Parallel Robots with Configurable Platforms

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen
op dinsdag 21 mei 2013 om 12.30 uur

door

Patrice LAMBERT

Maître ès Sciences (M. Sc)
Université Laval, Canada
geboren te Shawinigan-Sud, Canada.

Dit proefschrift is goedgekeurd door de promotoren:
Prof.ir. R. H. Munnig Schmidt
Prof. dr.ir. J.L. Herder

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof.ir. R. H. Munnig Schmidt,	Technische Universiteit Delft, promotor
Prof.dr.ir. J. L. Herder,	Technische Universiteit Delft, promotor
Prof.dr.ir. J.S. Dai,	King's College London, Verenigd Koninkrijk
Prof.dr. C. W. Scherer,	Universität Stuttgart, Duitsland
Prof.dr.ir. S. Stramigioli,	Universiteit Twente
Prof.dr.ir. M. Mulder,	Technische Universiteit Delft
Prof.dr. F.C.T van der Helm,	Technische Universiteit Delft
Prof.dr.ir. F. van Keulen,	Technische Universiteit Delft, reservelid

This research have been carried out with the support of the MicroNed program.

Copyright ©2013 by Patrice Lambert

All right reserved. No part of the material protected by this copyright notice may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without the prior permission from the author.

ISBN 978-94-91104-14-5

Author email: p.lambert@tudelft.nl; patrice-lambert@hotmail.com

Contents

1 Introduction 3

1.1 Parallel Robots with Configurable Platforms . . . 3
  1.1.1 Parallel Robots - Closing the loops . . . 3
  1.1.2 Configurable Platforms . . . 4
  1.1.3 Some Definitions . . . 4
1.2 State of the Art . . . 6
  1.2.1 Literature . . . 6
  1.2.2 Industry . . . 7
1.3 Objectives and Contributions of this thesis . . . 8
1.4 Structure of this thesis . . . 10

I Parallel Mechanisms with Configurable Platforms 13

2 Mobility and Overconstraints Analysis of Parallel Mechanisms 15

2.1 Introduction . . . 15
2.2 Standard Method for the Mobility Analysis of Overconstrained Parallel Mechanisms . . . 17
  2.2.1 Modified Chebychev–Grübler–Kutzbach Criterion . . . 17
  2.2.2 Basic Results From Screw Theory . . . 19
    2.2.2.1 The Screw . . . 19
    2.2.2.2 Screw Systems . . . 20
  2.2.3 Mobility of Overconstrained Parallel Mechanisms . . . 21
  2.2.4 An Example . . . 23
  2.2.5 Analytical VS Numerical Method for the Calculation of the Mobility 26
2.3 Solutions to Screw Systems under Linear Homogeneous Equations . . . 26
  2.3.1 Restricted Screw System . . . 28

2.3.2 Unrestricted Screw Systems and Distribution 30
2.3.3 Example 1: Distribution of Local Mobilities in a 6R Planar Chain. 31
2.3.4 Example 2: Distribution of the overconstraints in a 4-RRR Parallel Mechanism 32
2.4 Discussion and Summary 35

3 Mobility and Overconstraints Analysis of Parallel Mechanisms with Configurable Platforms 37

3.1 Introduction 37
3.2 Mobility of Series-Parallel Mechanisms 38
3.2.1 Introduction 38
3.2.2 Series-Parallel Graphs 39
3.2.3 Hybrid Mechanisms 40
3.2.4 Delta Mechanisms 41
3.2.5 General Series-Parallel Mechanisms 43
3.2.6 Cross-Jointing Mechanisms 45
3.2.7 Non Series-Parallel Mechanisms 46
3.3 Mobility Analysis of Parallel Mechanisms with Configurable Platforms . . 46
3.3.1 Graph Reduction 48
3.3.2 Legs Matrix of Mobilities 49
3.3.3 Connection of the Platform to the Legs 50
3.4 Overconstraints Analysis of Parallel Mechanisms with Configurable Platforms 52
3.4.1 Self Dual Topogoly of Parallel Mechanisms with Configurable Platforms 52
3.4.2 Platform Matrix of Overconstraints 53
3.4.3 Connection of the legs to the Platform 56
3.5 Summary of the General Method 56
3.6 Examples 59
3.6.1 Example 1: a 4-RRR Parallel mechanisms with a planar 8R configurable platform 59
3.6.1.1 Graph Reduction and Edge Screws Systems 59
3.6.1.2 Mobility Analysis 60
3.6.1.3 Overconstraints Analysis 63
3.6.2 Example 2: PentaG Mechanism 66
3.6.2.1 Graph Reduction and Screw Systems 66
3.6.2.2 Mobility Analysis 72
3.6.2.3 Overconstraints Analysis 74
3.7 Summary 76

II Parallel Robots with Configurable Platforms 79

4 Kinematics of the PentaG Robot 81
- 4.1 Position Kinematics . . . 82
  - 4.1.1 Inverse Position Kinematics . . . 82
    - 4.1.1.1 Inverse Position Kinematic of the Configurable Platform 83
    - 4.1.1.2 Leg Inverse Kinematics . . . 86
    - 4.1.1.3 Working Modes . . . 87
  - 4.1.2 Direct Position Kinematics . . . 88
    - 4.1.2.1 Analytical Methods . . . 88
    - 4.1.2.2 Numerical Methods . . . 89
    - 4.1.2.3 Assembly Modes . . . 89
- 4.2 Velocity Kinematics and Static Analysis . . . 90
  - 4.2.1 The Jacobian Matrix . . . 90
    - 4.2.1.1 The Jacobian Matrix of the leg . . . 91
    - 4.2.1.2 The Jacobian matrix of the configurable platform . . . 92
    - 4.2.1.3 The PentaG Jacobian Matrix . . . 95
  - 4.2.2 Accuracy and Stiffness . . . 96
- 4.3 Static Balancing . . . 97
  - 4.3.1 Concept . . . 98
  - 4.3.2 Application to the PentaG . . . 100
  - 4.3.3 Simulation . . . 102
  - 4.3.4 Discussion . . . 103
- 4.4 Singular Configurations . . . 105
  - 4.4.1 Serial Singularities of the PentaG . . . 107
  - 4.4.2 Parallel Singularities of the PentaG . . . 109
- 4.5 Summary . . . 112

5 Optimal Kinematic Design of the PentaG 113
- 5.1 Introduction . . . 113
- 5.2 Kinematic Parameters . . . 114
  - 5.2.1 On the Number of Kinematic Parameters . . . 115
  - 5.2.2 Geometric Conditions for the Feasibility of the PentaG . . . 118
  - 5.2.3 Reduction of the Number of Parameters . . . 118
  - 5.2.4 Upper and Lower Limits of Parameters . . . 120

5.3 Objective Function 122
5.3.1 Maximum Useful Workspace 123
5.3.2 Computation of the Singularity-Free Workspace 124
5.4 Optimisation Method 125
5.4.1 Genetic Algorithms 126
5.5 Results 127
5.6 Summary 131
5.7 Discussion 131

6 Prototype Design and Implementation 135
6.1 Introduction 135
6.2 Inertia-Stiffness Optimisation 136
6.2.1 Rigid Bodies Model and Finite Element Model 137
6.2.2 Inertia participation 138
6.2.2.1 Tip forces as function of velocity and acceleration 138
6.2.2.2 Velocity and Acceleration Cases 138
6.2.3 Compliance Contribution 139
6.2.4 Optimisation 140
6.2.5 Results 141
6.3 Technical Design 143
6.3.1 Specifications 143
6.3.1.1 Specification Transfer 144
6.3.2 Actuators 147
6.3.3 Sensors 147
6.3.4 Bearings 149
6.3.5 Mechanical Parts Design 149
6.3.5.1 Lower limb 149
6.3.5.2 Parallelogram units 150
6.3.5.3 Configurable Platform 150
6.3.5.4 Comparison between the desired and the final geometry . 152
6.3.6 Hardware Control 153
6.3.7 Software Control 153
6.3.8 Properties of the actuator-sensor systems 154
6.4 Implementation and Testing 154
6.5 Conclusion 159

7 Conclusions **161**
7.1 Contribution of this thesis 161
7.2 Some Remarks 163

Summary **171**

Samenvatting **175**

Acknowledgements **179**

Curriculum Vitae **181**

Chapter 1

Introduction

1.1 Parallel Robots with Configurable Platforms

1.1.1 Parallel Robots - Closing the loops

Robots are playing an increasingly prominent role in today's society and already affect a large spectrum of our live from robots built to manufacture goods and products in factories to robots built to interact directly with humans in a professional or in an entertainment context. Factors that can explain the rapid developments of robotics over the last two decades are due to the computer revolution that allows smaller and more efficient control hardware to operate in real-time and the miniaturization of accurate sensors and powerful motors that can be implemented in a robotic system. However, no matter how much efforts are put into the development of better control systems, algorithms, motors and sensors, which will not be discussed here, robot performance is ultimately limited by its mechanical structure, which presents inertia, friction, compliance and defines the workspace and the inputs – outputs relations between the control system and the interaction with the external world.

Some researchers have therefore considered the possibility to develop robots with different kinematic structures. Even nowadays, most of the robots used in the industry present a serial architecture that can somehow be related to a human arm. In those robots, each joint must be actuated to fully control the end-effector. The main drawback of such open-loop structures is that each motor must carry the weight and inertia of the motors further in the chain over the workspace. Also, an open structure is in general less stiff than a closed structure. High inertia and low stiffness lead to a poor mechanical transparency that is limiting the robot performance.

On the other hand, it is possible to use a mechanical structure in which the links and joints form closed-loops. Closing the loop of a kinematic chain has profound consequences since the closure introduces a dependency on the joints velocity and the assembly of the closed-loop can generate internal stresses in the structure that are present even when no external force is applied to the mechanism. However, since a dependency in the joints is created, it is now possible to control the whole structure by placing motors on only a subset of the joints, ideally joints that are located on the base of the robot so the motors have little contribution to the inertia and heavier, more powerful motors can be used.

