_d})}{(\frac{K_s^2}{K_d^2} - y^2) + \frac{c^2}{2K_d^2}(y^2(\cos 2\delta_1 - 1) + \cos 2\delta_1 + 1)} = 0, \tag{6.19}$$

$$\frac{\tan\frac{\psi_{13}}{2}(\frac{K_s^2}{K_d^2} - y^2) + c^2\frac{\tan\frac{\psi_{13}}{2}}{2K_d^2}(y^2(\cos 2\delta_2 - 1) + \cos 2\delta_2 + 1) - 2\frac{cy}{K_d}(\cos\delta_2 + \frac{K_s\sin\delta_2}{K_d})}{(\frac{K_s^2}{K_d^2} - y^2) + \frac{c^2}{2K_d^2}(y^2(\cos 2\delta_2 - 1) + \cos 2\delta_2 + 1)} = 0, \tag{6.20}$$

The numerator and denominator share two roots associated with $c = 0$, that is

$$c = 0, y = \pm\frac{K_s}{K_d}, \tag{6.21}$$

Table 6.1: Constants computed from the task positions for the RR robot.

Constant	Expression
K_s	$\frac{t_{12}\cos\delta_2 - t_{13}\cos\delta_1}{2\sin(\delta_1-\delta_2)}$
K_d	$\frac{t_{13}\sin\delta_1 - t_{12}\sin\delta_2}{2\sin(\delta_1-\delta_2)}$
K_{12}	$\frac{t_{12}/2}{\tan\frac{\psi_{12}}{2}}\left(\frac{1}{\sin^2\delta_1-\sin^2\delta_2}\right)$
K_{13}	$\frac{t_{13}/2}{\tan\frac{\psi_{13}}{2}}\left(\frac{1}{\sin^2\delta_1-\sin^2\delta_2}\right)$

which we eliminate them from the numerator forcing the following linear system to have more solutions than the trivial,

$$\begin{bmatrix} \tan\frac{\psi_{12}}{2} & \frac{\tan\frac{\psi_{12}}{2}}{2K_d^2}(y^2c(\cos 2\delta_1 - 1) + \cos 2\delta_1 + 1) - 2\frac{y}{K_d}(\cos\delta_1 + \frac{K_s\sin\delta_1}{K_d}) \\ \tan\frac{\psi_{13}}{2} & \frac{\tan\frac{\psi_{13}}{2}}{2K_d^2}(y^2c(\cos 2\delta_2 - 1) + \cos 2\delta_2 + 1) - 2\frac{y}{K_d}(\cos\delta_2 + \frac{K_s\sin\delta_2}{K_d}) \end{bmatrix} \begin{Bmatrix} \frac{K_s^2}{K_d^2} - y^2 \\ c \end{Bmatrix} = 0. \tag{6.22}$$

In order for this set of equation to have roots other than those given by Eq.(6.21), the determinant of the coefficient matrix must be zero. This yields an equation that is linear in c. Solve this to obtain

$$c = (K_{13} - K_{12})\sin\kappa, \tag{6.23}$$

where the constants K_{12} and K_{13} are shown in Table 6.1.

In order to determine κ, we substitute the expressions for a, b and c into one of the direction equations in Eq.(5.15). We obtain a cubic polynomial in y^2,

$$P: \quad C_3y^6 + C_2y^4 + C_1y^2 + C_0 = 0 \tag{6.24}$$

with the coefficients given by

$$
\begin{aligned}
C_3 &= -K_d^2, \\
C_2 &= K_s^2 - 2K_d^2 + 4(K_{12} - K_{13})(K_{13}\sin^2\delta_1 - K_{12}\sin^2\delta_2), \\
C_1 &= 2K_s^2 - K_d^2 - 4(K_{12} - K_{13})(K_{13}\cos^2\delta_1 - K_{12}\cos^2\delta_2), \\
C_0 &= K_s^2.
\end{aligned} \tag{6.25}
$$

Substitute $x = y^2$, and solve the cubic polynomial to determine its three roots. We can show that the polynomial $P(x)$ has only one positive real root for x, which we denote $x^\star$. To do this we compute the values,

$$
\begin{aligned}
P(0) &= K_s^2, \\
P(-1) &= -4(K_{12} - K_{13})^2, \\
P(+x_\infty) &\approx -K_d^2(+x_\infty)^3, \\
P(-x_\infty) &\approx -K_d^2(-x_\infty)^3.
\end{aligned} \tag{6.26}
$$

Notice that $P(0) > 0$, while for large positive values of x we have $P(+x_\infty) < 0$. From this we conclude that the polynomial has at least one positive real root. Furthermore, $P(-1) < 0$ for all values of its coefficients, and for large negative values of x the polynomial is positive, $P(-x_\infty) > 0$. This leads to the conclusion that there are at least two negative real roots. The cubic polynomial only has three roots, therefore one is positive and two are negative. This gives in turn two real roots for the original polynomial equation.

Table 6.2: The two sets of solutions for the Bennett linkage

Solution 1		**Solution 2**
$\mathsf{G}(a,\, b,\, c,\, \kappa)$	$=$	$\mathsf{H}(-b, -a, -c, -\kappa)$
$\mathsf{W}(a,\, b,\, c,\, \kappa)$	$=$	$\mathsf{U}(-b, -a, -c, -\kappa)$
$\mathsf{H}(a,\, b,\, c,\, \kappa)$	$=$	$\mathsf{G}(-b, -a, -c, -\kappa)$
$\mathsf{U}(a,\, b,\, c,\, \kappa)$	$=$	$\mathsf{W}(-b, -a, -c, -\kappa)$

To determine κ, we compute the square root of the single positive real root $x^\star$,

$$\tan\frac{\kappa}{2} = \pm\sqrt{x^\star}. \tag{6.27}$$

The result is two sets of solutions (a, b, c, κ) and $(-b, -a, -c, -\kappa)$.

The design procedure yields two unique algebraic solutions for the three position synthesis of the RR chain. According to the location on the tetrahedron, the second solution, given by the values $(-b, -a, -c, -\kappa)$, corresponds to the coordinates of the RR chain associated with the other two axes of the Bennett linkage, as defined in Eq.(6.16) and as can be seen in Table 6.2. Hence this procedure yields two RR chains, which combine to form a Bennett linkage.

6.7 Summary

In this chapter we developed a method to simplify the set of design equations obtained in Chapter 5 for the spatial RR chain, to obtain a closed algebraic solution. We performed a change of coordinates that simplified the original set of twelve nonlinear equations in twelve unknowns to three linear equations plus a third-degree polynomial.

The new reference frame for the problem is called *Bennett linkage coordinates* and is located at the principal axes of the workspace of relative screws generated by the Bennett linkage defined by the two real solutions of the RR problem.

We defined the workspace of the Bennett linkage from the dual quaternion workspace of the RR chain with a given relation between both joint variables, and identified this with the cylindroid generated by three task positions. We proved that this cylindroid lies in the center of the tetrahedron defined by the Bennett linkage, and calculated the transformation from the original frame. When the design equations were transformed to the new frame, they simplified to four equations that can be solved algebraically.

Chapter 7

Conclusions and Future Research

7.1 Contributions

In this dissertation we presented a methodology for the finite position synthesis of constrained robots based on the expression of the kinematics equations in dual quaternion form.

We developed the relation between the standard kinematics equations in 4×4 homogeneous matrix form and that in dual quaternion form, and showed that we can derive the dual quaternion form to the matrix form by transforming to relative displacments, to obtain successive screw displacements, which turns out to be equivalent to the product of exponentials based on Lie algebra methods.

The dual quaternion kinematics equations are equated to a set of task position expressed as dual quaternions. The number of task positions that we can define to obtain a finite number of robots needs to be known. We were able to create formulas, based on the parameters of each joint and the number of joint variables, to count

for the maximum number of positions that can be defined for each robot topology. This formula computes the maximal set of general positions that can be fitted in the surface that defines the workspace of the robot. Designing for a smaller number of task positions is also possible and leads to an infinity of solutions.

We identified special cases in which the robot is orientation-limited and translation-unconstrained, due to the semi-direct product structure of the group of spatial displacements. Orientation-limited robots can reach a certain number of orientations that is less than the maximum number of translations that they can reach. This can be solved with two different approaches: extra constraints can be defined so that the robot can reach at most as many translations as orientations; or, in addition to a certain number of general task positions, an extra number of task positions can be defined in which the translation is general but the orientation belongs to the workspace of the chain. These orientations can be computed from the chain or can be approximated from arbitrarily specified positions. We also identified the translation-unconstrained robots, those orientation-limited robots that are not constrained in the translations.

The design equations obtained by equating the dual quaternion kinematics equations to the set of task positions are parameterized by the joint variables, in the sense that, assigning the whole range of values to the joint variables, we obtain the workspace of the robot for the given values of the parameters of the axes. The parameterized equations contain trigonometric functions and can be solved numerically. We have found that, for some algorithms, it is preferred to have a higher number of

equations of lower degree, and this is the case of the parameterized equations, while others are better with a smaller number of equations of higher degree.

We devised a methodology to eliminate these non-structural joint variables, to obtain a set of algebraic equations that, in the simplest cases, have been proved to be equivalent to the equations obtained stating the geometric constraints of the chain or using the equivalent screw triangle. The method consists of eliminating linearly a combination of the joint variables, and creating reduced design equations by imposing the relations among the variables we are solving for. The goal of this elimination is, on one hand, to create equations that may be easier to solve numerically using certain methods, and, on the other hand, to work toward a closed algebraic solution in those cases where it is possible.

We have solved the design equations using different methods, depending on their complexity. For the most complex cases, numerical algorithms based on Newton-Raphson methods have been applied to the parameterized equations. These numerical algorithms have been also applied to reduced sets of equations. In cases of relative complexity, we have applied polynomial homotopy continuation methods. Finally, in some cases, we have been able to find closed algebraic solutions by using mainly resultant methods.

The work presented in Chapter 6 is a different methodology to simplify the design equations, applied to the spatial RR chain. It is basically a change of coordinates that simplifies the original set of twelve nonlinear equations in twelve unknowns to

three linear equations plus a third-degree polynomial.

We studied a reference frame that captures the symmetry of the solution by looking at the screws that generate the workspace of relative displacements of the chain; the new coordinates for the problem is called *Bennett linkage coordinates* and the reference frame is located at the principal axes of the screw system generated by the movement of the Bennett linkage, which is defined by the two real solutions of the RR problem.

We defined the workspace of the Bennett linkage from the dual quaternion workspace of the RR chain with a given relation between both joint variables, and identified this with the cylindroid generated by the three task positions. We computed the principal axes of the cylindroid and proved that this cylindroid lies in the center of the tetrahedron defined by the Bennett linkage, then calculated the transformation from the original frame. When the design equations are transformed to the new frame, they simplify to four equations that we solved algebraically.

7.2 Future Research

Future research can be developed in three different directions. The actual synthesis equations obtained with the dual quaternion methodology are still too complicated to be solved directly in the most complex cases, that is, those of five-degree-of-freedom robots with no additional constraints. Some work needs to be done to characterize the dimension of the solution set of these equations, and also to simplify them and

to devise solution methods.

A second direction is in the synthesis of parallel robots. Dimensional synthesis of parallel robots can be done by assembling together the different solutions of the synthesis of serial chains. The dimensional synthesis methodology could be combined with type synthesis methods to create a way to dimension a given workspace.

The last direction is the application of the methodology to solve different problems. The immediate application is in software for the design of spatial mechanisms, and for that some work is required to ensure that the solution methods are suited for computer programming, and also to define routines for choosing among solutions from a finite or infinite set based on some metrics. Other applications may include the identification of the structure of actual systems based on data about their movement in space, for instance the identification of skeletons in motion capture or the structure of molecules such as proteins.

Bibliography

[1] Al-widyan, K. and Angeles, J., 2001, "Uncertainty, isotropy, and optimality in the design of robotic mechanical systems", *Proc. International Conference on Multidisciplinary Design in Engineering*, Montreal, Que., Nov. 21-22, 2001.

[2] Angeles, J., 1982, *Spatial Kinematic Chains. Analysis, Synthesis, Optimization*, Springer-Verlag, New York.

[3] Angeles, J., 1998, "The application on dual algebra to kinematic analysis," *Computational Methods in Mechanical Systems, NATO ASI Series* (ed. J. Angeles and E. Zakhariev) Springer, Berlin.

[4] Artin, M., 1991,*Algebra*, Prentice-Hall, Upper Saddle River, NJ.

[5] Baker, J.E., 1998, "On the Motion Geometry of the Bennett Linkage." *Proc. 8th Internat. Conf. on Engineering Computer Graphics and Descriptive Geometry*, Austin, Texas, USA, **2**:433-437.

[6] Baker, J.E., and Parker, I.A., 2003, *Fundamentals of Screw Motion: Seminal papers by Michel Chasles and Olinde Rodrigues*, School of Information Technologies, University of Sydney.

[7] Ball, R.S., 1900, *A Treatise on the Theory of Screws*, Cambridge University Press, Cambridge, First published 1900, paperback edition 1998.

[8] Bennett, G.T., 1903, "A New Mechanism", *Engineering*, **76**:777-778.

[9] Bottema, O., and Roth, B., 1979, *Theoretical Kinematics*, North Holland Press, NY.

[10] Chedmail, P., 1998, "Optimization in Multi-DOF Mechanisms," *Computational Methods in Mechanical Systems, NATO ASI Series* (ed. J. Angeles and E. Zakhariev) Springer, Berlin.

[11] Clifford, W., 1873, "Preliminary sketch of bi-quaternions", *Proc. London Math. Soc.*, **4**:381395.

[12] Chen, P. and Roth, B., 1967, "Design Equations for Finitely and Infinitesimally Separated Position Synthesis of Binary Link and Combined Link Chains," *ASME J. Engineering for Industry* 91:209-219.

[13] Craig, J. J., 1989, *Introduction to Robotics, Mechanics and Control*, Addison Wesley Publ. Co.

[14] Cox, D., Little, J. and O'Shea, D., 1998, *Using Algebraic Geometry*, Springer, New York.

[15] Curtis, C.L., 1997, "Singularity Analysis and Design of Parallel Manipulators", emphPh.D. Thesis, Department of Mechanical and Aerospace Engineering, University of California, Irvine.

[16] Curtis, C.L., and McCarthy, J.M., 1998, "The Quartic Singularity Surfaces of Planar Platforms in the Clifford Algebra of the Projective Plane", *Mechanism and Machine Theory*, **33**(7):931-944.

[17] Gosselin, C. M., 1998, "On the design of efficient parallel mechanisms," *Computational Methods in Mechanical Systems, NATO ASI Series* (ed. J. Angeles and E. Zakhariev) Springer, Berlin.

[18] Daniilidis, K., and Bayro-Corrochano, E., 1996,"The dual quaternion approach to hand-eye calibration", *Proc. of the 13th Int. Conf. on Pattern Recognition*, 1996,**1**: 318 -322.

[19] De Sa, S., and Roth, B., 1981, "Kinematic Mappings. Part2: Rational Algebraic Motions in the Plane", *ASME Journal of Mechanical Design*, **103**:712-717.

[20] Denavit, J., and Hartenberg, R.S., 1955, "A Kinematic Notation for Lower-pair Mechanisms Based on Matrices", *ASME Journal of Applied Mechanics*, **22**:215-221.

[21] Dimarogonas, A.D., 1991,"The Origins of theTheory of Machines and Mechanisms", *Modern Kinematics. Developments in the last forty years.*, Arthur G. Erdman, ed., New York, 1993.

[22] Dimentberg, F.,1965, "The Screw Calculus and Its Applications in Mechanics", U.S. Dept. of Commerce. Translation No. AD680993.

[23] Dooley, J. R. and McCarthy, J. M., 1991, "Spatial Rigid Body Dynamics Using Dual Quaternion Components", *Proc. of IEEE International Conf. on Robotics and Automation*, Sacramento, CA, April 1991, pp.90-95.

[24] Dorst, L., and Mann, S., 2002, "Geometric Algebra: A Computational Framework for Geometrical Applications (Part 1)", *IEEE Computer Graphics and Applications*, **22**(3): 24-31.

[25] Gupta, K.G., and Roth, B., 1982, "Design Considerations for Manipulator Workspace", *ASME Journal of Mechanical Design*, **104**(4):704-711.

[26] Gupta, K.C., 1986, "Kinematic Analysis of Manipulators Using Zero Reference Position Description", *Int. J. Robot. Res.*, 5(2):5-13

[27] Su, H.-J. and McCarthy J.M, 2003, "Kinematic Synthesis of RPS Serial Chains", *ASME Design Engineering Technical Conferences*, Sept.02-06, Chicago, IL.

[28] Hamilton, W.R., 1899, *Elements of Quaternions*, Chelsea Publishing Company, New York, NY, 3rd edition, 1969.

[29] Hao, K., 1998, "Dual Number Method, Rank of a Screw System and Generation of Lie Sub-Algebras," *Mechanism and Machine Theory*, 33(7): 1063-1084.

[30] Hartenberg, R., and Denavit, J., 1964, *Kinematic Synthesis of Linkages*, McGraw-Hill.

[31] Horsch, Th., and Nolzen, H., 1992, "Local Motion Planning Avoiding Obstacles with Dual Quaternions", *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Nice, France, May 1992.

[32] Huang, C. and Roth, B., 1994, "Analytic Expressions for the Finite Screw Systems", *Mechanism and Machine Theory*, **29**(2):207-222.

[33] Huang, C., 1994, "On the Finite Screw System of the Third Order Associated with a Revolute-Revolute Chain" *Journal of Mechanical Design*, **116**:875-883.

[34] Huang, C., 1997, "The Cylindroid Associated with Finite Motions of the Bennett Mechanism", *ASME Journal of Mechanical Design*, **119**:521-524.

[35] Huang, C . and Chang, Y-J., 2000, "Polynomial Solution to the Five-Position Synthesis of Spatial C-C Dyads via Dialytic Elimination", *Proc. ASME Design Engineering Technical Conference*, Paper No. DETC2000/MECH-14102, Baltimore, Maryland, Sept. 10-13.

[36] Huang, C., and Wang, J.C., 2003, "The Finite Screw System Associated with the Displacement of a Line", *ASME Journal of Mechanical Design*, **125**(1):105-109.

[37] Hunt, K.H., 1978, *Kinematic Geometry of Mechanisms*. Clarendon Press.

[38] Innocenti, C., 1995, "Polynomial Solution of the Spatial Burmester Problem", *ASME J. Mech. Design* 117(1).

[39] Kim, H. S., and Tsai, L. W., 2002, "Kinematic Synthesis of Spatial 3-RPS Parallel Manipulators," *Proc. ASME Des. Eng. Tech. Conf.* paper no. DETC2002/MECH-34302, Sept. 29-Oct. 2, Montreal, Canada.

[40] Krovi, V., Ananthasuresh, G.K. and Kumar, V., 1999, "Kinematic Synthesis of Spatial R-R Dyads for Path Following Revisited Using the Rotation Matrix Approach", *Proc. of the ASME Design Engineering Technical Conferences*, Las Vegas, Nevada, USA, September 12-15, 1999.

[41] Kumar, A., and Waldron, K.J., 1981, "The Workspace of a Mechanical Manipulator", *ASME J. of Mechanical Design*, **103**: 665-672.

[42] Larochelle, P., and McCarthy J. M., 1995, "Designing Planar Mechanisms using a Bi-invariant Quaternion Metric", *ASME Journal of Mechanical Design*, **117**(4):646-651.

[43] Larochelle, P., 2000, "Approximate motion synthesis via parametric constraint manifold fitting," *Advances in Robot Kinematics* (eds. J. Lenarcic and M. M. Stanisic) Kluwer Acad. Publ., Dordrecht.

[44] Lee, K.W., and Yoon, Y.S., 1993, "Kinematic Synthesis of RRSS Spatial Motion Generators Using Euler Parameters and Quaternion Algebra", *Proc. of the IMEC part C: J. of Mech. Eng. Science*, **207**:355-359.

[45] Lee, E., Mavroidis, C., and Merlet, J.P., 2002a, "Five Precision Points Synthesis of Spatial RRR Manipulators Using Interval Analysis," *Proc. ASME 2002 Design Eng. Tech. Conf.*, paper no. DETC2002/MECH-34272, Sept. 29-Oct. 2, Montreal, Canada.

[46] Lee, E., and Mavroidis, D., 2002,b "Solving the Geometric Design Problem of Spatial 3R Robot Manipulators Using Polynomial Homotopy Continuation," *ASME J. Mechanical Design*, 124(4):652-661.

[47] Lee, E., and Mavroidis, D., 2002c, "Geometric Design of Spatial PRR Manipulators Using Polynomial Elimination Techniques," *Proc. ASME 2002 Design Eng. Tech. Conf.*, paper no. DETC2002/MECH-34314, Sept. 29-Oct. 2, Montreal, Canada.

[48] Liao, Q. and McCarthy, J. M., 2001, "On the Seven Position Synthesis of a 5-SS Platform Linkage," *ASME J. Mechanical Design*, 123(1):74-79.

[49] Manocha, D., 1993, "Solving Polynomial Systems for Curve, Surface and Solid Modeling", *Solid Modeling and Applications*, 1993:169-178.

[50] Manocha, D., and Demmel, J., 1994, "Algorithms for Intersecting Parametric and Algebraic Curves I: Simple Intersections", *ACM Transactions on Graphics*, **13**(1):73-100.

[51] Martinez, J. M. R., and Duffy, J., 1995 "On the metrics of rigid body displacements for infinite and finite bodies". *ASME J. of Mechanical Design*, **117**(1):41-47.

[52] Mavroidis, C., Lee, E., and Alam, M., 2001, "A New Polynomial Solution to the Geometric Design Problem of Spatial RR Robot Manipulators Using the Denavit-Hartenberg Parameters," *ASME J. Mechanical Design*, 123(1):58-67.

[53] McCarthy, J. M., 1990, *Introduction to Theoretical Kinematics*, The MIT Press.

[54] McCarthy, J. M., 1995, "The Synthesis of Planar RR and Spatial CC Chains and the Equation of a Triangle," Special Combined Issue of theASME J. of Mechanical Design and J. of Vibration and Acoustics, 117(B):101-106.

[55] McCarthy, J.M., 1993, "Dual Quaternions and the Pole Triangle", *Modern Kinematics. Developments in the last 40 years.*, Arthur G. Erdman, editor. John Wiley and Sons, New York.

[56] McCarthy, J.M., and Ahlers, S.G., 1999, "Dimensional Synthesis of Robots using a Double Quaternion Formulation of the Workspace", *9th. International Symposium of Robotic Research ISSR'99*. Snowbird, Utah, October 1999.

[57] McCarthy, J. M., and Ahlers, S., 2000, "Dimensional Synthesis of Robots using a Double Quaternion Formulation of the Workspace," *Robotics Research: The Ninth International Symposium*, (J. M. Hollerbach and D. E. Koditschek, editors), pp. 3-8, Springer-Verlag, 2000.

[58] McCarthy, J. M., 2000, *Geometric Design of Linkages*, Springer-Verlag, New York.

[59] Merlet, J.P., 1997, "Designing a parallel manipulator for a specific workspace", *Int. J. of Robotics Research*, **16**(4):545-556.

[60] Morgan, A. P, and Sommese, A. J, 1989, "Coefficient Parameter Polynomial Continuation," *Appl. Mat. and Comput.*, **29**:123-160.

[61] Murray, A., Pierrot, F., Dauchez, P., and McCarthy, J., 1996, "On the Design of Parallel Manipulators for a Prescribed Workspace: a Planar Quaternion Approach", *5th International Symposium on Advances in Robot Kinematics*, June, 1996.

[62] Murray, A., and Hanchak, M. 2000, "Kinematic Synthesis of Planar Platforms with RPR, PRR and RRR Chains", *Advances in Robot Kinematics* (eds. J. Lenarcic and M. M. Stanisic) Kluwer Acad. Publ., Dordrecht.

[63] Murray, R. M., Li, Z., and Sastry, S. S., 1994, *A Mathematical Introduction to Robotic Manipulation*, CRC Press.

[64] Nielsen, J. and Roth, B., 1995, "Elimination Methods for Spatial Synthesis," *Computational Kinematics*, (eds. J. P. Merlet and B. Ravani), Vol. 40 of *Solid Mechanics and Its Applications*, pp. 51-62, Kluwer Academic Publishers.

[65] Park, F. C., 1995, "Distance metrics on the rigid body motions with applications to mechanism design". *ASME J. of Mechanical Design*, **117**(1):48-54.

[66] Park, F. C., and Bobrow, J. E., 1995, "Geometric Optimization Algorithms for Robot Kinematic Design". *Journal of Robotic Systems*, **12**(6):453-463.

[67] Park, F.C., Bobrow, J.E., and Ploen, S.R., 1995 "A Lie group formulation of robot dynamics." *Int. J. Robotics Research*, **14**(6):609-618.