In addition, the closed mechanical structure that is now linking the motors and the end-effector will have in general a better stiffness than a comparable serial structure. This increased stiffness can be used to reduce the size of the mechanical links, thus reducing even more the total inertia.

The ratio between the stiffness and the inertia for a given structure can be called the structural stiffness. The structural stiffness can also somehow be related to the ratio between the payload and the total moving mass. Parallel robots are generally used for tasks in which a high structural stiffness in needed. Typical applications include flight simulators, high-speed pick-and-place machines and haptic devices. In a flight simulator, the payload is so high (often 3 tons or more) that it would be difficult to imagine a fast reacting robot based on a serial architecture performing this task. In the pick-and-place industry, the number of cycles per minute is the key factor in the profitability of a robotic system and the maximum speed of the robot is directly limited by its structural stiffness. Finally, haptic devices need a high stiffness to properly render the force interactions with a remote or virtual environment but they need to combine this with a low inertia so that the forces perceived by the operator are not disturbed by the device.

1.1.2 Configurable Platforms

Pure parallel mechanisms are formed by two rigid links, called the base and the end-effector, connected in parallel by serial chains, called legs. The novel concept behind parallel mechanisms with configurable platforms is that the rigid link (non-configurable) end-effector is replaced by an additional closed-loop chain (the configurable platform). Some of the links of this closed-loop are attached to the legs so its configuration can be fully controlled from the motors located on the base.

The use of a closed-loop instead of a rigid end-effector allows the robot to interact with the environment from multiple contact points on the platform. The contact points have a relative mobility between each other that can be fully controlled from the actuators located on the base. This results in a robot that can combine motions and grasping capabilities into a structure that provides an inherent high structural stiffness. High-speed pick-and-place robots and haptic devices can both benefit from this type of architecture. In both applications, the addition of grasping capabilities can be beneficial for some tasks and a high mechanical structural stiffness will improves the performance of the device given a certain hardware and software control system.

1.1.3 Some Definitions

Research over parallel mechanisms has grown exponentially in the last decades and this sudden grown interest came with a lack of well-established terminology. This results in a lack of coherence in the research community and some articles, despite overlapping subjects, unfortunately fail to cite other papers properly partially due to the use of different terminologies. To avoid confusion and help coherence, this section presents the definition of some common terms that will be used in this thesis. The definitions are based on what seems to be most commonly accepted in literature but are also made to suit well the structure and content of this thesis, allowing an easier reading.

Mechanism The International Federation for the Promotion of Mechanism and Machine Science (IFToMM) defines a mechanism as “Constrained system of bodies designed to convert motions of, and forces on, one or several bodies into motions of, and forces on, the remaining bodies”. In the first part of this thesis, a mechanism is defined as “a set of rigid bodies with pairwise connections constraining their relative motions”. The bodies, considered infinitely stiff, will be called “links” and the constrainning connections will be call “joints”. A link can share an infinite number of joints but a joint strictly connects two links. Joints can be revolute, prismatic or allow more than one independent degree of freedom (DOF) between its two connected links. In this thesis, only 1-DOF joints are considered without losing generality since multi-DOF joints can be virtually described as a series of 1-DOF joints with zero-lenght links in between.

Pure Parallel Mechanism The term pure parallel mechanism is not officially recognized by IFToMM but has been defined in some articles [85, 49, 18] as “A mechanism that is composed of an end-effector connected to the base by independent, serial leg chains”. The vast majority of research presented on closed-loop mechanisms refers to this particular class and this definition can be used to differentiate mechanisms that can be treated with standard analysis methods[11] or type synthesis[42] methods from mechanisms that have more exotic topology such as Delta or hybrid mechanisms.

Parallel Mechanism with Configurable Platform In the first part of this thesis, a parallel mechanism with configurable platform (PMCP) is defined as “a parallel mechanism in which the rigid end-effector is replaced by a single closed-loop”. Some of the links of the closed-loop configurable platform are connected to the legs while others are not. The links that are connected to both the platform and the legs are called “platform connector” while links that are not connected to any leg are called “platform segment”.

Robot There is no general definition of the term robot that satisfies everyone. Some authors claim that “as soon as it is possible to control several degrees of freedom of the end-effector via a mechanical system, this system can be called a robot” [51]. But the term robot is also sometime used for 1-DOF devices that have a high level of decision-making and cognitive functions. A car with an embedded computer can sometime be called a robot depending on the level of sensors and programmability. A CNC milling machine is rarely called a robot even if it has most of the characteristics to be called such. Robots in popular culture are mainly associated to the class of humanoid robots. Common thought among engineers is that they don’t know what a robot is but they can recognize one when they see one.

Clearly, a robot must have at least the capability to interact with the environment in an automated and programmable way. In the structure of this thesis, a robot is simply “A mechanism in which the motion and forces of some of the joints are controlled with an automated and programmable motor-sensor system”. This means that some of the joints of the mechanism are now considered as the inputs of the system and some links are now considered as the outputs of the system.

Fully Parallel Robot A fully parallel manipulator is defined as “A Parallel robot for which the number of legs is strictly equal to the number of DOF of the end-effector”[24]. Because of this property, the control of these parallel robots can be done using a single base-located motor in each leg. In a fully parallel robot, the grounded motors have little contribution to the inertia of the mechanism and a more powerful motor can be used. In this thesis, it is assumed that the the analysed robots are fully parallel.

Parallel Robot with Configurable Platform The results presented in the first part of this thesis are characteristic for the mechanism itself and are independent of which joints and links are considered as the inputs and outputs of the system. The second part of this thesis focuses on the concepts that are related to the inputs – outputs relations of parallel mechanisms with configurable platforms. A parallel robot with configurable platform (PRCP) is "a robot based on a PMCP structure, in which each leg hosts a single motor actuating the joint located on the base". The actuated joint variables are considered the inputs of the system and the variables that describe fully the pose and the configuration of the configurable platform are the outputs of the system.

1.2 State of the Art

1.2.1 Literature

While a relatively large number of articles are published each year on topics related to parallel robots with rigid, non-configurable platforms, articles concerning parallel robots or parallel mechanisms with configurable platforms are very scarce even when considering all possible different terminology that could be used to describe this concept.

The first article about a PRCP is probably from Yi and al. [81] who proposed in 2002 a planar parallel mechanism with a parallelogrammic planar platform that can be folded. Based on the special geometry of the platform, they described the kinematic analysis of their mechanism. The analysis they proposed was specific to this particular design without generalization of the concept and the spatial case was not addressed.

Mohamed and Gosselin [56] proposed in 2005 a first generalization of the concept of both planar and spatial PRCPs. They emphasised the general concept of PRCPs and proposed several designs of planar robots and one design of spatial robot with prismatic actuators. They proposed a general framework to compute the mobility of these mechanisms but this framework was not valid for overconstrained mechanisms and the case of overconstrained PRCPs was not addressed.

In the title and introduction of their paper, they call those robots "kinematically redundant parallel manipulators with configurable platforms" where kinematic redundancy means that the mobility of the manipulator is greater than the required degrees of freedom. According to this definition, kinematic redundancy is a concept that is only defined relatively to the task and not a property of the mechanism itself. While the concept of kinematic redundancy can be well applied to a 6D robot executing a 5 DOF wielding task or a 7 DOF robot arm moving in space, it is difficult to apply it to PRCPs because the configurable platform doesn't have a single physical end-effector to perform the required main task. As long as it is accepted that PRCPs have more than one end-effector and that the grasping capabilities are parts of the task of the robot, the concept of kinematic redundancy doesn't apply to them.

Other articles in the literature that focus on specific architectures of PRCPs include the design of the Par4, by Company and al. [59] in 2005. Like for Yi, their robot also uses a

1 DOF paralellogrammic configurable platform, but in this case, the configuration of the platform is used to provide rotation via a gearing system instead of grasping. Although they already introduced the concept of “articulated nacelle” in earlier articles with the H4 and the I4 robot, the nacelle of these two robots was not formed by a closed-loop chain but was simply a two by two connection of the legs. Such architectures are not considered as PRCPs since they don’t have a closed-loop platform that allows multi end-effectors and grasping capability. The Heli4 from the same laboratory also connects four legs two by two and can also be defined as “robots with articulated nacelle”. Their specific kinematic and singularity analysis of these robots was presented. It was also found in their research that the Par4, their only robot with a PMCP structure, has much better dynamic properties than the H4, I4 and Heli4 robot.

Due to the large number of publications concerning parallel robots in general, it is not possible to shortly give an overview of the state of the art of all the topics in particular. The first book solely dedicated to parallels robots was written by Merlet in 2000[51] and contained already more than 650 citations. Type synthesis is the branch of parallel robots that focuses on systematic methods to create new architectures given a certain motion requirement. Current developed methods include Kong and Gosselin[42] in 2007, which addressed the synthesis of pure parallel robot, and Gogu[23] in 2010, which addressed more complex type of legs. It should be noted that none of the current type synthesis methods include PRCPs.

Unlike general parallel robots, PRCPs can interact with the environment from multiple contact points, thanks to the configurable platform. They are therefore very suitable for applications that require grasping capabilities and a high structural stiffness, such as haptic devices. There are currently a few haptic systems that can generate grasping capability. In those devices, the master robot interacts with the operator through two distinct end-effectors or two tips. This provides a multi point contact that is transferred from virtual or distant objects to the human operator and the operator can use his index and thumb to grasp and interact with the objects in a more natural way than with a single point contact. The solution generally used to provide grasping capabilities is to add a gripper at the top of an existing robot [60]. However, in case of parallel robots, their good performance relies mainly on the fact that all their actuators are located on the base. Since grasping forces are often in the same order of magnitude than the other types of force, the mass of the motor that is needed to actuate the gripper may considerably increase the total inertia, as this motor has to be positioned over the whole workspace.

Another solution, sometimes used in haptic systems, consists in coordinating two independent devices, one per finger tip, to simulate the grasping sensations [57],[53]. This is done at the expense of unnecessary complexity and inertia, and the total system generates usually more degrees of freedom than needed between the two finger tips. In the second part of this thesis, the design and implementation of the PentaG device, a novel 5 DOF robot based on a PMCP structure, is presented for haptic applications. To the knowledge of the author, it is the only haptic device that provides grasping capabilities with all the motors located on the base and without the use of cable transmission that would increase friction and decrease accuracy.