[68] Parkin, I.A., 1992, "A Third Conformation with the Screw Systems: Finite Twist Displacements of a Directed Line and Point".*Mechanisms and Machine Theory*, **27**:177-188.

[69] Parkin, I.A., "Finding the Principal Axes of Screw Systems." *Proceedings of the ASME Design Engineering Technical Conferences*, Sacramento, CA, 1997.

[70] Perez, A., and McCarthy, J.M., 2003, "Dimensional Synthesis of Bennett Linkages", *ASME Journal of Mechanical Design*, 125(1):98-104.

[71] Perez, A. and McCarthy, J. M., 2002, "Dual Quaternion Synthesis of Constrained Robots," *Advances in Robot Kinematics*, (J. Lenarcic and F. Thomas, eds.) Kluwer Academic Publ. 443-454. Caldes de Malavella, Spain, June 24-29.

[72] Perez, A., and McCarthy, J.M., 2002, Bennetts Linkage and the Cylindroid, *Mechanism and Machine Theory*, **37**(11), pp. 1245-1260.

[73] Perez, A. and McCarthy, J.M., 2003, "Dual Quaternion Synthesis of Constrained Robotic Systems", *submitted to Journal of Mechanical Design*, March 2003.

[74] Perez, A. and McCarthy, J.M., 2003, "Dimensional Synthesis of CRR Serial Chains", *ASME Design Engineering Technical Conferences*, Chicago, IL, September 2003.

[75] Perez A., and McCarthy, J. M., 2002, "Dual Quaternion Synthesis of a Parallel 2-TPR Robot," *Proc of the Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators*, October 3-4, 2002, Quebec City, Quebec, CA.

[76] Primrose, E., Freudenstein, F., and Sandor, G.N., 1964, "Finite Burmester Theory in Plane Kinematics", *ASME J. of Applied Mechanics*, **86**:683-693.

[77] Ravani, B., and Roth, B., 1983, "Motion Synthesis Using Kinematic Mappings", *J. of Mechanisms, Transmission and Automation in Design*, **105**: 460-467.

[78] Ravani, B. and Ge Q. J.,1991, "Kinematic Localization for World Model Calibration in Off-Line Robot Programming Using Clifford Algebra", *Proc. of IEEE International Conf. on Robotics and Automation*, Sacramento, CA, April 1991, pp. 584-589.

[79] Releaux, F., 1875, *The Kinematics of Machinery: Outlines of a Theory of Machines*, Dover Publications, New York, translation of 1963.

[80] Rico, J.M., and Ravani, B., 2003, "On Mobility Analysis of Linkages Using Group Theory", *ASME J. of Mechanical Design*, **125**(1):70-80.

[81] Roth, B., 1967, "On the screw axis and other special lines associated with spatial displacements of a rigid body", *ASME J. Eng. Ind.*, **89**:102-110.

[82] Roth, B., 1967, "Finite Position Theory Applied to Mechanism Synthesis", *Journal of Applied Mechanics*, **34**E:599-605.

[83] Roth, B., 1968, "The design of binary cranks with revolute, cylindric, and prismatic joints", *J. Mechanisms*, 3(2):61-72.

[84] Roth, B., and Freudenstein, F., 1963, "Synthesis of path-generating mechanisms by numerical methods," *ASME J. Eng. Industry*, 85B:298306.

[85] Roth, B., 1975, "Performance Evaluation of Manipulators from a Kinematic Viewpoint", *NBS Special Publications*, No.459, pp. 39-61.

[86] Sandor, G.N., 1968, "Principles of a General Quaternion-Operator Method of Spatial Kinematic Synthesis", *Journal of Applied Mechanics*, **35**(1):40-46.

[87] Sandor, G.N., and Bisshopp, K.E., 1969, "On a General Method of Spatial Kinematic Synthesis by Means of a Stretch-Rotation Tensor", *Journal of Engineering for Industry*, **91**:115-122.

[88] Sandor, G. N., and Erdman, A. G., 1984, *Advanced Mechanism Design: Analysis and Synthesis, Vol. 2.* Prentice-Hall, Englewood Cliffs, NJ.

[89] Sandor, G.N., Xu, Y., and Weng, T.C., 1986, "Synthesis of 7-R Spatial Motion Generators with Prescribed Crank Rotations and Elimination of Branching", *The International Journal of Robotics Research*, **5**(2):143-156.

[90] Sandor, G.N., Weng, T.C., and Xu, Y., 1988, "The Synthesis of Spatial Motion Generators with Prismatic, Revolute and Cylindric Pairs without Branching Defect", *Mechanism and Machine Theory*, **23**(4):269-274.

[91] Selig, J.M., 1996, *Geometrical Methods in Robotics*. Springer-Verlag, New York.

[92] Shoham, M., and Jen, F.H., 1994, "On rotations and translations with application to robot manipulators", *Advanced Robotics*, **8**(2):203-229.

[93] Su, H., Collins, C., and McCarthy, J. M., "An Extensible Java Applet for Spatial Linkage Synthesis," *Proc. ASME Des. Eng. Technical Conferences*, paper no. DETC2002/MECH-24271, Montreal, Canada, 2002.

[94] Suh, C.H., 1968, "Design of Space Mechanisms for Rigid-Body Guidance", *ASME J. of Engineering for industry*, **90**B:499-506.

[95] Suh, C.H., 1969, "On the Duality in the Existence of R-R Links for Three Positions." *J. Eng. Ind. Trans. ASME*, **91**(B):129-134.

[96] Suh, C.H. and Radcliffe, C.W., 1978, *Kinematics and Mechanisms Design*, John Wiley & Sons, New York.

[97] Tsai, L. W., 1972, *Design of Open Loop Chains for Rigid Body Guidance*, Ph.D. Thesis, Department of Mechanical Engineering, Stanford University.

[98] Tsai, L. W., and Roth, B., 1972, "Design of Dyads with Helical, Cylindrical, Spherical, Revolute and Prismatic Joints," *Mechanism and Machine Theory*, 7:591-598.

[99] Tsai, L. W., and Roth, B., "A Note on the Design of Revolute-Revolute Cranks," *Mechanism and Machine Theory,* Vol. 8, pp. 23-31, 1973.

[100] Tsai, Y.C., and Soni, A.H., 1981, "Accessible Region and Synthesis of Robot Arm", *ASME Journal of Mechanical Design*, **103**:803-811.

[101] Tsai, L. W., 1999, *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*, John Wiley and Sons, New York, NY.

[102] Vance, J. M., Larochelle, P., and Dorozhkin, D., 2002, "VRSpatial: Designing Spatial Mechanisms Using Virtual Reality," *Proc. Des. Eng. Tech. Conf.* paper no. DETC2002/MECH-34377, Sept. 29-Oct. 2, Montreal, Canada.

[103] Veldkamp, G.R., 1967, "Canonical Systems and Instantaneous Invariants in Spatial Kinematics." *Journal of Mechanisms*, **3**: 329-388.

[104] Verschelde, J., 1999, "Algorithm 795: PHCpack: A generalpurpose solver for polynomial systems by homotopy continuation," *ACM Transactions on Mathematical Software*, 25(2): 251276, 1999. Software available at http://www.math.uic.edu/jan.

[105] Vijaykumar, R., Waldron, K., and Tsai, M. J., 1987, "Geometric Optimization of Manipulator Structures for Working Volume and Dexterity", *Kinematics of Robot Manipulators (J. M. McCarthy, ed.)*, pp. 99-111, MIT Press, Cambridge, MA.

[106] Walker, M., 1988, "Manipulator kinematics and the epsilon algebra", *IEEE J. Robot. Automat.*, **4**:186-192

[107] Yang, A.T., and Freudenstein, F., 1964, "Application of Dual-Number Quaternion Algebra to the Analysis of Spatial Mechanisms", *ASME Journal of Applied Mechanics*, June 1964, pp.300-308.

[108] Yu, H.C., 1981, "The Bennett Linkage, its Associated Tetrahedron and the Hyperboloid of its Axes". *Mechanism and Machine Theory*, **16**:105-114.

Appendix A: Numerical Results

1. Solutions for the RR robot

Table A-1: The solutions for the RR chain

G	$(0.14, 0.94, 0.30), (-1.3, -0.33, 1.7)$
W	$(0.59, 0.032, 0.81), (-0.81, -2.6, 0.69)$
G	$(-0.54 - 0.02i, 1.03 + 0.21i, 0.34 - 0.66i), (-2.8 + 0.5i, -1.0 + 1.5i, 1.1 - 1.0i)$
W	$(0.54 - 0.56i, 0.72 - 0.04i, 0.82 + 0.41i), (-0.9 - 1.5i, -1.2 - 1.3i, 2.8 + 0.1i)$
G	$(-0.54 + 0.02i, 1.03 - 0.21i, 0.34 + 0.66i), (-2.8 - 0.5i, -1.0 - 1.5i, 1.1 + 1.0i)$
W	$(0.54 + 0.56i, 0.72 + 0.04i, 0.82 - 0.41i), (-0.9 + 1.5i, -1.2 + 1.3i, 2.8 - 0.1i)$
G	$(-0.88, 0.45, 0.17), (-1.6, -2.8, -0.73)$
W	$(0.14, 0.94, 0.30), (-1.3, -0.33, 1.7)$
G	$(1.6 - 0.04i, 0.76 + 0.44i, 0.20 - 1.38i), (-3.9 + 1.2i, 1.7 + 1.3i, 3.2 + 3.4i)$
W	$(0.69 + 1.17i, -1.4 + 0.1i, 0.79 - 0.86i), (-1.1 - 1.5i, 0.4 + 2.4i, -3.7 + 3.2i)$
G	$(1.6 + 0.04i, 0.76 - 0.44i, 0.20 + 1.38i), (-3.9 - 1.2i, 1.7 - 1.3i, 3.2 - 3.4i)$
W	$(0.69 - 1.17i, -1.4 - 0.1i, 0.79 + 0.86i), (-1.1 + 1.5i, 0.4 - 2.4i, -3.7 - 3.2i)$

2. Solutions for the RRP robot

Table A-2: The solutions for the RRP chain

$\mathbf{h}$	$\mathbf{g}$	$\mathbf{g}^0$	$\mathbf{w}$	$\mathbf{w}^0$
$(-0.091, 0.131, 0.987)$	$(0, -0.896, 0.443)$	$(1.95, -3.40, -6.87)$	$(0.297, -0.829, -0.474)$	$(-2.59, 2.48, -5.96)$
$(-0.533, 0.451, 0.716)$	$(0, -0.896, 0.443)$	$(0.600, -0.979, -1.98)$	$(0.297, -0.829, -0.474)$	$(1.05, 1.42, -1.83)$
$(-0.350, 0.724, 0.594)$	$(0, -0.896, 0.443)$	$(-1.60, -0.596, -1.21)$	$(0.297, -0.829, -0.474)$	$(0.253, 0.272, -0.317)$
$(-0.821, 0.548, 0.160)$	$(0, -0.896, 0.443)$	$(6.45, 0.334, 0.676)$	$(0.297, -0.829, -0.474)$	$(3.51, 0.913, 0.605)$
$(-0.350, 0.191, 0.917)$	$(0, 0.494, 0.869)$	$(1.08, -3.83, 2.17)$	$(0.143, -0.488, 0.861)$	$(1.70, -3.29, -2.14)$
$(-0.495, 0.576, 0.650)$	$(0, 0.494, 0.869)$	$(-1.19, -1.50, 0.851)$	$(0.143, -0.488, 0.861)$	$(-0.725, -1.59, -0.783)$
$(0.919, 0.333, 0.210)$	$(0, 0.494, 0.869)$	$(-0.810, 0.133, -0.076)$	$(0.143, -0.488, 0.861)$	$(1.17, 6.79, 3.65)$
$(-0.032, 0.999, 0.023)$	$(0, 0.494, 0.869)$	$(-19.4, -16.9, 9.58)$	$(0.143, -0.488, 0.861)$	$(-34.5, 8.14, 10.3)$
$(-0.179, -0.026, 0.983)$	$(0, 0.895, 0.446)$	$(1.08, -2.97, 5.96)$	$(0.639, 0.544, 0.544)$	$(0.173, -1.33, 1.13)$
$(-0.485, 0.532, 0.694)$	$(0, 0.895, 0.446)$	$(-0.753, -0.922, 1.85)$	$(0.639, 0.544, 0.544)$	$(-0.596, -1.05, 1.76)$
$(-0.790, 0.435, 0.433)$	$(0, 0.895, 0.446)$	$(-3.23, -0.304, 0.610)$	$(0.639, 0.544, 0.544)$	$(-2.30, 2.82, -0.116)$
$(-0.134, 0.853, 0.504)$	$(0, 0.895, 0.446)$	$(0.532, -0.122, 0.244)$	$(0.639, 0.544, 0.544)$	$(-0.934, -0.548, 1.65)$

3. Solutions for the TP robot

Table A-3: The solutions for the TP chain

Joint Axis	*Direction*	*Moment*
G_1	$(0, -0.67, 0.74)$	$(-0.06, -0.90, -0.82)$
G_2	$(-0.17, 0.73, 0.66)$	$(-4.19, -1.33, 0.38)$
H	$(0, -0.49, 0.87)$	$(-1.34, 0.66, 0.37)$
G_1	$(0, -0.67, 0.74)$	$(5.01, -1.11, -1.01)$
G_2	$(-0.17, 0.73, 0.66)$	$(-6.83, -2.45, 0.92)$
H	$(0, 0.66, 0.75)$	$(-6.36, 0.36, -0.31)$
G_1	$(0, 0.35, 0.93)$	$(-3.82, -1.13, 0.43)$
G_2	$(-0.99, -0.07, 0.02)$	$(0.14, -3.13, -2.99)$
H	$(0, -0.49, 0.87)$	$(-2.29, -1.18, -0.67)$
G_1	$(0, 0.35, 0.93)$	$(-3.90, -1.40, 0.53)$
G_2	$(-0.99, -0.07, 0.02)$	$(0.55, -8.52, -1.06)$
H	$(0, 0.66, 0.75)$	$(-8.19, -1.23, 1.08)$
G_1	$(0, 0.99, 0.17)$	$(-3.55, -0.20, 1.196)$
G_2	$(-0.52, -0.14, 0.84)$	$(-2.01, -2.64, -1.70)$
H	$(0, 0.66, 0.75)$	$(-0.65, -2.54, -1.43)$
G_1	$(0, 0.99, 0.17)$	$(-8.54, -0.25, 1.48)$
G_2	$(-0.52, -0.14, 0.84)$	$(0.41, -5.72, -0.71)$
H	$(0, 0.66, 0.75)$	$(-7.13, -2.41, 2.11)$

4. Solutions for the RPP robot

Table A-4: The solutions for the RPP chain

Axis	*Direction*	*Moment*
G	$(0.98, -0.14, 0.16)$	$(0.25, 1.45, -0.29)$
$\mathbf{n}$	$(-0.26, 0.93, 0.25)$	
G	$(0.98, -0.14, 0.16)$	$10^{16}(2.3 - 0.5i, 4.9 - 9.7i, -9.6 - 5.2i)$
$\mathbf{n}$	$(-0.9 + 0.1i, 0.3 + 0.2i, 0.0 - 0.1i)$	
G	$(0.98, -0.14, 0.16)$	$10^{16}(2.3 + 0.5i, 4.9 + 9.7i, -9.6 + 5.2i)$
$\mathbf{n}$	$(-0.9 - 0.1i, 0.3 - 0.2i, 0.0 + 0.1i)$	
G	$(0.98, -0.14, 0.16)$	$10^{15}(2.0 + 0.4i, 7.7 - 5.99i, -5.8 - 7.8i)$
$\mathbf{n}$	$(0.9 - 0.1i, -0.7 + 0.2i, 0.3 + 0.5i)$	
G	$(0.98, -0.14, 0.16)$	$10^{15}(2.0 - 0.4i, 7.7 + 6.0i, -5.8 + 7.8i)$
$\mathbf{n}$	$(0.9 + 0.1i, -0.7 - 0.2i, 0.3 - 0.5i)$	
G	$(0.98, -0.14, 0.16)$	$10^{16}(-1.0, 0.05, 6.3)$
$\mathbf{n}$	$(0.98, -0.14, 0.16)$	

5. Solutions for the RRR robot

Table A-5: The real solutions for the RRR chain

$\mathbf{g}$	$\mathbf{g}^0$	$\mathbf{w}$	$\mathbf{w}^0$	$\mathbf{f}$	$\mathbf{f}^0$
$(-0.98, -0.013, 0.19)$	$(-0.34, 2.21, -1.57)$	$(-0.3, 0.94, -0.15)$	$(3.81, 0.99, -1.38)$	$(0.33, 0.65, -0.68)$	$(1.53, -1.45, -0.66)$
$(-0.84, 0.35, 0.42)$	$(-1.14, 2.52, -4.33)$	$(0.56, -0.40, 0.725)$	$(-3.75, -2.94, 1.26)$	$(-0.91, -0.32, -0.27)$	$(-1.25, 2.94, 0.76)$
$(-0.75, 0.65, -0.11)$	$(2.33, 2.90, 1.16)$	$(0.385, 0.92, 0.01)$	$(5.52, -2.31, 0.40)$	$(0.69, -0.375, 0.62)$	$(-2.63, -2.91, 1.15)$
$(-0.70, -0.65, -0.305)$	$(0.09, 0.27, -0.78)$	$(-0.18, -0.78, -0.6)$	$(0.12, -2.00, 2.55)$	$(0.245, -0.49, 0.84)$	$(-0.68, 0.975, 0.765)$
$(-0.51, 0.05, -0.86)$	$(2.98, 1.51, -1.70)$	$(-0.17, -0.04, 0.98)$	$(-1.23, 2.26, -0.11)$	$(0.945, -0.099, 0.31)$	$(-0.97, -1.36, 2.51)$
$(-0.48, -0.74, -0.48)$	$(-4.17, 0.085, 4.04)$	$(0.61, -0.565, -0.56)$	$(-0.46, -2.66, 2.19)$	$(0.55, 0.81, -0.18)$	$(3.93, -3.28, -2.77)$
$(-0.45, 0.265, -0.85)$	$(4.58, -0.52, -2.60)$	$(-0.23, -0.75, -0.62)$	$(-0.68, 0.17, 0.04)$	$(-0.75, 0.415, -0.52)$	$(1.18, 0.17, -1.56)$
$(-0.42, -0.83, 0.37)$	$(-4.83, 2.37, -0.22)$	$(-0.68, 0.01, -0.73)$	$(2.03, -4.27, -1.93)$	$(-0.49, 0.35, -0.80)$	$(3.45, 1.97, -1.24)$
$(-0.37, -0.91, -0.21)$	$(-0.48, 0.38, -0.80)$	$(-0.39, 0.43, -0.81)$	$(2.97, 1.39, -0.69)$	$(0.77, 0.56, -0.296)$	$(1.45, -1.68, 0.60)$
$(-0.25, 0.49, 0.835)$	$(0.59, 2.13, -1.07)$	$(0.88, -0.05, -0.47)$	$(1.64, -0.42, 3.09)$	$(-0.78, -0.42, -0.46)$	$(-1.40, 2.93, -0.33)$
$(-0.23, -0.93, 0.28)$	$(-5.10, 1.28, 0.12)$	$(-0.42, 0.45, -0.79)$	$(3.83, -4.26, -4.46)$	$(-0.80, 0.396, -0.44)$	$(2.63, 3.66, -1.50)$
$(-0.20, 0.95, -0.24)$	$(3.63, 1.23, 1.79)$	$(0.84, -0.50, 0.20)$	$(-0.43, -1.48, -1.89)$	$(-0.47, -0.16, 0.87)$	$(-3.95, -0.61, -2.24)$
$(-0.20, -0.46, -0.87)$	$(-0.11, 0.72, -0.35)$	$(0.45, -0.18, 0.87)$	$(-2.37, 0.45, 1.32)$	$(0.95, 0.21, -0.24)$	$(1.32, -2.87, 2.69)$
$(-0.16, 0.38, -0.91)$	$(6.94, -4.46, -3.07)$	$(0.81, -0.57, 0.13)$	$(-0.01, 0.24, 1.17)$	$(-0.715, -0.68, 0.16)$	$(0.58, -0.63, -0.096)$
$(-0.11, -0.37, -0.92)$	$(-0.21, -0.34, 0.16)$	$(0.95, -0.26, -0.18)$	$(-0.00, -2.45, 3.56)$	$(0.43, 0.097, -0.90)$	$(3.40, -2.09, 1.41)$
$(-0.10, -0.37, -0.92)$	$(-0.18, -0.44, 0.196)$	$(-0.52, -0.59, 0.62)$	$(-3.64, 1.94, -1.20)$	$(0.47, -0.88, 0.115)$	$(-2.41, -0.96, 2.53)$
$(-0.10, 0.21, 0.97)$	$(-0.65, 1.43, -0.37)$	$(0.79, 0.42, -0.44)$	$(2.88, -1.90, 3.38)$	$(-0.55, 0.39, -0.74)$	$(2.32, 1.67, -0.85)$
$(-0.06, 0.41, -0.91)$	$(6.52, -4.47, -2.45)$	$(-0.96, 0.22, -0.19)$	$(0.10, -0.37, -0.94)$	$(0.40, 0.91, 0.07)$	$(-1.13, 0.52, -0.29)$
$(0.96, -0.08, 0.27)$	$(0.15, -3.74, -1.61)$	$(0.68, -0.72, 0.10)$	$(-1.76, -1.09, 3.99)$	$(-0.38, -0.72, -0.59)$	$(-2.02, 0.55, 0.64)$

6. Solutions for the RPRP robot

Joint Axis	*Direction*
g	$(-1.06 + 0.09i, -0.40 + 0.31i, 0.38 + 0.58i)$
w	$(-0.29 - 0.08i, 0.76 + 0.96i, 1.27 - 0.59i)$
g	$(-1.06 - 0.09i, -0.40 - 0.31i, 0.38 - 0.58i)$
w	$(-0.29 + 0.08i, 0.76 - 0.96i, 1.27 + 0.59i)$
g	$(-0.48, -0.78, 0.39)$
w	$(0.33, -0.23, 0.91)$
g	$(0.04, 0.05, 0.99)$
w	$(0.70, 0.48, 0.53)$
g	$(0.47 - 0.40i, -1.27 - 0.18i, 0.06 - 0.80i)$
w	$(1.06 - 0.05i, 0.15 - 0.30i, 0.27 + 0.36i)$
g	$(0.47 + 0.40i, -1.27 + 0.18i, 0.06 + 0.80i)$
w	$(1.06 + 0.049i, 0.15 + 0.30i, 0.27 - 0.36i)$

Table A-6: Solutions for the directons of the revolute joints of the RPRP robot

$\mathbf{g}^0$	$\mathbf{h}$	$\mathbf{w}^0$	$\mathbf{u}$
First real solution			
$(-5.17, 5.49, 4.64)$	$(0.48, 0.36, 0.83)$	$(0.73, -8.39, -2.38)$	$(0.66, -0.53, 0.53)$
$(-3.69, 1.71, -1.15)$	$(0.30, -0.22, 0.93)$	$(-0.50, 0.01, 0.18)$	$(0.36, -0.21, 0.91)$
$(0.11, -0.60, -1.07)$	$(-0.06, -0.26, 0.96)$	$(-2.61, -1.82, 0.50)$	$(0.63, -0.13, 0.76)$
$(2.85, 0.415, 4.39)$	$(0.46, 0.21, 0.86)$	$(-0.18, 0.51, 0.19)$	$(0.39, -0.78, 0.49)$
$(-0.75, 1.65, 2.40)$	$(-0.56, 0.61, 0.56)$	$(5.72, 7.44, -0.23)$	$(-0.39, 0.50, 0.77)$
$(-8.27, 9.19, 8.25)$	$(0.88, 0.25, 0.40)$	$(3.43, 32.6, 6.94)$	$(0.70, -0.66, 0.29)$
$(-1.09, -0.04, -1.43)$	$(0.26, -0.22, 0.94)$	$(-2.03, -0.51, 0.62)$	$(0.39, -0.21, 0.90)$
Second real solution			
$(3.35, 0.25, -0.15)$	$(-0.71, -0.55, 0.44)$	$(0.57, -0.47, -0.33)$	$(-0.11, -0.98, 0.17)$
$(1.43, -0.09, -0.06)$	$(-0.33, -0.45, 0.83)$	$(1.85, -1.40, -1.16)$	$(0.30, 0.84, 0.44)$
$(-70.4, 92.8, -1.31)$	$(0.98, 0.16, 0.12)$	$(-31.0, -92.1, 122.)$	$(0.96, -0.09, 0.25)$
$(2.81, 4.81, -0.34)$	$(0.05, -0.98, 0.18)$	$(6.86, -5.55, -3.98)$	$(0.67, -0.73, 0.13)$
$(-2.00, 1.68, 0.01)$	$(-0.44, 0.24, 0.86)$	$(9.94, -23.9, 8.37)$	$(-0.36, 0.22, 0.90)$