1.2.2 Industry

The only PRCP currently commercially available on the market is the Adept Quattro, which was initially developed by Fundacion Fatronik, who integrated the Par4 design into a commercial product. The targeted market is the fast pick-and-place and assembly production lines. In this design, the 1 DOF configurable platform is not used to provide grasping capabilities but is used to provide rotation of the end-effector via a gearing system. In a standard Delta robot, the rotation is provided by a central shaft with a passive prismatic joint that have the tendency to limit the dynamic performance of the robot and therefore the maximal number of cycles per minutes that the robot can achieve.

Thanks to the use of a configurable platform for the rotation of the end-effector, the central prismatic leg could be replaced by a fourth Delta leg. Those types of leg are recognised for their excellent dynamic properties. The robot was launched into the market in 2007 and has gained enormous popularity since then. Adept developed additional advanced control strategies and embedded vision systems and they now claim to have the fastest robot of the whole pick-and-place industry.

To the best knowledge of the author, no commercial device, neither in the pick-and-place or in the haptic market, uses currently a configurable platform to provide grasping capabilities. In the pick-and-place industry, robots use generally a vacuum gripper to handle the products. However, vacuum gripping is often not possible when the products present non-flat, porous or irregular surfaces as common in the food industry and mechanical grippers must then be mounted on the rigid platform, increasing the total inertia and degrading the robot performance. In the PentaG architecture, all the motors are located on the base and the device can be in principle lighter and faster than comparable robots with actuated mechanical gripper.

1.3 Objectives and Contributions of this thesis

The first part for this thesis focuses on fundamentals of parallel mechanisms with configurable platforms (PMCPs). Mechanisms that do not satisfy the Chebichev criterion of mobility have fascinated kinematicians since decades. Those mechanisms have been called exceptional or paradoxical. They fail to satisfy the mobility criterion because their constraint equations are not full rank. They are nowadays generally called overconstrained mechanisms. Mobility and oversconstraint of mechanisms are intimately linked and must be considered at once. Each overconstraint is associated to an additional underconstraint, also known as degree of freedom which increases the total mobility of the mechanism in comparison of the prediction to the Chebichev criterion. Once this indisputable fact is established, one can now wonder how to determine the rank of the constraint equations in a mechanism.

Constraint equations can be organized into a matrix form for which the rank is computed when numerical values are used. However these numerical values are dependent on the configuration of the mechanism. The number of DOF and the direction of the mobility obtained are only valid for this particular position of the mechanism. The computation of the rank of a matrix with numerical values is also sensitive to round-off errors and exact values for the geometry of the mechanism must be used, which is not realistic in practice. Finally, not only the direction of the mobility may be dependent on the mechanism configurations but also the number of degree of freedom and the number of overconstraints may change in particular configurations such as serial singularities and constraint singularities.

Screw Theory is a powerful conceptual framework to analyse the kinematics and statics of spatial mechanisms. The compactness of the notation and the geometrical insights given by this framework are very useful to calculate analytical vector based representations that are independent of the particular configuration of a mechanism. Most of the basic vector space operations on common screw systems have already been studied analytically and those results can be used directly in more complex analyses.

Graph Theory can be described as the study of pairwise relations between a set of objects. While Screw Theory is concerned with the particular geometrical relations between the joints of a mechanism, Graph Theory, when applied to a mechanism, only focuses on the connectivity between the various links of a mechanism from a network point of view, regardless of the type and position of the joints connecting those links. A complete description of a general mechanism in a particular configuration can be obtained by assigning a screw system to each edge of its corresponding graph. Mobility and overconstraint analysis of pure parallel mechanisms exploit the relatively simple topology of these mechanisms in the derivation of their equations. Properties of the corresponding graph become handy when analysing mechanisms that have a more exotic topology such as Delta mechanism, hybrid mechanism and PMCPs.

PRCPs have been so far scarcely investigated in the literature or used in the industry. Especially, the mobility of overconstrained PMCPs hasn't been addressed yet in a general analytical vector representation. Mobility and overconstraint are among the most fundamental aspects of mechanisms. Not so long time ago, the only way to verify the full-cycle mobility and the possibility of assembly of a mechanism was to actually build it to observe its behaviour. This is less required nowadays with the advance of CAD programs in which the mobility and assembly of any mechanism can be tested relatively quickly. Still these software programs usually rely on numerical iterations to compute the assembly of the mechanism so they are not aware of the geometrical conditions needed for the assembly. These conditions link the manufacturing tolerance of the parts to the stiffness and play in the mechanism and must be evaluated before the construction of the prototype. In addition to a more purely scientific contribution to the field of mechanism science, the theoretical analysis of mobility and overconstraint of PMCPs presented in the first part of this thesis can help engineers and designers to consider these alternative mechanisms as valid options and is also an essential aspect in the future development of a broader type synthesis method that includes those mechanisms.

Therefore the objective of the first part of this thesis is to investigate the fundamentals of mobility and overconstraints of PMCPs and to provide a general method to obtain a configuration-free, analytical vector representation of the distribution of their mobility and overconstraint via Screw Theory and Graph Theory.

The second part of this thesis focuses on the analysis of parallel robots with configurable platforms (PRCPs) and the specific analysis, optimization, design and implementation of the PentaG robot, a novel 5 DOF PRCP that include a 2 DOF configurable platform. In the second part of this thesis, concepts that are related to the input – output relations of PRCPs, i.e. robots that are based on a PMCP structure are investigated.

Obviously, robots are expected to accomplish certain tasks and task requirements are needed in their optimization and design process. Parallel robots are recognized for their inherently high structural stiffness. Unlike general parallel robots, PRCPs can interact with the environment from multiple contact points, thanks to the configurable platform. They are therefore very suitable for applications that require grasping capabilities and a high structural stiffness such as pick-and-place devices and haptic devices. The novel PentaG robot is a good example of a PRCP and its kinematics analysis, optimization and detailed design for haptic application is presented. The particular mobility and overconstraint of the PentaG mechanism are properties of the mechanism itself that are independent of the choice of the actuated joints. The direct and inverse kinematics, dexterity, statics, singularities, working modes and assembly modes are based on the input − output relations between the robot actuators and end-effectors.

A motivation for the design and implementation of the PentaG haptic prototype is to see to which extend the general design principles and analysis tools that are used in the study and implementation of parallel robots can be applied in the design and analysis of PRCPs. The first step in the design of a parallel robot is the choice or the creation of a mechanism that has the desired mobility. It is aimed for that the fundamentals of PMCPs covered in the first part of this thesis can help robot designers to consider new PMCPs that satisfy their mobility requirements. The second step is generally the computation of the kinematic relations between the actuators and the end-effector of the robot. General formulas for the kinematics and singularity analysis of parallel robots are based on their relatively simple topology and modified formulations should be investigated for the kinematic analysis of PRCPs. The next steps are the geometrical optimisation, the detailed design based on elastic body considerations, and finally the drive system and control system design.

The objective of the second part of this thesis is therefore to show the feasibility of PRCPs that use the configurable platform as a grasping device and to show how the general design principles of conventional parallel robots can be applied directly or must be modified for the design of a PRCP. During the design process, the applicability of the new techniques that are developed for the PentaG robot design will be extended to PRCPs in general or extended to all parallel robots when possible.

1.4 Structure of this thesis

The thesis is structured into two distinct parts. The first part includes Chapter 2 and 3 and focuses on the fundamentals of mobility and overconstraint analysis of PMCPs and the properties of those mechanisms regardless if the system is actuated or not. The second part includes Chapter 4, 5 and 6 and focuses on the input-output relations of PRCPs and presents the design, optimization and implementation of the PentaG robot for haptic application. When possible, the results obtained in the design process of the PentaG are generalized to any PRCPs.

Chapter 2 presents a state of the art method in the mobility analysis of overconstrained parallel mechanisms. The method presented is not based on one article in particular but instead uses elements and techniques from various published methods. The basics of Screw Theory are introduced and a 4-RRR parallel mechanism is used as an example to show how the method combined with an analytical vector representation can be used to generate an analytical solution of the distribution of mobility and overconstraints in this mechanism. In particular, the concept of restricted screw systems is introduced to calculate analytically both the local mobility and overconstraint using vector space operations of screw systems.

Chapter 3 extends the method presented in Chapter 2 to mechanisms that are not purely parallel and to PMCPs. The basics of Graph Theory are introduced and are used to analyse the topology of those mechanism. Current existing methods for the mobility and overconstraints analysis of mechanisms that are not purely parallel but have a series-parallel graph are unified with a graph reduction analysis of their topology and recursive formulas are derived for the general case. It will be shown that PMCPs have a wheel graph that is non series-parallel and a method based on a matrix representation of the mobility of their legs and the concept of restricted screw systems is used to obtain an analytical distribution of their global mobility. The special self duality of PMCPs is then exploited to extend the method to the overconstraint analysis. Two mechanisms, a 4-RRR with a 8R configurable platform and the PentaG mechanism are used to illustrate the method and an analytical representation of the distribution of their mobility and overconstraints is presented.

Chapter 4 introduces the novel PentaG robot and its kinematic analysis including an analytical representation of its inverse position and velocity kinematics. Unlike general parallel robots, a two stages method must be used in the calculation of the inverse kinematics of general PRCPs. Direct position and velocity kinematics, statics, dexterity and stiffness are obtained from numerical procedures using the inverse velocity kinematic as input. Serial singular configurations are defined analytically and the working modes of the PentaG robot are enumerated. Parallel singular configurations are obtained numerically. New types of serial and parallel singular configurations that are specific to PRCPs are described. In addition, a general method to statically balance from springs translational parallel robots with or without configurable platform from springs is presented.