Table A-7: The real solutions for the RPRP robot

7. Solutions for the RPC robot

Joint Axis	*Direction*	*Moment*
G	$(-0.42, -0.48, 0.77)$	$(3.52, -0.36, 1.73)$
h	$(0.29, -0.87, -0.39)$	
W	$(-0.92, -0.36, 0.12)$	$(0.25, -0.05, 1.78)$
G	$(0.20, -0.43, 0.88)$	$(0.07, 1.18, 0.55)$
h	$(0.44, -0.76, -0.47)$	
W	$(-0.37, -0.63, 0.68)$	$(0.10, 0.32, 0.36)$
G	$(-0.61 - 0.23i, -0.14 - 0.01i, 0.83 - 0.17i)$	$(0.16 + 2.22i, -1.54 - 2.93i, -0.93 + 0.96i)$
h	$(-0.55 - 0.21i, 0.94 - 0.17i, -0.11 - 0.35i)$	
W	$(-0.17 - 0.71i, 0.24 + 0.021i, 1.19 - 0.11i)$	$(0.69 - 0.81i, -1.45 - 5.57i, 0.64 + 1.49i)$
G	$(-0.61 + 0.23i, -0.14 + 0.01i, 0.83 + 0.17i)$	$(0.16 - 2.22i, -1.54 + 2.93i, -0.93 - 0.96i)$
h	$(-0.55 + 0.21i, 0.94 + 0.17i, -0.11 + 0.35i)$	
W	$(-0.17 + 0.71i, 0.24 - 0.021i, 1.19 + 0.11i)$	$(0.69 + 0.81i, -1.45 + 5.57i, 0.64 - 1.49i)$
G	$(1.55 - 1.10i, -1.62 - 1.57i, 0.83 - 1.01i)$	$(25.09 + 11.74i, 11.63 - 23.31i, 7.64 - 2.65i)$
h	$(0.35 - 0.82i, -0.84 + 0.20i, -1.04 - 0.44i)$	
W	$(-0.12 - 0.97i, -1.51 + 0.02i, 0.14 - 0.61i)$	$(6.49 - 1.83i, -2.16 - 5.07i, 2.09 - 1.81i)$
G	$(1.55 + 1.10i, -1.62 + 1.57i, 0.83 + 1.01i)$	$(25.09 - 11.74i, 11.63 + 23.31i, 7.64 + 2.65i)$
h	$(0.35 + 0.82i, -0.84 - 0.20i, -1.04 + 0.44i)$	
W	$(-0.12 + 0.97i, -1.51 - 0.02i, 0.14 + 0.61i)$	$(6.49 + 1.83i, -2.16 + 5.07i, 2.09 + 1.81i)$

Table A-8: The six solutions for the RPC robot

8. Solutions for the PRRR robot

Table A-9: The ten shorter solutions for the PRRR chain

h	G	W	F
$(0.441, -0.142, 0.89)$	$(-0.60, -0.04, 0.796; 1.62, -0.29, 1.21)$	$(0.55, -0.78, -0.28; 0.23, 1.38, -3.42)$	$(-0.94, -0.34, -0.06; 0.23, -0.99, 2.01)$
$(0.158, 0.986, 0.04)$	$(-0.74, 0.596, -0.31; -0.498, -0.18, 0.84)$	$(0.02, -0.07, -0.997; -1.29, 0.62, -0.07)$	$(-0.53, 0.77, -0.345; 0.367, 0.94, 1.55)$
$(-0.34, -0.64, -0.69)$	$(0.57, -0.47, -0.675; -1.01, 0.15, -0.95)$	$(0.418, -0.838, 0.35; 2.03, -0.332, -3.21)$	$(-0.16, 0.12, 0.98; 2.27, -3.64, 0.818)$
$(0.24, 0.96, 0.16)$	$(0.40, -0.74, 0.54; 1.32, -0.156, -1.19)$	$(0.57, 0.77, 0.29; 1.70, -1.20, -0.13)$	$(-0.22, 0.099, -0.97; -1.87, 1.30, 0.55)$
$(0.816, 0.58, 0.01)$	$(-0.795, 0.28, 0.54; -0.235, 0.50, -0.60)$	$(0.62, -0.32, 0.71; 0.239, 1.03, 0.255)$	$(0.42, -0.71, -0.56; 1.25, 1.60, -1.10)$
$(-0.25, -0.97, -0.01)$	$(0.77, -0.63, 0.10; -0.36, -0.76, -2.01)$	$(0.36, 0.47, -0.81; -0.765, -0.36, -0.55)$	$(0.54, -0.71, -0.45; 1.09, 0.997, -0.27)$
$(0.087, 0.81, -0.578)$	$(0.825, -0.565, 0.01; 0.797, 1.15, -0.62)$	$(0.44, 0.80, -0.40; 0.82, -0.349, 0.21)$	$(-0.385, 0.51, 0.768; -1.87, -3.21, 1.20)$
$(0.25, 0.91, -0.34)$	$(0.85, 0.30, -0.42; 0.12, -0.78, -0.32)$	$(-0.08, 0.74, 0.66; -1.00, -0.59, 0.54)$	$(-0.85, 0.29, -0.44; -0.60, -0.29, 0.96)$
$(-0.696, -0.0, -0.72)$	$(0.148, 0.345, -0.93; -2.05, 0.556, -0.12)$	$(-0.068, 0.93, 0.36; 0.62, -1.03, 2.75)$	$(-0.95, -0.264, 0.175; 0.39, -0.54, 1.28)$
$(-0.61, 0.44, -0.65)$	$(0.36, 0.90, -0.225; -0.70, 0.32, 0.17)$	$(0.43, -0.096, 0.89; 1.44, -1.23, -0.83)$	$(0.23, -0.175, -0.96; -2.46, 3.45, -1.22)$

9. Solutions for the CRR robot

Table A-10: Five of the solutions for the CRR robot

Length	*Joint*	*Line*
1.66	G	$(-0.45, -0.14, -0.88) + \epsilon(-1.91, 0.59, 0.89)$
	W	$(0.25, -0.89, 0.37) + \epsilon(1.40, -0.87, -3.03)$
	F	$(-0.88, -0.23, -0.42) + \epsilon(-0.96, 0.69, 1.64)$
1.75	G	$(0.14, 0.98, 0.11) + \epsilon(0.14, -0.02, -0.01)$
	W	$(-0.61, -0.60, 0.51) + \epsilon(0.79, -0.20, 0.72)$
	F	$(0.44, -0.89, 0.12) + \epsilon(-0.59, -0.50, -1.55)$
2.59	G	$(-0.73, -0.07, -0.68) + \epsilon(-1.50, 0.32, 1.60)$
	W	$(0.29, -0.82, -0.50) + \epsilon(-1.14, 1.54, -3.22)$
	F	$(-0.49, 0.399, -0.78) + \epsilon(-1.88, 2.14, 2.26)$
2.65	G	$(0.39, 0.90, -0.18) + \epsilon(-0.55, 0.025, -1.08)$
	W	$(-0.42, 0.54, -0.73) + \epsilon(-1.45, 0.40, 1.14)$
	F	$(0.74, 0.13, -0.66) + \epsilon(-0.66, 0.55, -0.63)$
2.68	G	$(0.31, 0.95, 0.02) + \epsilon(0.51, -0.16, -0.26)$
	W	$(0.71, 0.60, -0.37) + \epsilon(0.01, -0.40, -0.62)$
	F	$(0.62, -0.61, -0.49) + \epsilon(-2.15, -0.35, -2.29)$

Table A-11: The other five solutions for the CRR robot

Length	*Joint*	*Line*
2.99	G	$(0.31, 0.95, 0.02) + \epsilon(0.51, -0.16, -0.26)$
	W	$(0.71, 0.60, -0.37) + \epsilon(0.01, -0.40, -0.62)$
	F	$(0.62, -0.61, -0.49) + \epsilon(-2.15, -0.35, -2.29)$
3.00	G	$(-0.12, -0.98, -0.17) + \epsilon(-1.32, 0.25, -0.46)$
	W	$(0.88, 0.28, 0.37) + \epsilon(1.60, -2.84, -1.63)$
	F	$(0.26, 0.487, -0.83) + \epsilon(-1.45, 0.67, -0.06)$
3.06	G	$(0.18, -0.94, -0.27) + \epsilon(-0.76, 0.32, -1.64)$
	W	$(0.98, -0.15, -0.09) + \epsilon(-0.39, -1.93, -0.89)$
	F	$(-0.20, -0.55, 0.81) + \epsilon(0.51, -0.76, -0.39)$
3.08	G	$(0.05, -0.69, -0.71) + \epsilon(-2.22, 0.25, -0.40)$
	W	$(-0.35, -0.25, -0.90) + \epsilon(-2.75, 2.09, 0.47)$
	F	$(0.90, 0.36, -0.25) + \epsilon(-0.07, -0.50, -0.99)$
3.20	G	$(-0.146, -0.77, -0.62) + \epsilon(-2.67, 0.17, 0.41)$
	W	$(-0.90, 0.41, -0.15) + \epsilon(0.89, 2.10, 0.42)$
	F	$(-0.68, -0.09, 0.72) + \epsilon(2.09, 0.59, 2.04)$

10. Real solutions for the TPR robot

Table A-12: Three real solutions for the TPR robot

Joint Axis	*Direction*	*Moment*
$\mathbf{g}_1$	$(-0.72, 0.13, -0.68)$	$(-3.07, 2.02, -2.58) \times \mathbf{g}_1$
$\mathbf{g}_2$	$(-0.54, 0.51, 0.67)$	$(-3.07, 2.02, -2.58) \times \mathbf{g}_2$
$\mathbf{h}$	$(0.45, 0.89, -0.11)$	
W	$(0.29, -0.88, -0.36)$	$(-2.66, -1.61, 1.79)$
$\mathbf{g}_1$	$(-0.60, 0.79, -0.15)$	$(0.10, 3.6, 1.7) \times \mathbf{g}_1$
$\mathbf{g}_2$	$(0.37, 0.44, 0.82)$	$(0.10, 3.6, 1.7) \times \mathbf{g}_2$
$\mathbf{h}$	$(-0.63, 0.32, -0.71)$	
W	$(0.58, -0.57, -0.59)$	$(-1.3, -3.0, 1.6)$
$\mathbf{g}_1$	$(-0.27, 0.84, 0.47)$	$(-2.7, 0.25, -0.60) \times \mathbf{g}_1$
$\mathbf{g}_2$	$(0.76, -0.11, 0.64)$	$(-2.7, 0.25, -0.60) \times \mathbf{g}_2$
$\mathbf{h}$	$(-0.47, -0.66, 0.58)$	
W	$(-0.18, 0.034, 0.98)$	$(-0.55, 2.7, -0.19)$

Table A-13: Three more real solutions for the TPR robot

Joint Axis	*Direction*	*Moment*
$\mathbf{g}_1$	$(-0.13, -0.71, 0.69)$	$(-0.86, 1.9, 0.67) \times \mathbf{g}_1$
$\mathbf{g}_2$	$(-0.90, -0.21, -0.38)$	$(-0.86, 1.9, 0.67) \times \mathbf{g}_2$
$\mathbf{h}$	$(-0.49, 0.76, -0.44)$	
W	$(-0.062, 0.84, 0.54)$	$(0.87, 0.86, -1.2)$
$\mathbf{g}_1$	$(0.42, 0.28, 0.86)$	$(-1.0, 0.13, -4.1) \times \mathbf{g}_1$
$\mathbf{g}_2$	$(0.91, -0.096, -0.41)$	$(-1.0, 0.13, -4.1) \times \mathbf{g}_2$
$\mathbf{h}$	$(-0.15, 0.18, 0.97)$	
W	$(0.74, -0.58, 0.34)$	$(-0.74, 0.19, 2.0)$
$\mathbf{g}_1$	$(0.75, -0.069, -0.66)$	$(0.078, 4.7, 2.5 \times \mathbf{g}_1)$
$\mathbf{g}_2$	$(0.54, -0.51, 0.67)$	$(0.078, 4.7, 2.5) \times \mathbf{g}_2$
$\mathbf{h}$	$(0.16, 0.79, -0.60)$	
W	$(-0.73, 0.67, 0.11)$	$(-0.63, -1.2, 2.7)$