Chapter 5 presents the geometrical optimization of the PentaG robot when used as a haptic device. The concept of useful workspace is presented and the robot is optimised in order to maximise its compactness relatively to the size of the useful workspace. The optimization problem is tedious due to a large number of geometrical parameters and a complex discontinuous objective function that has several local optima and is computational intensive. Several techniques are shown to reduce the parameters to a number that is manageable and a more efficient algorithm is developed to reduce the computation time of the objective function. Due to the discontinuities and local optima of the objective function, Genetic Algorithms are used in the search for a global optimal solution. The final results are presented using an atlas of the dexterity of the robot in the horizontal plane for various heights, rotation and grasping values.

Chapter 6 shows the detailed design and implementation of a prototype of the PentaG robot for haptic application. The kinematic parameters obtained in Chapter 5 are used in this prototype and a new technique to optimise the structural stiffness of the robot without modifying its kinematic parameters is presented. Specifications are defined at the finger tips of the robot based on human ergonomic and the propagation of those specifications to the design of the motor-sensor system is done using a statistical distribution over the useful workspace. The implementation of the final design is presented and both simulations and simple experiments are used to show the validity of the final prototype and method.

Finally, Chapter 7 gives a list of what are considered the claims and original contributions that can be drawn from this thesis.

Part I

Parallel Mechanisms with Configurable Platforms

Chapter 2

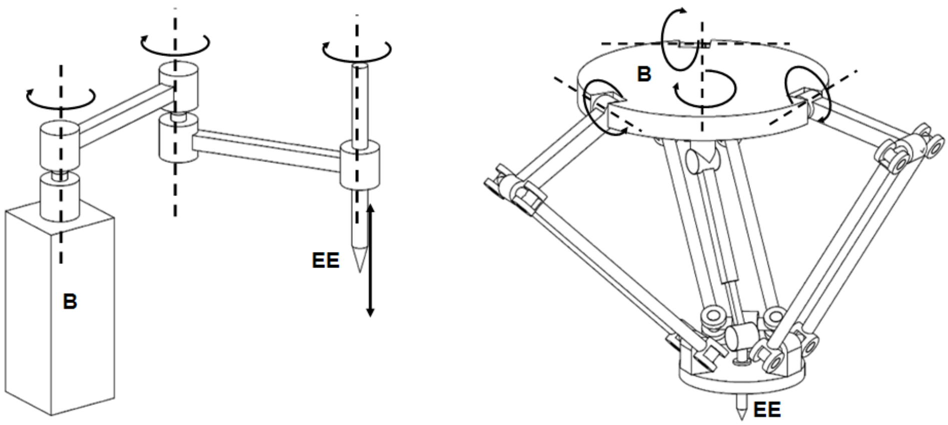
Mobility and Overconstraints Analysis of Parallel Mechanisms

2.1 Introduction

Serial robots are the most common robots in use in the industry. A serial robot is formed by a set of n links connected by $n-1$ kinematic pairs or joints. Since they have no closed-loop, all joints are independent and must be actuated by a motor in order to fully control the robot. Serial robots may have a structure which is anthropomorphic to the human arm, with a series of shoulder, elbow and wrist joints. One of the most common types of robot used in the industry is the Scara Robot, a 4 DOF serial robot that can perform translations in the 3D space and rotation around the vertical axis. Such motions are called Schoenflies motions [34] and are typically encountered in pick-and-place and packaging industries. Figure 2.1 a) shows a schematic representation of a Scara Robot. The base is noted B and the end-effector EE. All joints are actuated and the possible movements of the four motors are represented with arrows.

In a parallel robot, the end-effector is linked to the base by a set of independent chains acting in parallel. Due to the presence of closed-loops, the joints of parallel robots are not independent and only some of them, generally located on the base, need to be actuated in order to fully control the robot. Over the last two decades, parallel robots have increasingly replaced their serial counterpart in a large number of applications. Typical applications include flight simulators, high-speed pick-and-place machines, haptic devices and machine tool. Figure 2.1 b) shows a schematic representation of 4 DOF Delta robot that can perform the same type of movement than the Scara robot. Although they can perform the same task, these two robots are very different in structure and performance. These two examples are used to illustrate the difference between serial and parallel robots.

The main drawback of serial robots is that each motor has to carry and move the weight of the next motor in the chain. In a typical Scara robot, the mass of the motors represents a large part of the total moving mass and inertia. In a parallel robot such as the Delta, all the motors are grounded on the base and have little contribution to the inertia of the mechanism. A second drawback of serial robots is the fact that an open mechanical structure has less stiffness than a comparable closed structure. The obvious result of



a) b)

Figure 2.1: Schematic Representation of a) a 4 DOF Scara Robot and b) a 4 DOF Delta Robot

these two characteristics is that serial robots have a much lower payload-to-weight ratio than parallel robots. The payload-to-weight ratio can be considered as a measure of the structural stiffness of a robot.

In any application that requires a high structural stiffness, parallel robots will normally show a better performance. For flight simulators, the payload is so high that the use of a serial robot is not possible. For the pick-and-place industry, the inertia of the robot must be reduced and the stiffness increased in order to reach a very high speed. Finally, when using a haptic device, mass should be decreased so the human operator doesn't feel the inertia of the device itself, and stiffness should be increased so the he can feel clearly the forces from the motors.

The main disadvantage of parallel robots is that they have a workspace that is usually smaller than the robot itself. Another disadvantage is that the workspace may include some singular positions, in which one loses control of the mechanism. The first parallel robot is generally attributed to Gough [29] who designed a 6 DOF parallel platform to test tires wear and tear.

For maximum application flexibility, a robot should have 6 DOF. For many applications, less than 6 DOF are required and there are advantages in these cases to use a limited DOF parallel robot. Limited-DOF parallel robots have a less complex mechanical structure, lower manufacturing cost, and a simple control algorithm [40]. Typically, the first and probably the most crucial step in the design of a parallel robot is the choice of the closed-loop structure that will satisfy the motion requirements from the application. The choice of the mechanical structure will greatly influence the results of the following steps in the design process, such as the geometric optimization and control system design. As opposed to serial structures, there are a very large number of parallel structures that can lead to the same type of mobility.

The presence of closed-loops in parallel mechanisms introduces dependencies on the joints of the mechanisms which makes the computation of their mobility a rather complex process. In addition, the closed-loops may also introduce overconstraints in the mechanism.

Over-constraints must carefully be taken into account regarding the manufacturing tolerances in order to be able to assemble the mechanism. Clearly, a good general understanding of the concepts of mobilities and overconstraints in parallel mechanisms can lead to a better choice of the appropriated mechanical structure. Mobility of over-constrained closed-loop mechanisms is a subject that is more than 150 years old and still an important research topic in parallel robotics.

The first section of this chapter presents the current knowledge about the mobility analysis of parallel mechanisms. The method described here relies heavily on the use of screw theory, which is also introduced in this chapter. It seems that the major concern in recent publications about mechanism mobilities [22, 11] is the computation of local mobilities and the nature and distribution of the overconstraints. Both overconstraints and local mobilities can be seen as screw redundancies in the mechanical system. In the second section of this chapter, an original mathematical framework is introduced to calculate the local mobilities and overconstrains. The last section of this chapter discusses the advantages of the proposed method.

2.2 Standard Method for the Mobility Analysis of Overconstrained Parallel Mechanisms

In this section, a state-of-the-art method based on screw theory is presented to determine the mobility of overconstrained parallel mechanisms. The mobility or spatial degree of freedom (DOF) of a mechanism is defined by IFToMM as "the number of independent coordinates needed to define the configuration of a kinematic chain or closed-loop mechanism". As it will be shown in this section, the concept of mobility is closely related to the concept of overconstraints, since mobilities can also be considered as under-constraints. Motion and constraints of parallel mechanisms are most naturally described by screw theory, for which a quick introduction is provided. The method is finally illustrated by the analysis of a 4-RRR parallel mechanism.

2.2.1 Modified Chebychev–Grübler–Kutzbach Criterion

A first formula for the calculation of the mobility of mechanism is generally attributed to Chebyshev [9] in 1869. Further contributions came from Grübler [30] in 1884 and from Kutzbach [43] in 1929 to form the Chebychev–Grübler–Kutzbach mobility criterion. When considering a general mechanism with n rigid links and m joints, each joint having f_i DOF. Most authors also consider a parameter d, which stands for the mechanism motion system dimension. $d = 6$ in case of a spatial mechanism and $d = 3$ for a planar or a spherical mechanism. However, planar and spherical mechanisms can be considered as special cases of overconstrained spatial mechanisms and an analysis with $d = 6$ can be done without loss of generality. The Chebychev–Grübler–Kutzbach criterion for the mobility M, the number of independent parameters needed to describe the configuration of a spatial mechanism, is given by

$$M = 6\,(n - m - 1) + \sum_{i=1}^{m} f_i \qquad (2.1)$$

The Chebyshev formula can be understood physically in two ways that lead to the same result. A first approach is to consider that all the rigid links aren't initially connected to each other and float freely into space. Each rigid body has 6 DOF and the total freedom of the system is given by $6\,(n-1)$, since one link must be considered has the reference. Each joint allows f_i DOF of mobilities and constraints $(6-f_i)$ DOF between 2 links. The total number of constraints is therefore given by $\sum_{i=1}^{m}(6-f_i)$. The mobility M of a closed-loop mechanism is given by

$$M = 6\,(n-1) - \sum_{i=1}^{m}(6-f_i) \tag{2.2}$$

which is equivalent to Equation 2.1. A second approach is to consider that for each closed-loop, one link is virtually cut in half in order to open the loop. Initially, the mechanism has n links, m joints and $m-n+1$ independent closed-loops. Without any closed-loop, all m joints are independent and the complete mechanical system has $\sum_{i=1}^{m}(f_i)$ DOF. The $m-n+1$ half-split links are now glued together to reconstruct each loop. Each time the two halves of a link are alligned to reconstruct it, 6 DOF are removed from the system. Therefore, the mobility M of a closed-loop mechanism is given by

$$M = \sum_{i=1}^{m}(f_i) - 6\,(m-n+1) \tag{2.3}$$

which is also equivalent to Equation 2.1. However, people realized quickly that many mechanisms had a different mobility than the one predicted by the Chebychev–Grübler–Kutzbach criterion. This is due to special geometrical arrangements that introduce redundancies in the constraints. In both physical interpretations, it is indeed possible that certain constraints were imposed on degrees of freedom of the system that were already constrained. These redundant constraints are called overconstraints. This phenomenon was already pointed out in the 60's by Waldron [77] among other authors. If the number of over-constraint is noted R_C, the actual mobility of a mechanism is given by

$$M = 6\,(n-m-1) + \sum_{i=1}^{m}(f_i) + R_C \tag{2.4}$$

This equation is valid for any mechanism. The problem of course is now to calculate R_C, the number of overconstraints. The term M in Equation 2.4 represents the mobility of the whole mechanism, including the mobility that may be locally present in redundant serial chain. Those local mobilities are independent of the end-effector motion and are present even when the end-effector is fully constrained. In such case, M will be greater than the number of DOF between the base and the end-effector. Some authors prefer to note M as the mobility between the end-effector and the base, and add a term, that is noted here R_M, which corresponds to the total number of local DOF. In his review of mechanism mobility criteria in 2005, Gogu [22] proposed to consider the local mobilities separately. A form of this modified criterion can be given by

$$M = 6\,(n-m-1) + \sum_{i=1}^{m}(f_i) + R_C - R_M \tag{2.5}$$

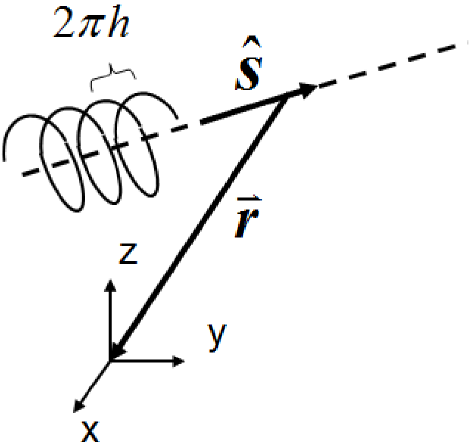


Figure 2.2: The Screw

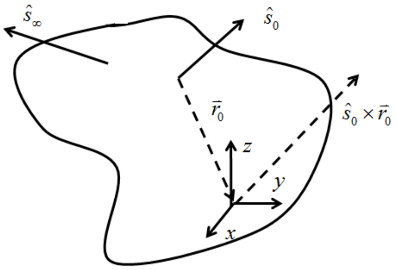
The problem being to calculate R_C and R_M, the number of overconstraints and local mobilities present in the mechanism. In order to do this, more information is needed than simply the amount of joints and links but the geometrical arrangements of the constraints imposed by the joints in space must be considered. This leads not only to the right number of DOF in closed-loop mechanisms, but also to a description of the orientation and distribution of the mobilities and constraints within the mechanism. For this purpose, screw theory provides an elegant and intuitive framework.

2.2.2 Basic Results From Screw Theory

Screw theory is a conceptual and mathematical framework that combines linear and rotational velocities, and linear forces and moments into a 6 dimensional vector space. The theory finds its origin in the work of the Irish astronomer Sir Robert Stawel Ball [1], and was further developed for kinematic analysis by Hunt [36], Roth [61], Waldron [77], Phillips [65], and Dai [12] among others. The two fundamental theorems are the Poinsot theorem in 1803, which states that any system of forces and moments applied to a rigid body can be replaced by a single force combined with a single couple along the same axis, and the Chasles theorem in 1830, which states that any displacement of a rigid body can be replaced by a single rotation combined with a single translation along the same axis. The theorem is generally attributed to Chasles althougt Mozzi and Cauchy are credited with earlier results that are similar.

2.2.2.1 The Screw

A screw is a geometric element formed by a straight line in 3D space to which a scalar called pitch is associated. A vector representation of a unit screw $\hat{\$}$ is given by



	Twist	Wrench
Zero Pitch Screw $\begin{bmatrix}\hat{s}_0 \\ \hat{s}_0 \times \bar{r}_0\end{bmatrix}$	Revolute Joint	Force Constraint
Infinite Pitch Screw $\begin{bmatrix}0 \\ \hat{s}_\infty\end{bmatrix}$	Prismatic Joint	Moment Constraint

Figure 2.3: A zero pitch screw and an infinite pitch screw used as a twist or a wrench. An infinite pitch screw is a directed free vector. The first part of a twist represents an angular velocity while the second part represents a linear velocity at the origin of the reference frame. The first part of a wrench represents a linear force while the second part represents a torque applied at the origin of the reference frame.

$$\hat{\$} = \begin{bmatrix} \$_F \\ \$_S \end{bmatrix} = \begin{cases} \begin{bmatrix} \hat{\mathbf{s}} \\ \hat{\mathbf{s}} \times \mathbf{r} + h\hat{\mathbf{s}} \end{bmatrix} & \text{if } h \text{ is finite} \\ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}} \end{bmatrix} & \text{if } h = \infty \end{cases} \tag{2.6}$$

where $\$_F$ and $\$_S$ are the first and second part of the screw and are 3 dimensional vectors, $\hat{\mathbf{s}}$ is a unit vector directed along the screw. $\mathbf{r}$ is a vector from any point along the screw to the origin of the coordinate frame, and h is the scalar pitch. This is shown on Figure 2.2.

A unit screw can be multiplied by magnitude scalar λ to either form a twist, representing instantaneous velocity at the origin of the coordinate frame, or a wrench, representing instantaneous forces at the origin of the coordinate frame. The first part of a twist is an angular velocity around $\hat{\mathbf{s}}$ and the second part is the translational velocity. The first part of a wrench is a linear force and the second part is a couple. In a mechanism, a zero pitch screw represents a revolute joint or a linear force while an infinite pitch screw represents a prismatic joint or a torque. This is summarised in Figure 2.3.

2.2.2.2 Screw Systems

The configuration space of a single rigid body in the 3D Euclidean space is described by the Special Euclidean group of transformations SE(3) which has 6 DOF. A 6-dimensional vector space based on the geometry of screws can be constructed to represent instantaneous velocities on those configurations. A $n \leq 6$ screw system is a vector subspace within the screw vector space that is formed by all the screws that can be obtained from a linear combination of n independent basis unit screws. The basis of a screw system is not unique and various sets of basis screws can be used to describe the same screw system, as long as the n basis screws are independent. A n screw system S with unit basis screws $\hat{\$}_i$, $i = 1..n$ is noted as

$$S = \left\{ \hat{\$}_1, \cdots, \hat{\$}_i, \cdots, \hat{\$}_n \right\} \tag{2.7}$$

S is used to represent a general screw system. A twist system is noted T and represents velocities while a wrench system is noted W and represents forces. The reciprocal product between two screws $\$_1$ and $\$_2$ is given by

$$\$_1 \circ \$_2 = \$_1 \begin{bmatrix} 0_{3,3} & I_{3,3} \\ I_{3,3} & 0_{3,3} \end{bmatrix} \$_2 \tag{2.8}$$

The reciprocal product between a twist and a wrench on a rigid body gives the power that is transmitted to the body. Two screws $\$_1$ and $\$_2$ are said to be reciprocal if $\$_1 \circ \$_2 = 0$. Two screw systems S_1 and S_2 are reciprocal if all the screws in S_1 are reciprocal to all the screws in S_2. Any screw system S of dimension n has a unique reciprocal screw system $S^{\perp}$of dimension $(6 - n)$.

$$S = \left(S^{\perp}\right)^{\perp} \tag{2.9}$$

where $()^{\perp}$denotes the reciprocal screw system. Considering two rigid bodies, the twist system T , representing their relative mobility, is reciprocal to the wrench system W, representing their relative constraints, and therefore $T = W^{\perp}$ and $W = T^{\perp}$. Screw systems are vector subspaces of the Lie algebra se(3) and two basic operations, the sum and intersection, are often performed on them during kinematic analysis. The sum of two screw systems S_1 and S_2 is given by all the screws that can be generated by all linear combinations of the basis screws of both S_1 and S_2 combined and is noted $S_1 + S_2$. The intersection of two screw systems S_1 and S_2 comprises all the screws which are common to both S_1 and S_2 and is noted $S_1 \cap S_2$. Note that, unlike the sum of sets, the sum of vector spaces $S_1 + S_2$ include vectors which doesn't belong to neither S_1 or S_2 but are obtained by linear combination of screws from both systems. If the intersection of two screw systems is zero $S_1 \cap S_2 = \{0\}$, a valid basis for $S_1 + S_2$ can be obtained by the union of the two sets of basis screws of S_1 and S_2. The sum of two non-intersecting screw systems is called the direct sum and is noted $S_1 \oplus S_2$.