11. Solutions for the RRRR robot

Table A-14: The solutions for the RRRR chain

G	W	F	U
$(-0.74, -0.21, 0.64; -1.32, 3.03, -0.53)$	$(-0.5, -0.41, -0.76; 1.30, -2.24, 0.37)$	$(-0.44, 0.87, -0.21; -4.39, -2.56, -1.39)$	$(0.79, -0.44, 0.43; -0.07, 2.92, 3.11)$
$(-0.69, -0.36, 0.63; -1.83, -0.33, -2.16)$	$(0.52, -0.44, 0.74; -6.72, -3.37, 2.73)$	$(0.33, -0.11, 0.94; -3.27, 4.10, 1.64)$	$(-0.30, 0.59, 0.75; 2.31, -1.59, 2.19)$
$(-0.83, -0.06, 0.56; -1.21, 0.66, -1.71)$	$(0.33, 0.59, 0.74; -4.92, 2.90, -0.12)$	$(-0.25, -0.01, -0.97; 0.17, -3.25, -0.018)$	$(-0.99, -0.05, -0.02; 0.11, -3.52, 2.83)$
$(0.66, 0.08, 0.74; 1.94, -1.80, -1.54)$	$(0.04, 0.97, -0.22; 1.18, -0.08, -0.17)$	$(0.6, -0.36, 0.71; -4.75, -5.74, 1.06)$	$(0.86, 0.31, 0.40; 1.34, -3.73, -0.04)$
$(0.58, -0.81, 0.01; 1.11, 0.85, 4.61)$	$(-0.72, -0.16, -0.67; 1.69, -1.16, -1.52)$	$(0.28, -0.51, -0.81; -0.13, -3.09, 1.92)$	$(-0.19, 0.87, -0.45; -2.04, -0.50, -0.12)$
$(-0.99, -0.05, 0.0; 0.21, -4.11, 0.06)$	$(-0.22, -0.94, 0.26; -0.48, 0.11, -0.01)$	$(-0.87, -0.47, 0.16; -0.13, 1.85, 4.66)$	$(0.19, 0.42, -0.88; 1.29, -3.65, -1.47)$
$(-0.85, -0.05, 0.52; 1.46, -0.07, 2.41)$	$(-0.06, 0.62, -0.78; -3.18, -6.37, -4.78)$	$(0.82, 0.57, 0.04; 0.61, -0.64, -3.41)$	$(-0.46, -0.39, 0.8; 1.53, 0.91, 1.31)$
$(-0.62, -0.67, 0.40; 2.78, -0.53, 3.45)$	$(-0.86, 0.48, -0.16; -0.98, -2.34, -1.67)$	$(0.0, -0.99, 0.11; 0.68, 0.09, 0.80)$	$(0.85, 0.49, -0.19; -0.62, -0.04, -2.95)$
$(-0.52, -0.53, 0.66; 0.23, 1.96, 1.75)$	$(-0.24, 0.97, -0.01; 0.49, 0.11, -0.68)$	$(-0.62, 0.38, -0.69; -0.65, -3.65, -1.42)$	$(-0.12, -0.98, -0.15; 2.24, -0.4, 0.84)$
$(0.08, -0.98, -0.18; 3.31, -0.72, 5.27)$	$(0.94, -0.28, -0.19; 0.48, 0.58, 1.54)$	$(0.88, -0.35, 0.33; 1.54, 1.91, -2.06)$	$(-0.54, 0.08, 0.84; 1.77, -0.80, 1.21)$
$(0.95, 0.27, 0.17; -0.37, 0.74, 0.90)$	$(-0.40, 0.57, -0.72; -2.10, -1.99, -0.42)$	$(0.13, -0.86, 0.5; -0.14, -1.04, -1.77)$	$(0.08, 0.98, 0.20; -0.31, 0.23, -1.0)$

12. Solutions for the PPS robot

Sol.	**n**	**c**
1	$(-0.377, 0.918, 0.123)$	$(1.59, 2.51, 0.0385)$
2	$(0.212, 0.846, 0.489)$	$(7.89, -1.96, -0.596)$
3	$(-0.565, 0.157, 0.810)$	$(-11.2, -9.66, -7.10)$
4	$(0.573, 0.809, 0.131)$	$(2.51, -2.40, 0.675)$
5	$(0.75 + 0.35i, -0.45 + 0.32i, 0.70 - 0.17i)$	$(-2.39 + 1.06i, -0.75 + 0.16i, 2.61 + 3.13i)$
6	$(0.75 - 0.35i, -0.45 - 0.32i, 0.70 + 0.17i)$	$(-2.39 - 1.06i, -0.75 - 0.16i, 2.61 - 3.13i)$
7	$(0.07 - 0.90i, -1.39 + 0.15i, 0.47 + 0.58i)$	$(-0.41 - 1.06i, 1.281 + 0.17i, 0.30 - 1.28i)$
8	$(0.07 + 0.90i, -1.39 - 0.15i, 0.47 - 0.58i)$	$(-0.41 + 1.06i, 1.28 - 0.17i, 0.30 + 1.28i)$
9	$(-0.22, 0.048, 0.975)$	$(-7.33, -3.21, -2.16)$
10	$(-0.893, -0.234, 0.384)$	$(-0.729, -1.22, -1.57)$

Table A-15: Ten real solutions for the PPS robot

Appendix B: The Cylindroid in the Tetrahedron

1. Bennett's Linkage

Consider separately the input and output cranks of Bennett's linkage which we denote as GW and HU, respectively (Figure B-1). For the input crank, let N_1 be the common normal between the axes G and W of the revolute joints, and let $\mathbf{B}$ and $\mathbf{P}$ be their points of intersection with N_1. The angle α and distance a from G to W measured around and along N_1 define the dimensions of the input crank, and can be assembled into the dual number $\hat{\alpha} = (\alpha, a)$. The output crank HU of Bennett's linkage has the same dimensions, that is $\hat{\alpha} = (\alpha, a)$ defines the angle and distance from H to U around and along its common normal N_2. Let $\mathbf{Q}$ and $\mathbf{C}$ be the intersections of N_2 with H and U, respectively.

In order to form the linkage, we assemble the base link so that the points $\mathbf{B}$ and $\mathbf{Q}$ lie on the common normal between the fixed joint axes G and H, and $\mathbf{P}$ and $\mathbf{C}$ lie on the common normal to the two moving axes W and U. The dimensions of these two links are to be the same, such that $\hat{\gamma} = (\gamma, g)$ defines the angle and distance between these axes around the common normal.

It is possible to show that in order to achieve this assembly, the dimensions of

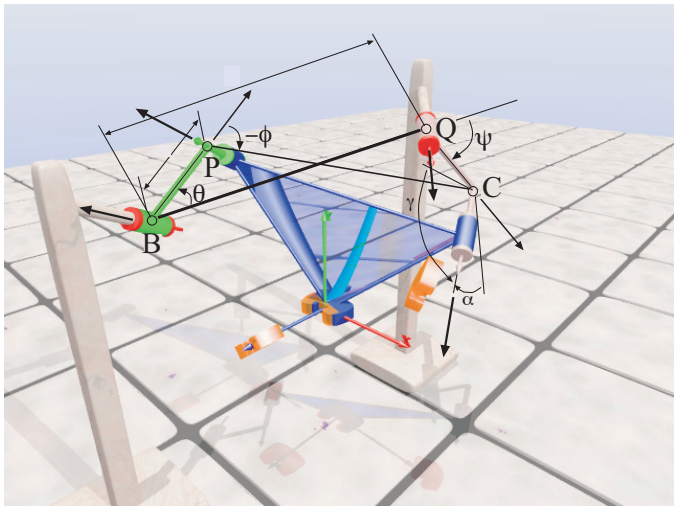

Figure B-1: Bennett's linkage defined by axes GWUH and their common normals.

Bennett's linkage must satisfy the relation (Hao[29]),

$$\frac{\sin\alpha}{a} = \frac{\sin\gamma}{g}. \tag{B-1}$$

For the analysis of the Bennett's linkage, we introduce a frame F with origin at the point **B**, its z-axis along the fixed axis G, and its x-axis along the common normal between G and H.

The input-output equation for Bennett's linkage [54] is derived from the fact that that the points **P** and **C** must remain separated by the distance g as the linkage moves. This constraint equation,

$$(\mathbf{C}-\mathbf{P})\cdot(\mathbf{C}-\mathbf{P}) = g^2 \tag{B-2}$$

yields

$$\tan\frac{\psi}{2} = k^{\pm}\tan\frac{\theta}{2}, \tag{B-3}$$

where

$$k^{+} = \frac{\sin\frac{\alpha+\gamma}{2}}{\sin\frac{\alpha-\gamma}{2}}, \quad \text{and} \quad k^{-} = \frac{\cos\frac{\alpha-\gamma}{2}}{\cos\frac{\alpha+\gamma}{2}}. \tag{B-4}$$

Hunt [37] shows that k^+ is the correct root for the conventions that we have chosen for the angles α and γ. The negative root yields the constant associated with a different sign convention. In what follows we focus on the positive root, which we denote at k, and drop the + superscript.

To relate the output angle ψ to the coupler angle ϕ (the angle measured around W that defines the position of the coupler WU relative to the input crank), we write the loop equations for the linkage. We equate the two expressions of the point $\mathbf{C}$ to obtain

$$\begin{aligned} g(\cos\psi - \cos\phi) &= 0, \\ g\sin\alpha(\sin\psi - \sin\phi) &= 0. \end{aligned} \tag{B-5}$$

Thus, we see that the coupler angle ϕ is equal to the output angle ψ. This allows us to use (B-3) to compute

$$\tan\frac{\phi}{2} = k\tan\frac{\theta}{2}, \tag{B-6}$$

where, as we have seen previously, k depends only on the linkage dimensions.

2. The Screw Surface

In this section we compute the expression of the screw axes belonging to the cylindroid on a certain reference frame. The location of the cylindroid depends on the reference configuration; to make this dependence explicit, we need to express the rotation angles as the difference from the reference angle. We denote $\Delta\theta = \theta - \theta_0$, $\Delta\phi = \phi - \phi_0$, being θ_0, ϕ_0 the angles that define the reference configuration.

Expanding the dual quaternion product according to this notation and the geometry of the Bennett's linkage, we obtain

$$\cos\frac{\Delta\theta}{2}\cos\frac{\Delta\phi}{2} = \cos\frac{\Delta\psi}{2} - \frac{\frac{\Delta t}{2}\,\sin\frac{\Delta\psi}{2}}{a\,\sin\alpha} \tag{B-7}$$

for the scalar, and dividing this result into the dual vector part, we obtain

$$\mathsf{F} = M(\Delta\psi)\left(1, \frac{\frac{\Delta t}{2}}{\tan\frac{\Delta\psi}{2}}\right)\mathsf{S} = \tan\frac{\Delta\theta}{2}\,\mathsf{G} + \tan\frac{\Delta\phi}{2}\,\mathsf{W} + \tan\frac{\Delta\theta}{2}\tan\frac{\Delta\phi}{2}\,\mathsf{G}\times\mathsf{W}, \tag{B-8}$$

where

$$M(\Delta\psi) = \frac{a\,\sin\alpha\sin\frac{\Delta\psi}{2}}{a\,\sin\alpha\cos\frac{\Delta\psi}{2} - \frac{\Delta t}{2}\sin\frac{\Delta\psi}{2}} \tag{B-9}$$

is the *magnitude* of each screw and S is the Plücker coordinates of its axis. The parameter

$$P = \frac{\frac{\Delta t}{2}}{\tan\frac{\Delta\psi}{2}}, \tag{B-10}$$

is the pitch of the screw.

The axes of the screws F are the axes of relative displacement screws of the coupler of Bennett's linkage, and form the *screw surface* for the movement.