2.2.3 Mobility of Overconstrained Parallel Mechanisms

A pure parallel mechanism is a mechanism in which two links, the base and the end-effector, are linked in parallel by two or more independent chains. The origin of the method can be found in the works of Waldron [77] who first established the serial and parallel laws for instantaneous kinematics. The serial law states that the mobility between the two end links of a serial chain is given by the sum of the twist systems of all joints within the chain. For a serial chain i with q joints $j = 1..q$, the twist system is expressed as

$$T_i = \sum_{j=1}^{q} T_{ij} \tag{2.10}$$

where T_{ij} is the twist system of joint j, and T_i is the twist system between the 2 end links of the serial chain i. For example, if the axis of rotation of a one-dof revolute joint is given by $\hat{\mathbf{s}}$ and the position of the joint relatively to the reference frame is given by $\mathbf{r}$, the corresponding twist system is given by $T_{ij} = \begin{bmatrix} \hat{\mathbf{s}} & \mathbf{r} \times \hat{\mathbf{s}} \end{bmatrix}^T$, where the second part

$\mathbf{r} \times \hat{\mathbf{s}}$ represented the linear velocity created at the reference frame by the rotation of the joint. The parallel law states that the mobility between 2 links connected by parallel independent chains is given by the intersection of the twist systems of all chains. For a parallel mechanism with $i = 1..p$ independent chains, the twist system is expressed as

$$T = \bigcap_{j=1}^{p} T_i \tag{2.11}$$

where T_i is a vector space representing the twist system of the serial chain i. A dual set of laws exists for the wrench systems of parallel mechanisms. In a serial chain, the constraints system between the 2 end links is given by the intersection of the wrench systems of all joints within the chain. For a serial chain i with $j = 1..q$ joints, the wrench system is

$$W_i = \bigcap_{j=1}^{q} W_{ij} \tag{2.12}$$

where W_{ij} is the wrench system of joint j and W_i is the wrench system of the serial chain i. The constraints between 2 links connected in parallel by independent chains is given by the sum of the wrench systems of all chains. For a parallel mechanism with p independent chains $i = 1..p$, the wrench system is

$$W = \sum_{i=1}^{p} W_i \tag{2.13}$$

For any screw system, either representing a single joint, a serial chain or the whole parallel mechanism, the twist system, which represents the mobility, and the wrench system, which represents the constraints, are reciprocal to each other and $T = W^{\perp}$ and $W = T^{\perp}$. Reciprocal screw systems also follow a form of De Morgan's law in the following way

$$\begin{aligned} \left(\sum_i S_i\right)^{\perp} &= \bigcap_i S_i^{\perp} \\ \left(\bigcap_i S_i\right)^{\perp} &= \sum_i S_i^{\perp} \end{aligned} \tag{2.14}$$

Combining Equation 2.10 to 2.13, a formulation is obtained for the mobilities and the constraints between the end-effector and the base of a parallel mechanism in which the legs are formed by serial chains.

$$\begin{aligned} T &= \bigcap_{i=1}^{p} \left(\sum_{j=1}^{q} T_{ij}\right) \\ W &= \sum_{i=1}^{p} \left(\bigcap_{j=1}^{q} W_{ij}\right) \end{aligned} \tag{2.15}$$

The two equations in 2.15 are reciprocal, and $W = T^{\perp}$. The set of screws that belongs to the screw system T , which is formed generally by an infinite number of screws, describe all the possible instantaneous motion between the base and the end-effector. The number of DOF, corresponding to the term M in equation 2.5 is simply given by $M = dim\,(T)$, i.e. the maximum number of independent screws in T.

It is usually simpler to perform sum and intersection operations on screw systems of smaller dimension than on larger ones. Operations on small screw systems are simpler and already well known for several special cases while operations on screw systems of dimension greater than 3 are known to be more tedious. Reciprocal screw systems are also well defined for several known screw systems. The wrench space of a single joint is generally of higher dimension than its twist space. On the other hand, in most parallel mechanisms, the twist space of a serial chain is of higher dimension that its wrench space. Because of all those reasons, kinematicians usually prefer to combine Equation 2.14 and 2.15 and use the following equation to calculate the mobility of parallel robots.

$$T = W^{\perp} = \left(\sum_{i=1}^{p} W_i\right)^{\perp} = \left(\sum_{i=1}^{p} \left(T_i^{\perp}\right)\right)^{\perp} = \left(\sum_{i=1}^{p} \left(\sum_{j=1}^{q} T_{ij}\right)^{\perp}\right)^{\perp} \tag{2.16}$$

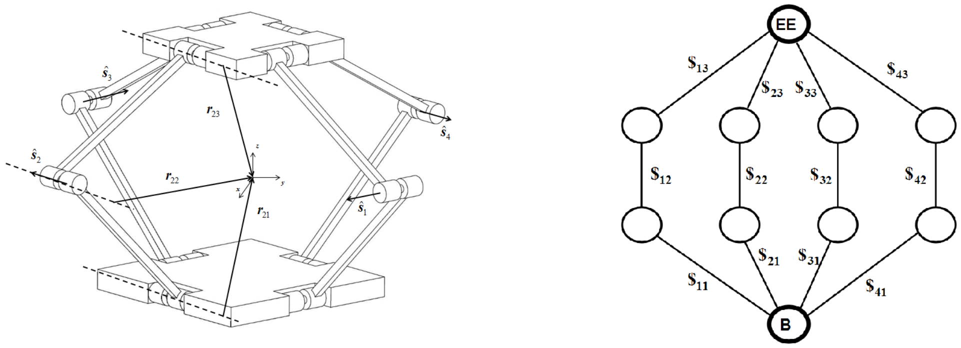
Another advantage of using Equation 2.16 instead of the two equations in 2.15 is the possibility to calculate in one process both the local mobilities and the overconstraints, referred as R_M and R_C respectively in Equation 2.5. The number of local mobility in each leg is the dimension of the redundancy that occurs in the sum of the joint twist systems. The number of overconstraints is the dimension of the redundancy that occurs in the sum of the leg wrench systems. Using the same notation, their values are given by

$$\begin{aligned} R_M &= \textstyle\sum_{i=1}^{p} \left(\sum_{j=1}^{q} dim\,(T_{ij}) - dim\,(T_i)\right) \\ R_C &= \textstyle\sum_{i=1}^{p} dim\,(W_i) - dim\,(W) \end{aligned} \tag{2.17}$$

2.2.4 An Example

In this section, the standard mobility method is applied to a 4-RRR parallel mechanism like the one used in [11]. The mathematical analysis differs a little and is shorter than [11] which focused on dividing the constraints into mechanism constraints and end-effector constraints. The mechanism constraints are the constraints common to all the legs, and its reciprocal screw system forms the d dimensional mechanism motion system. If $d < 6$, the mechanism can be considered a special case of general spatial mechanisms, but it's always possible to perform the analysis using the general spatial $d = 6$ framework. A $d = 3$ framework is often used to simplify the analysis of planar mechanisms such as in [5], which uses 3-D planar screw systems. In theory, one should be aware that closed-loop $d = 3$ planar mechanisms, like any $d < 6$ mechanisms are always overconstrained from a $d = 6$ analysis point of view, and that a full analysis in $d = 6$ will reveal those overconstraints. In practice, this means that manufacturing tolerances should always be taken into account for the assembly of planar closed-loop mechanisms. In this chapter, it is always considered that $d = 6$ without loss of generality.

The analysis of the 4-RRR is shown here to illustrate the standard mobility analysis method, but it will also serve as a basis to illustrate further concepts developed in this chapter.



a) b)

Figure 2.4: a) Schematic representation and b) graph representation of the 4-RRR pure parallel mechanism. Each vertex represents a rigid link and each edge represents a revolute joint with the twist system $\$_{ij}$.

Figure 2.4 shows a schematic and a graph representation of the 4-RRR mechanism. It is possible to represent the topology of a mechanism with a graph in which each vertex represents a link and each edge represents a joint. Each of the four legs of the mechanism has 3 coplanar revolute joints and each leg is oriented perpendicularly to its neighbor legs. The screw $\$_{ij}$ represents the twist of the revolute joint j of leg i and is given by

$$\$_{ij} = \begin{bmatrix} \hat{\mathbf{s}}_i \\ \hat{\mathbf{s}}_i \times \mathbf{r}_{ij} \end{bmatrix} \tag{2.18}$$

where $\hat{\mathbf{s}}_i$ is a unit vector parallel to the axis of rotation of the revolute joints of leg i and $\mathbf{r}_{ij}$ is a vector from any point on the axis of rotation of the joint j of leg i to the origin of the coordinate frame. The first step of equation 2.16 is used to calculate the twist system ${}^iT$ of leg i. Each leg is formed by three parallel joints all oriented along the axis $\hat{\mathbf{s}}_i$ and positioned at $\mathbf{r}_{i1}$, $\mathbf{r}_{i2}$ and $\mathbf{r}_{i3}$ respectively.

$$\begin{gathered} T_i = \textstyle\sum_{j=1}^{3} \$_{ij} = \left\{ \begin{bmatrix} \hat{\mathbf{s}}_i \\ \hat{\mathbf{s}}_i \times \mathbf{r}_{i1} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_i \\ \hat{\mathbf{s}}_i \times \mathbf{r}_{i2} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_i \\ \hat{\mathbf{s}}_i \times \mathbf{r}_{i3} \end{bmatrix} \right\} \\ T_i = \left\{ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_i \times (\mathbf{r}_{i2} - \mathbf{r}_{i1}) \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_i \times (\mathbf{r}_{i3} - \mathbf{r}_{i2}) \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_i \\ \hat{\mathbf{s}}_i \times \mathbf{r}_{i3} \end{bmatrix} \right\} \end{gathered} \tag{2.19}$$

Where the independent screws within the brackets represent a non unique basis of the screw system T_i. The first set of basis screws correspond directly to the joint screws while the second set is a more convenient set obtained by linear combinations of the basis screws. Each leg has 3 DOF that can be interpreted as two linear translations in the plane perpendicular to the $\hat{\mathbf{s}}_i$ axis and a rotation around $\hat{\mathbf{s}}_i$ located on the last joint. The screw system that is reciprocal to T_i represents the constraints of the serial leg.

$$W_i = T_i^{\perp} = \left\{ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_i \times (\mathbf{r}_{i2} - \mathbf{r}_{i1}) \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_i \times (\mathbf{r}_{i3} - \mathbf{r}_{i2}) \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_i \\ \hat{\mathbf{s}}_i \times \mathbf{r}_{i3} \end{bmatrix} \right\} \tag{2.20}$$

where each screw in W_i is reciprocal to each screw in T_i such that their reciprocal product as defined in equation 2.8 is equal to zero. In the current case, the screw system is self-reciprocal. The constraints of the leg can be interpreted as two couples in the plane perpendicular to $\hat{\mathbf{s}}_i$ and a linear force in the $\hat{\mathbf{s}}_i$ direction at the third joint location.

The total constraints applied on the end-effector are given by the sum of the constraints of each leg. The four constraint systems are

$$\begin{aligned}
{}^1W &= \left\{ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_1 \times (\mathbf{r}_{12}-\mathbf{r}_{11}) \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_1 \times (\mathbf{r}_{13}-\mathbf{r}_{12}) \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_1 \\ \hat{\mathbf{s}}_1 \times \mathbf{r}_{13} \end{bmatrix} \right\} \\
{}^2W &= \left\{ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_2 \times (\mathbf{r}_{22}-\mathbf{r}_{21}) \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_2 \times (\mathbf{r}_{23}-\mathbf{r}_{22}) \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_2 \\ \hat{\mathbf{s}}_2 \times \mathbf{r}_{23} \end{bmatrix} \right\} \\
{}^3W &= \left\{ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_3 \times (\mathbf{r}_{32}-\mathbf{r}_{21}) \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_3 \times (\mathbf{r}_{33}-\mathbf{r}_{22}) \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_3 \\ \hat{\mathbf{s}}_3 \times \mathbf{r}_{13} \end{bmatrix} \right\} \\
{}^4W &= \left\{ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_4 \times (\mathbf{r}_{42}-\mathbf{r}_{41}) \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_4 \times (\mathbf{r}_{43}-\mathbf{r}_{42}) \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_4 \\ \hat{\mathbf{s}}_2 \times \mathbf{r}_{23} \end{bmatrix} \right\}
\end{aligned} \tag{2.21}$$

A set of independent screws that represents the 12 constraints of equation 2.21 must be calculated. For opposite legs $i = 1, 3$ and $i = 2, 4$, the axis $\hat{\mathbf{s}}_i$ are parallel and the second parts of the basis screws span the same plane perpendicular to $\hat{\mathbf{s}}_i$. It can be concluded that all constraints in leg 3 and 4 are redundant. Among the 6 screws of leg 1 and 2, four of them are pure moment constraints, so only three can be independent. Finally, the wrench system of the end-effector can be expressed as

$$\begin{aligned}
W = \sum_{i=1}^{4} W_i \quad = \quad & \left\{ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_1 \times (\mathbf{r}_{12}-\mathbf{r}_{11}) \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_1 \times (\mathbf{r}_{13}-\mathbf{r}_{12}) \end{bmatrix}, \right. \\
& \left. \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_2 \times (\mathbf{r}_{22}-\mathbf{r}_{21}) \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_1 \\ \hat{\mathbf{s}}_1 \times \mathbf{r}_{13} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_2 \\ \hat{\mathbf{s}}_2 \times \mathbf{r}_{23} \end{bmatrix} \right\}
\end{aligned} \tag{2.22}$$

The $\hat{\mathbf{s}}_1$ and $\hat{\mathbf{s}}_2$ axis are orthogonal. The only screw which is reciprocal to the 5 screws of W gives the twist system of the end-effector.

$$T = W^{\perp} = \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_1 \times \hat{\mathbf{s}}_2 \end{bmatrix} \tag{2.23}$$

The end-effector can only perform linear translations perpendicular to the revolute axis of the base. Referring to equation 2.5, the degree of mobility of the end-effector is $M = dim(T) = 1$, the number of overconstraints is given by $R_C = \sum_{i=1}^{4} dim\left({}^iW\right) - dim\left(W\right) = 12 - 5 = 7$, and the number of local mobility R_M is zero since ${}^iR_M = \sum_{j=1}^{3} dim\left({}^i\$_j\right) - dim\left({}^iT\right) = 0$ for each leg. The mechanism has $n = 10$ links and $m = 12$ single DOF joints. This can be verifies with Equation 2.5.

$$\begin{gathered}
M = 6\left(n - m - 1\right) + \textstyle\sum_{i=1}^{m}\left(f_i\right) + R_C - R_M \\
1 = -18 + 12 + 7 - 0
\end{gathered} \tag{2.24}$$

2.2.5 Analytical VS Numerical Method for the Calculation of the Mobility

The mobility analysis presented in Section 2.2.4 uses an analytical representation of vectors. It can be noted that no preferred reference frame was mentioned during the process in order to show that the method is independent of the choice of any particular reference frame. In practice however, the choice of a judicious coordinate frame can highly facilitate the analysis without compromising the generality of the results. Some geometric relations between the vectors were assumed in order to calculate dependencies and reciprocity of the screw systems. In this analysis for example, it was assumed that all joints in a leg were parallel and that the legs were orthogonal. The method can of course be applied directly with numerical values for a given configuration. Although often simpler to perform and technically possible to automate with an algorithm, there are several disadvantages of using numerical values instead of vector representations to calculate the mobility of mechanisms.

The first one concerns the numerical errors that can occur in divisions by the use of floating-point numerical values. On two occasions, the method requires the computation of the number of independent screws when performing the sum of screws systems. In other words, it required the computation of the rank of matrices. This can be only performed correctly if the numerical values don't contain numerical errors. Screw systems of lower dimensions are relatively well known analytically and have been widely studied. Given an analytical expression of a screw system and considering some geometric vector relations, it is generally easy to determine the dimension of the screws system and its reciprocal.

A second point is that analytical formulations are not configuration dependent and give better insights in the nature and the inter-relations of the constraints and mobilities in the system. For example, using vectors to represent the mobility of a serial chain can help to determine if the twist system obtained is valid for the full-cycle mobility of the chain or will change according to the chain configuration.

Finally, vector representation may allows the determination of the singularities of the mechanism by inspection. Singularities occur when the rank of a screw system changes in a certain configuration. Serial singularities occurs when the dimension of the sum of twist systems is reduced in a certain configuration while constraint singularities [83] occurs when the dimension of the sum of wrench systems is reduced in a certain configurations. As opposed to parallel robot singularities [26] which depend on input-output relations from the actuators to end-effector, these singularities don't depend on the actuators positions and are a fundamental property of the mechanism itself. Vector representation can help to establish for which conditions and configurations these singularities occur while it is impossible to do with numerical method. When the mobility is obtained from numerical methods, one should repeat the process again with a different configuration to insure that the mechanism wasn't already in a singular configuration when the mobility was first calculated.

2.3 Solutions to Screw Systems under Linear Homogeneous Equations

In the standard mobility analysis method presented in Section 2.2, the number of the local mobilities R_M and the number of overconstraints R_C were calculated as the number of redundant basic screws in the sum of screw systems. Given p screw systems S_i with $i = 1..p$, the redundancy R is given by

$$R = \sum_{i=1}^{p} dim\,(S_i) - dim\left(\sum_{i=1}^{p} S_i\right) \tag{2.25}$$

which is the total dimensions of the screw systems minus the dimension of the sum of the screw systems. The number of local mobilities and the number of overconstraints can both be calculated with Equation 2.25. However the equation only tells the number of redundancies and doesn't tell us how these redundancies are organized into screw systems and how they are distributed on the total screw system $S = \sum_{i=1}^{p} S_i$. Local mobilities occur if some joints can have a non-zero velocity while the end links of the chain is fully constraint. In such a case, the sum of the twist screws associated with the joint velocities must be zero. When overconstraints occur, internal stresses are introduced in the joints when the mechanism is assembled. If no external load is applied on the mechanism, the sum of the internal stresses on one link must be zero. In both cases, the set of screws describing the local mobilities and the overconstraints must satisfy a homogenous system of 6 equations. In this section, a framework that allows us to describe analytically the screw systems of these redundancies is presented.

A screw system S can be represented by a set of independent basis unit screws $\hat{\$}_i$. The number of basis screws n corresponds to the dimension of the screw system and any particular screw $\$$ within the system can be obtained by a linear combination of the basis screws.

$$S = \left\{\hat{\$}_1, \cdots, \hat{\$}_i, \cdots, \hat{\$}_n\right\} \tag{2.26}$$

$$\$ = \sum_{i=1}^{n} \lambda_i \hat{\$}_i \tag{2.27}$$

The screws $\hat{\$}_i$ are said to span the screw system S. The set of basis screws is not unique and different sets can span the same screw system. Given a set of basis screws, each screw $\$$ within S is defined by a unique set of magnitudes λ_i since the basis screws $\hat{\$}_i$ are independent. The sum of p screw systems represents all the screws that can be obtained from linear combinations of the p set of basis screws. The sum of the p screw systems S_i with $i = 1..p$. is noted S

$$S = \sum_{i=1}^{p} S_i \tag{2.28}$$

where $S_i = \left\{\hat{\$}_{i1}, \cdots, \hat{\$}_{ij}, \cdots, \hat{\$}_{in_i}\right\}$. Any particular screw $\$$ within S can be obtained from linear combinations of the unit basis screws $\hat{\$}_{ij}$.

$$\$ = \sum_{i=1}^{p} \sum_{j=1}^{n_i} \lambda_{ij} \hat{\$}_{ij} \tag{2.29}$$

If the screw systems S_i have some intersections, i.e the unit screws $\hat{\$}_{ij}$ are not independent, an infinite number of set of magnitude λ_{ij} can be used to obtain \$. In practice, when performing the sum of p screw systems, the inputs are the p sets of basis screws and the output is a single set of independent basis screws that span the resulting sum. If the screw systems don't intersects, i.e $S_i \cap \left(\sum_{k \neq i} S_k\right) = \{0\}$, the basis screws of the sum S are simply all the basis screws $\hat{\$}_{ij}$ of the p screw systems.

As mentioned, local mobilities and overconstraints are represented by the set of screws that are restricted by a homogeneous system of linear equations. The next section presents a general procedure to calculate these redundancies.