The expression of the relative displacement screws can be computed using Eq.(B-8), to be

$$\mathsf{F}(\theta,\theta_0) = \tan\frac{\theta-\theta_0}{2}\,\mathsf{G} + \tan\frac{\phi-\phi_0}{2}\,\mathsf{W} + \tan\frac{\theta-\theta_0}{2}\tan\frac{\phi-\phi_0}{2}\,\mathsf{G}\times\mathsf{W}. \tag{B-11}$$

We expand this equation using the standard reference frame to define G and W, and the relationship between θ and ϕ given by the Bennett linkage condition. Factor the common terms from the components of F to obtain the simplified expression F',

$$\mathsf{F}'(\theta,\theta_0) = \left(\begin{Bmatrix} -(\mathrm{c}\alpha - \mathrm{c}\gamma)\,\mathrm{s}\alpha\,\mathrm{s}(\frac{\theta+\theta_0}{2}) \\ (\mathrm{c}\alpha - \mathrm{c}\gamma)\,\mathrm{s}\alpha\,\mathrm{c}(\frac{\theta+\theta_0}{2}) \\ \mathrm{s}\alpha\,(\mathrm{c}(\frac{\theta-\theta_0}{2})\,\mathrm{s}\alpha - \mathrm{c}(\frac{\theta+\theta_0}{2})\,\mathrm{s}\gamma) \end{Bmatrix}, \begin{Bmatrix} -a(\mathrm{c}\alpha - \mathrm{c}\gamma)\,\mathrm{c}\alpha\,\mathrm{s}(\frac{\theta+\theta_0}{2}) \\ a(\mathrm{c}\alpha - \mathrm{c}\gamma)\,\mathrm{c}\alpha\,\mathrm{c}(\frac{\theta+\theta_0}{2}) \\ a(\mathrm{c}\alpha - \mathrm{c}\gamma)\,\mathrm{s}\alpha\,\mathrm{c}(\frac{\theta-\theta_0}{2}) \end{Bmatrix}\right), \tag{B-12}$$

where "c" and "s" denote the cosine and sine functions.

Huang [34] has shown that this screw surface is a *cylindroid*. See Fig. 6.2 for several views of a cylindroid.

A change in the reference position,

$$[D_{1\theta}] = [D_{0\theta}][D_{10}], \tag{B-13}$$

where $[D_{10}] = [D(\theta_0, \phi_0)][D(\theta_1, \phi_1)]^{-1}$, is a right translation of the constraint manifold. It is important to note that this is not a rigid transformation of the screw surface. In what follows, we show how the cylindroid is affected by a change in reference position.

3. Principal Axes of the Cylindroid

Our computation of the principal axes follow the methodology of Hunt [37]. We generate the cylindroid from two arbitrary screws U and V ,

$$\mathsf{U} = M_1(1, P_1)\mathsf{S}_1, \quad \mathsf{V} = M_2(1, P_2)\mathsf{S}_2. \tag{B-14}$$

The line K, which is the common normal to S_1 and S_2. Denote $\mathsf{S}_1 = \mathsf{I}$ and compute $\mathsf{J} = \mathsf{K} \times \mathsf{I}$. Then, we can define S_2 as

$$\mathsf{S}_2 = \cos\hat{\delta}\mathsf{I} + \sin\hat{\delta}\mathsf{J}, \tag{B-15}$$

where $\hat{\delta} = (\delta, d)$ is the dual angle from S_1 to S_2 measured around and along K.

We can now write the linear combination of U and V as

$$\begin{aligned} \mathsf{F} &= a(1, P_1)\,\mathsf{I} + b(1, P_2)(\cos\hat{\delta}\,\mathsf{I} + \sin\hat{\delta}\,\mathsf{J}), \\ &= \left(a(1, P_1) + b(1, P_2)\cos\hat{\delta}\right)\mathsf{I} + b(1, P_2)\sin\hat{\delta}\,\mathsf{J} \end{aligned} \tag{B-16}$$

where we have absorbed the magnitudes M_1 and M_2 into the constants a and b, respectively. This equation shows that the axes of all the screws F intersect K in a right angle. K is nodal line of the cylindroid.

Using the nodal line, the axes of screws of the cylindroid can be located by measuring the dual angle $\hat{\zeta} = (\zeta, z)$ relative to $\mathsf{I} = \mathsf{S}_1$ around the nodal line K, that

is

$$\mathsf{F} = M(1,P)(\cos\hat{\zeta}\,\mathsf{I} + \sin\hat{\zeta}\,\mathsf{J}). \tag{B-17}$$

The parameter z is called the *offset* and locates the intersection of the axis of each screw of the cylindroid with the nodal line K.

We can equate (B-16) and (B-17) and eliminate the constants a and b in order to derive a formula for the offset z and pitch P of each screw in the cylindroid,

$$\begin{Bmatrix} z(\zeta) \\ P(\zeta) \end{Bmatrix} = \begin{bmatrix} -\sin\zeta & \cos\zeta \\ \cos\zeta & \sin\zeta \end{bmatrix} \begin{bmatrix} P_1 & (P_2 - P_1)\cot\delta - d \\ 0 & P_2 + d\cot\delta \end{bmatrix} \begin{Bmatrix} \cos\zeta \\ \sin\zeta \end{Bmatrix}. \tag{B-18}$$

The screws with the maximum and minimum values of the pitch P are the principal axes of the cylindroid. Set the derivative of the above expression to zero, to obtain

$$\frac{d}{d\zeta}\begin{Bmatrix} z \\ P \end{Bmatrix} = \begin{bmatrix} -(P_2 - P_1)\cot\delta + d & (P_2 - P_1) + d\cot\delta \\ (P_2 - P_1) + d\cot\delta & (P_2 - P_1)\cot\delta - d \end{bmatrix} \begin{Bmatrix} \sin 2\zeta \\ \cos 2\zeta \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}. \tag{B-19}$$

The angles $\zeta = \sigma$ with extreme values of P are given by

$$\tan 2\sigma = \frac{-(P_2 - P_1)\cot\delta + d}{(P_2 - P_1) + d\cot\delta}, \tag{B-20}$$

which yields two angles separated by $\pi/2$. This defines the directions of principal axes X and Y of the cylindroid. The offset s to the principal axes is given by substituting σ into equation (B-18) to obtain

$$s = \frac{1}{2}\left(d - (P_2 - P_1)\frac{\cos\delta}{\sin\delta}\right). \tag{B-21}$$

Thus, the principal axes are located by the dual angle $\hat{\sigma} = (\sigma, s)$, as

$$\mathsf{X} = \cos\hat{\sigma}\mathsf{I} + \sin\hat{\sigma}\mathsf{J}, \quad \mathsf{Y} = -\sin\hat{\sigma}\mathsf{I} + \cos\hat{\sigma}\mathsf{J}. \tag{B-22}$$

The height of the cylindroid is the distance along K between minimum and maximum offsets. This is calculated from from equation (B-19) to be

$$L_c = \frac{\sqrt{(P_2 - P_1)^2 + d^2}}{\sin\delta}. \tag{B-23}$$

If the principal screws are used to define the cylindroid, then $\hat{\delta} = (\delta, d) = (\pi/2, 0)$, and the height is given by (Hunt [37]),

$$L_c = |P_Y - P_X|. \tag{B-24}$$

4. The Bennett Tetrahedron

The reference configuration of the Bennett linkage defines locations of the points **B**,**P**,**C**, and **Q** that form the tetrahedron.

We introduce the line L_{CB} that forms the edge of the tetrahedron defined by the segment **CB**. Similarly, let L_{PQ} be the line defined by **PQ**. The dual vectors for these lines are calculated to be

$$\mathsf{L}_{CB} = (\mathbf{B} - \mathbf{C}, \mathbf{B} \times (\mathbf{B} - \mathbf{C})) \quad \text{and} \quad \mathsf{L}_{PQ} = (\mathbf{Q} - \mathbf{P}, \mathbf{Q} \times (\mathbf{Q} - \mathbf{P}))\,. \tag{B-25}$$

Using the expressions of the points **B**, **Q**, **P** and **C** in the reference frame F, we can compute

$$\begin{aligned}
\mathsf{L}_{CB} &= \left(\begin{Bmatrix} \sin\alpha\cos\gamma\cos\theta_0 - \cos\alpha\sin\gamma \\ \sin\alpha\cos\gamma\sin\theta_0 \\ \sin\alpha\sin\gamma\sin\theta_0 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \right) \text{ and} \\
\mathsf{L}_{PQ} &= \left(\begin{Bmatrix} \sin\gamma - \sin\alpha\cos\theta_0 \\ -\sin\alpha\sin\theta_0 \\ 0 \end{Bmatrix}, \begin{Bmatrix} 0 \\ 0 \\ -g\sin\alpha\sin\theta_0 \end{Bmatrix} \right).
\end{aligned} \tag{B-26}$$

We now show that the this tetrahedron is symmetric relative to the line joining the midpoints of the edges **CB** and **PQ**. In particular, this line is the common normal

to the two lines L_{CB} and L_{PQ}. Define the midpoints $\mathbf{V}_1$ and $\mathbf{V}_2$, such that

$$\mathbf{V}_1 = \frac{\mathbf{B}+\mathbf{C}}{2} = \begin{Bmatrix} \frac{1}{2}(g + a\cos\psi_0) \\ \frac{1}{2}(a\cos\gamma\sin\psi_0) \\ \frac{1}{2}(a\sin\gamma\sin\psi_0) \\ 1 \end{Bmatrix} \quad \text{and} \quad \mathbf{V}_2 = \frac{\mathbf{P}+\mathbf{Q}}{2} = \begin{Bmatrix} \frac{1}{2}(g + a\cos\theta_0) \\ \frac{1}{2}(a\sin\theta_0) \\ 0 \\ 1 \end{Bmatrix}. \tag{B-27}$$

The vector $\mathbf{T} = \mathbf{V}_2 - \mathbf{V}_1$ is perpendicular to the lines L_{CB} and L_{PQ}, if it is perpendicular to the the direction components, $\mathbf{L}_1$ and $\mathbf{L}_2$, respectively. Computing the dot product $\mathbf{T}\cdot\mathbf{L}_1$, we find that

$$\begin{aligned}(\mathbf{V}_2 - \mathbf{V}_1)\cdot\mathbf{L}_1 &= (a/2)\left((g - a\cos\theta)\cos\psi - (a\cos\gamma\sin\theta)\sin\psi + (a + g\cos\theta)\right) \\ &= 0. \end{aligned} \tag{B-28}$$

A similar calculation yields $\mathbf{T}\cdot\mathbf{L}_2 = 0$. The line along $\mathbf{T}$ is the *axis* of the Bennett tetrahedron,

The dual cross product $\mathsf{T} = \mathsf{L}_{CB} \times \mathsf{L}_{PQ}$ defines a screw along the direction $\mathbf{T}$,

$$\mathsf{T} = \left(\begin{Bmatrix} \sin\alpha\sin\theta_0 \\ \sin\gamma - \sin\alpha\cos\theta_0 \\ \cos\alpha - \cos\gamma \end{Bmatrix}, \begin{Bmatrix} -g\cot\gamma\sin\alpha\sin\theta_0 \\ g(\cot\gamma\sin\alpha\cos\theta_0 - \cos\alpha) \\ 0 \end{Bmatrix} \right). \tag{B-29}$$

Compute the dual dot product

$$\mathsf{F}'(\theta,\theta_0)\cdot\mathsf{T} = (\mathbf{F},\mathbf{F}^\circ)\cdot(\mathbf{T},\mathbf{T}^\circ) = (\mathbf{F}\cdot\mathbf{T}, \mathbf{F}^\circ\cdot\mathbf{T} + \mathbf{F}\cdot\mathbf{T}^\circ) = (0,0), \tag{B-30}$$

which shows that the axis of T is the common normal to each of the axes of the screws F'.

This proves that the axis of the Bennett tetrahedron is the nodal line of the cylindroid of relative screw axes. The proof of the location of the tetrahedron in the middle point of the cylindroid for the general case is a tedious calculation that we did with Mathematica and it is not included here. We include below the special case of the folded configuration, in which the expressions become simpler.

5. The Folded Configuration

The screw T, which defines the nodal line of the cylindroid, takes a simple form if the initial configuration of the Bennett linkage is $\theta_0 = 0$. In this configuration the the linkage *folds* to form a straight line. See Figure B-2.

Introduce the angles $\xi = (\alpha + \gamma)/2$ and $\eta = (\alpha - \gamma)/2$, substitute $\alpha = \xi + \eta$ and $\gamma = \xi - \eta$ into (B-29), and drop the common factor to obtain

$$\mathsf{T} = \left(\begin{Bmatrix} 0 \\ \cos\xi \\ \sin\xi \end{Bmatrix}, \begin{Bmatrix} 0 \\ g\frac{\cos\eta}{\sin(\eta-\xi)} \\ 0 \end{Bmatrix} \right). \tag{B-31}$$

It is easy to see that the axis of T is directed along a line that bisects the axes of G and U parallel to the yz plane. We determine a point $\mathbf{r}$ on the axis of $\mathsf{T} = (\mathbf{T}, \mathbf{T}^\circ)$ using the formula

$$\mathbf{r} = \frac{\mathbf{T} \cdot \mathbf{T}^\circ}{\mathbf{T} \cdot \mathbf{T}} = \begin{Bmatrix} \frac{a+g}{2} \\ 0 \\ 0 \end{Bmatrix}. \tag{B-32}$$

In this calculation, we have used the fact that $a = g\sin\alpha/\sin\gamma$. Thus, the axis of T intersects the x-axis midway along the length of the folded linkage.

We now introduce the coordinate frame M with its y-axis aligned the x-axis of F, its origin at $\mathbf{r}$ and its z-axis along T. Coordinates in the frame F are transformed to M by the 6×6 matrix,

$$[\hat{T}] = \begin{bmatrix} 0 & \sin\xi & -\cos\xi & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \cos\xi & \sin\xi & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sin\xi & -\cos\xi \\ 0 & 0 & \frac{g+a}{2} & -1 & 0 & 0 \\ 0 & -\frac{g+a}{2} & 0 & 0 & \sin\xi & -\cos\xi \end{bmatrix}. \tag{B-33}$$

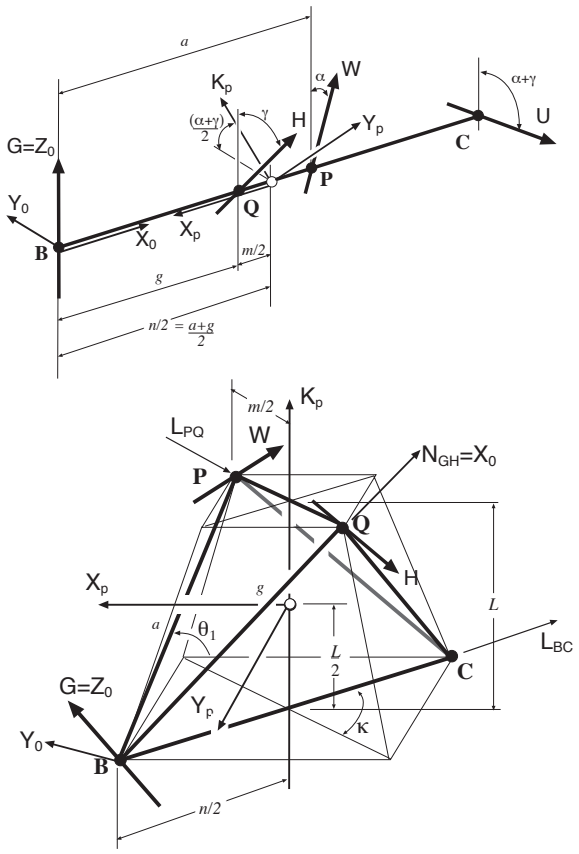

Figure B-2: The Bennett linkage with initial angle $\theta = 0$ and $\theta = \theta_1$

Applying this transformation to the screws F' in the cylindroid (B-12), we obtain

$$\mathsf{F}' = \left(\begin{Bmatrix} \cos\frac{\theta}{2}\frac{1}{\sin\xi} \\ \sin\frac{\theta}{2} \\ 0 \end{Bmatrix}, \begin{Bmatrix} g\cos\frac{\theta}{2}\frac{\sin\eta}{\sin(\eta-\xi)} \\ -g\sin\frac{\theta}{2}\frac{\cos(\xi+\eta)}{\sin(\eta-\xi)} \\ 0 \end{Bmatrix} \right). \tag{B-34}$$

The zero values in the z components of the two vectors defining F' show that the z-axis of M is the nodal line of this cylindroid.

In order to compute the principal screws, we choose two arbitrary screws of this cylindroid and apply the formulas of the earlier section. For this purpose the screws

$\mathsf{U} = \mathsf{F}'(0)$ and $\mathsf{V} = \mathsf{F}'(\pi)$ are convenient choices. We have

$$\mathsf{U} = \left(\left\{ \begin{array}{c} \frac{1}{\sin\xi} \\ 0 \\ 0 \end{array} \right\}, \left\{ \begin{array}{c} g\frac{\sin\eta}{\sin(\eta-\xi)} \\ 0 \\ 0 \end{array} \right\} \right), \quad \mathsf{V} = \left(\left\{ \begin{array}{c} 0 \\ 1 \\ 0 \end{array} \right\}, \left\{ \begin{array}{c} 0 \\ -g\frac{\cos(\xi+\eta)}{\sin(\eta-\xi)} \\ 0 \end{array} \right\} \right). \tag{B-35}$$

The axes of these screws, denoted X and Y, are easily seen to be the x and y axes of M. Therefore, the angle between them is $\delta = \pi/2$ and the normal distance is $d = 0$. Equations (B-20) and (B-21) yield $\sigma = 0$ and $s = 0$ which means that X and Y are, in fact, the principal axes of the cylindroid F'.

We can now calculate the pitch $P(\theta)$ of each of the screws in the cylindroid to be

$$P(\theta) = \left(\frac{g\sin\xi}{\sin(\eta-\xi)} \right) \left(\frac{\cos(\frac{\theta}{2})^2 \sin\eta - \sin(\frac{\theta}{2})^2 \sin\xi\cos(\eta+\xi)}{\cos(\frac{\theta}{2})^2 + \sin(\frac{\theta}{2})^2 \sin\xi^2} \right). \tag{B-36}$$

The z coordinate of their intersection with the axis of T is given by

$$z(\theta) = \frac{-g\sin\xi\cos\xi\sin\theta}{2\sin(\eta-\xi)(\cos(\frac{\theta}{2})^2 + \sin(\frac{\theta}{2})^2 \sin\xi^2)}. \tag{B-37}$$

This provides a complete characterization of the cylindroid associated with Bennett's linkage when displacements of the coupler are measured relative to its folded configuration.

6. The Coordinate Transformation

The axis of the Bennett tetrahedron is known to be the line of symmetry for the movement of the Bennett linkage, See Yu[108] and Baker [5]. We denote the axis of the Bennett tetrahedron by K, as it coincides with the nodal line of the cylindroid. We now construct a rectangular pyramid with K as its axis, that contains this tetrahedron. This pyramid provides a convenient way to find the principal axes of the cylindroid. See Figure B-3.

Consider the rectangle in a plane perpendicular to K that has vertices **B** and **C** on its diagonal. Similarly, consider the second rectangle perpendicular to K, that

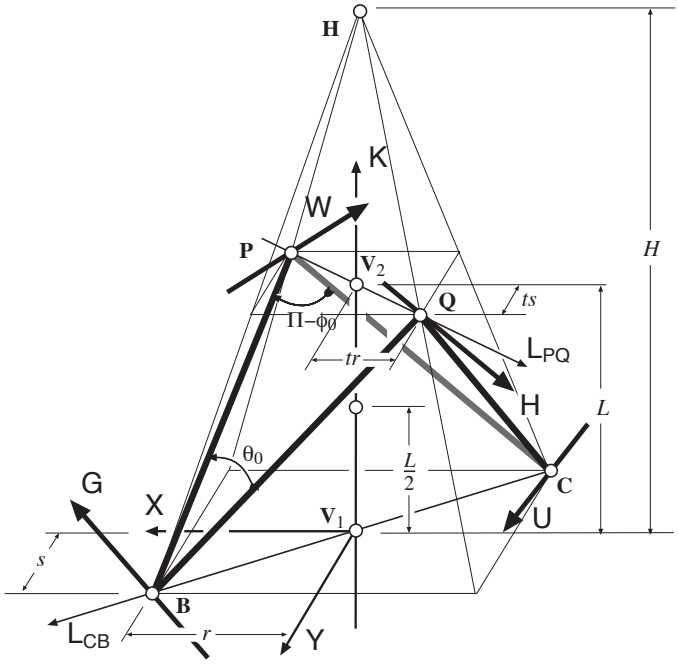

Figure B-3: The pyramid associated with the Bennett Linkage GWUH.

has vertices **P** and **Q** on its diagonal. These two rectangles are cross sections of a rectangular pyramid. The four vertices of these two cross-sections are labeled $\mathbf{A}_i$ and $\mathbf{B}_i$, $i = 1, 2, 3, 4$ counter-clockwise around the axis K. The vertices

$$\mathbf{A}_1 = \mathbf{B}, \quad \mathbf{B}_2 = \mathbf{Q}, \quad \mathbf{A}_3 = \mathbf{C}, \quad \mathbf{B}_4 = \mathbf{P}, \tag{B-38}$$

define the Bennett tetrahedron.

We now assume that the length $m = |\mathbf{Q} - \mathbf{P}|$ is less than the length $n = |\mathbf{C} - \mathbf{B}|$, which means the cross-section $\mathbf{A}_i$ can be viewed as the base of the reference pyramid. Introduce the coordinate frame M' with its z-axis along K, and its x-axis directed along $\mathbf{A}_1 - \mathbf{A}_2$, which means its y-axis is directed along $\mathbf{A}_1 - \mathbf{A}_4$. This allows us to define the coordinates of $\mathbf{A}_i$ as

$$\mathbf{A}_1 = \begin{Bmatrix} r \\ s \\ 0 \\ 1 \end{Bmatrix}, \mathbf{A}_2 = \begin{Bmatrix} -r \\ s \\ 0 \\ 1 \end{Bmatrix}, \mathbf{A}_3 = \begin{Bmatrix} -r \\ -s \\ 0 \\ 1 \end{Bmatrix}, \mathbf{A}_4 = \begin{Bmatrix} r \\ -s \\ 0 \\ 1 \end{Bmatrix}. \tag{B-39}$$

The coordinates of the points $\mathbf{A}_i$ and $\mathbf{B}_i$ are related by the fact that they lie parallel cross-sections of the pyramid. Let $\mathbf{H} = (0, 0, H, 1)^T$ define the apex of the pyramid measured from the first cross-section along K, then we have

$$\mathbf{B}_i = (1-t)\mathbf{A}_i + t\mathbf{H}, \quad i = 1, \ldots, 4. \tag{B-40}$$

The parameter t ranges between 0 and 1. The value of t that locates the second cross-section is determined by computing the ratio of the lengths $m = |\mathbf{Q} - \mathbf{P}|$ and $n = |\mathbf{C} - \mathbf{B}|$, that is

$$t = 1 - \frac{m}{n}. \tag{B-41}$$

The lengths m and n are found using the cosine law, that is

$$m = \sqrt{a^2 + g^2 - 2ag\cos\theta_0}, \quad n = \sqrt{a^2 + g^2 + 2ag\cos\phi_0}. \tag{B-42}$$

The distance L between the two cross-sections and the height H of the pyramid are related by the formula

$$L = tH. \tag{B-43}$$

The value of L is obtained by evaluating the magnitude of the vector joining the midpoints $\mathbf{V}_1$ and $\mathbf{V}_2$, which, after some algebra, is given by

$$L = |\mathbf{V}_2 - \mathbf{V}_1| = \sqrt{\frac{1}{2}ag(\cos\theta_0 - \cos\phi_0)}. \tag{B-44}$$

In order to complete the specification of this pyramid, we determine the lengths r and s that locate the points $\mathbf{A}_i$. This is done using the identities $a^2 = (\mathbf{B}_4 - \mathbf{A}_1) \cdot (\mathbf{B}_4 - \mathbf{A}_1)$ and $g^2 = (\mathbf{B}_2 - \mathbf{A}_1) \cdot (\mathbf{B}_2 - \mathbf{A}_1)$, which yield the matrix equation

$$\begin{bmatrix} t^2 & (t-2)^2 \\ (t-2)^2 & t^2 \end{bmatrix} \begin{Bmatrix} r^2 \\ s^2 \end{Bmatrix} = \begin{Bmatrix} a^2 - (tH)^2 \\ g^2 - (tH)^2 \end{Bmatrix}. \tag{B-45}$$

This is easily solved to determine the values for r and s.

The parameters r, s, t and L completely define the reference pyramid. They are obtained from the reference configuration of the Bennett linkage as defined by the specified input angle $\theta = \theta_0$. See Figure B-2.

The coordinate transformation $[D] = [A, \mathbf{d}]$ from the standard frame F to the frame M' located on the base of the pyramid in Figure B-3 can be obtained from the matrix expression

$$\begin{bmatrix} [A] & \mathbf{d} \\ 0\,0\,0 & 1 \end{bmatrix} \begin{bmatrix} 0 & g & g + a\cos\phi_0 & a\cos\theta_0 \\ 0 & 0 & a\cos\gamma\sin\phi_0 & a\sin\theta_0 \\ 0 & 0 & a\sin\gamma\sin\phi_0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} r & -r(1-t) & -r & r(1-t) \\ s & s(1-t) & -s & -s(1-t) \\ 0 & tH & 0 & tH \\ 1 & 1 & 1 & 1 \end{bmatrix}. \tag{B-46}$$

Given the coordinates of these points in both frames, we can multiply on the right by the inverse of the matrix defined by points in F to compute the transformation $[D]$.