2.3.1 Restricted Screw System

Lets S_i be a screw system that can be described by a set of q_i independent basis unit screws $\hat{\$}_{ij}$ noted $S_i = \left\{\hat{\$}_{i1}, \cdots, \hat{\$}_{ij}, \cdots, \hat{\$}_{iq_i}\right\}$. Any screw $\$_i$ in S_i can be described by a linear combination of the basis screws. Six linear homogenous equations now restrict the p screw systems such that the sum of the screws $\$_{ij} = \hat{\$}_{ij}\lambda_{ij}$ is zero. The set of restricted screws $\$_{ij}$ are the unit screws $\hat{\$}_{ij}$ and magnitudes λ_{ij} that satisfy the following relations

$$\sum_{i=1}^{p}\sum_{j=1}^{q_i} \hat{\$}_{ij}\lambda_{ij} = \mathbf{0}_{6,1} \tag{2.30}$$

Putting the last screw system on the right side, Equation 2.30 can be rewritten the following way

$$\sum_{i=1}^{p-1}\sum_{j=1}^{q_i} \hat{\$}_{ij}\lambda_{ij} = -\sum_{j=p}^{q_p} \hat{\$}_{pj}\lambda_{pj} \tag{2.31}$$

The complete space of solutions of this system of equations, like any non-homogeneous system of linear equations, can be represented by the addition of the subspace of particular solutions and the subspace of homogeneous solutions. The particular solutions S_p^R form a screw system given by

$$S_p^R = S_p \bigcap \left(\sum_{i=1}^{p-1} S_i\right) \tag{2.32}$$

which is spanned by a set of independent basis screws $\$_{pk}^R$. The set of homogeneous solutions must satisfy the following relation

$$\sum_{i=1}^{p-1}\sum_{j=1}^{q_i} \hat{\$}_{ij}\lambda_{ij} = \mathbf{0}_{6,1} \tag{2.33}$$

The last screw system is again put on the rigth side to obtain

$$\sum_{i=1}^{p-2}\sum_{j=1}^{q_i}\hat{\$}_{ij}\lambda_{ij} = -\sum_{j=(p-1)}^{q_p}\hat{\$}_{(p-1)j}\lambda_{(p-1)j} \tag{2.34}$$

which is the same as Equation 2.30 except that the $i = p$ screw system has been removed. A particular solution S^R_{p-1} to Equation 2.33 can be found again as

$$S^R_{p-1} = S_{p-1}\bigcap\left(\sum_{i=1}^{p-2} S_i\right) \tag{2.35}$$

By applying recursively this relation until $i = 2$, a set of $p-1$ screw systems is obtained in which S^R_i represents all the independent solutions of the sum of screw systems under homogeneous linear equations. The restricted screw system S^R_i form a vector subspace of S_i given by

$$S^R_i = S_i\bigcap\left(\sum_{j=1}^{i-1} S_j\right) \tag{2.36}$$

It will now be proven that the sum of the dimensions of the restricted screw systems S^R_i is equal to the redundancy of the system, and that this result is independent of the order in which the screw systems are labelled in Equation 2.36. The total of the dimension of the restricted screw systems R is given by

$$R = \sum_{i=2}^{p} dim\left(S^R_i\right) \tag{2.37}$$

$$R = \sum_{i=2}^{p} dim\left(S_i\bigcap\left(\sum_{j=1}^{i-1} S_j\right)\right) \tag{2.38}$$

The dimension of the intersection of two screw systems S_1 and S_2 is given by

$$dim\left(S_1\bigcap S_2\right) = dim\left(S_1\right) + dim\left(S_2\right) - dim\left(S_1 + S_2\right) \tag{2.39}$$

Applying this result to 2.37 gives

$$R = \sum_{i=2}^{p}\left(dim\left(S_i\right) + dim\left(\sum_{j=1}^{i-1} S_j\right) - dim\left(\sum_{j=1}^{i} S_j\right)\right) \tag{2.40}$$

$$R = \sum_{i=2}^{p} dim\left(S_i\right) + dim\left(S_1\right) - dim\left(\sum_{j=1}^{p} S_j\right) \tag{2.41}$$

$$R = \sum_{i=1}^{p} dim\left(S_i\right) - dim\left(\sum_{i=1}^{p} S_i\right) \tag{2.42}$$

which is the same result as Equation 2.25. It can be seen that in both terms of equation 2.42 the order in which the screw systems are labelled is not important. By definition, $S_1^R = \{0\}$, and a different labelling order will produce different solutions for the restricted screw system $S_i^R \subseteq S_i$ defined by Equation 2.36, but the total dimension will be the same and correspond to the redundancy of the system. .

2.3.2 Unrestricted Screw Systems and Distribution

In Section 2.3.1, a number of restricted screw systems $S_i^R \subseteq S_i$ were defined, to which the dimensions correspond to the redundancy in the sum of screw systems. The restricted screw systems S_i^R are formed by all the screws that are solution to the right part of the following equation

$$\sum_{k=1}^{i-1}\sum_{j=1}^{q_k} \hat{\$}_{kj}\lambda_{kj} = -\sum_{j=1}^{q_i} \hat{\$}_{ij}\lambda_{ij} \tag{2.43}$$

The restricted screw system is included in the original screw system $S_i^R \subseteq S_i$, and can be expressed with a set of basis screws $S_i^R = \left\{\left[\hat{\$}_1^R, \cdots, \hat{\$}_{q_i^R}^R\right]\right\}$, where q_i^R is the dimension of the restricted screw system. An independent variable λ_{ij}^R can be assigned to each redundancy and Equation 2.43 can be expressed as

$$\sum_{k=1}^{i-1}\sum_{j=1}^{q_k} \hat{\$}_{kj}\lambda_{kj} = \sum_{j=1}^{q_i^R} \hat{\$}_{ij}^R\lambda_{ij}^R \tag{2.44}$$

The solutions for the left part represent the distribution of the redundancy expressed by the variables λ_{ij}^R on the remaining $k = 1..i-1$ screw systems. If the screw systems S_k intersect, there are multiple ways to distribute S_i^R on the sum of the S_k screw systems. Any solution can be chosen and will represent the redundancies correctly, and the different solutions lead to different physical interpretations which are equivalent. A convenient way to distribute the redundancy is the use of the unrestricted screw system. For a restricted screw system $S_i^R \subseteq S_i$ that has been defined based on the intersection of S_i with the previous screw systems, it is possible to find a complement unrestricted screw systems S_i^U that satisfies the following equation

$$S_i^R \oplus S_i^U = S_i \tag{2.45}$$

where the operator $\oplus$ is the direct sum. A direct sum implies that the two vector subspaces S_i^R and S_i^U do not intersect, so $S_i^R \bigcap S_i^U = \{0\}$. The complement screw system is not unique and the choices of unrestrited screw systems will lead to equally valid but different physical interpretations. A complementary screw system can be expressed with a set of basis screws $S_i^U = \left\{\$_i^U, \cdots, \$_{q_i^U}^U\right\}$, where $q_i^U = dim\left(S_i^U\right)$ and $q_i^R + q_i^U = q_i$. A specific solution of Equation 2.44 can now be expressed as

$$\sum_{k=1}^{i-1}\sum_{j=1}^{q_k^C} \hat{\$}_{kj}^U\lambda_{kj}^{Ui} = \sum_{j=1}^{q_i^R} \hat{\$}_{ij}^R\lambda_{ij}^R \tag{2.46}$$

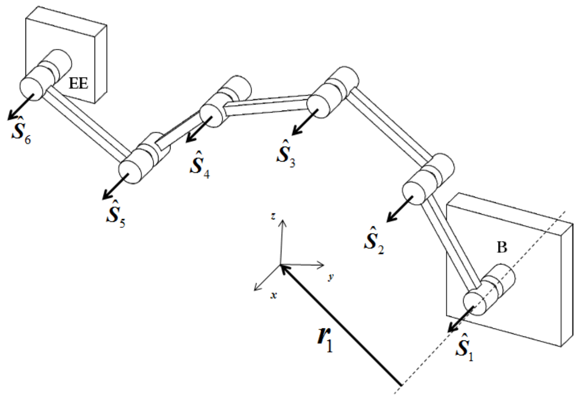


Figure 2.5: A planar 6R serial chain in which each the mobility of each joint is represented by a zero-pitch screw.

Given a set of values for λ_{ij}^R, this equation has an unique solution in the space of values for λ_{kj}^{Ui}.

This formulation can be applied to the distribution of local mobilities and the distribution of overconstraints in a mechanism. In case of local mobilities, the velocity of joint i can be expressed as

$$T_i = \sum_{j=1}^{q_i^R} \hat{\$}_{ij}^R \lambda_{ij}^R + \sum_{k=i+1}^{p} \sum_{j=1}^{q_k^U} \hat{\$}_{ij}^U \lambda_{kj}^{Ui} \tag{2.47}$$

where λ_{kj}^{Ui} is a linear function of λ_{ij}^R described by Equation 2.46. The distribution of overconstraints can be represented in the same way. The next two sections show an example of local mobilities distribution and an example of overconstraints distribution using this method.

2.3.3 Example 1: Distribution of Local Mobilities in a 6R Planar Chain.

In this section, the method proposed for the distribution of redundancy in sum of screw systems is used to define analytically the local mobilities in a serial chain. Figure 2.5 shows a planar serial chain formed by six revolute joints with parallel axis. Joints are labeled $i = 1..6$ from the base to the end-effector. Each joint has 1 DOF and the same rotation vectors $\mathbf{s}$. $\mathbf{r}_i$ is any vector from the axis i to the origin of the reference frame. The screw system of joint i is given by

$$S_i = \begin{bmatrix} \hat{\mathbf{s}} \\ \hat{\mathbf{s}} \times \mathbf{r}_i \end{bmatrix} \tag{2.48}$$

The method can be applied directly to calculate the global and local mobility of the chain. For each joint, the restricted screw system is calculated using Equation 2.36 and an unrestricted screw system is calculated using Equation 2.45.

$$
\begin{array}{ll}
S_1^R = \{0\} & S_1^U = \begin{bmatrix} \hat{\mathbf{s}} \\ \hat{\mathbf{s}} \times \mathbf{r}_1 \end{bmatrix} \\
S_2^R = S_2 \bigcap S_1^U = \{0\} & S_2^U = \begin{bmatrix} \hat{\mathbf{s}} \\ \hat{\mathbf{s}} \times \mathbf{r}_2 \end{bmatrix} \\
S_3^R = S_3 \bigcap \left(S_1^U \bigoplus S_2^U\right) = \{0\} & S_3^U = \begin{bmatrix} \hat{\mathbf{s}} \\ \hat{\mathbf{s}} \times \mathbf{r}_3 \end{bmatrix} \\
S_4^R = S_4 \bigcap \left(\bigoplus_{i=1}^3 S_i^U\right) = \begin{bmatrix} \hat{\mathbf{s}} \\ \hat{\mathbf{s}} \times \mathbf{r}_4 \end{bmatrix} & S_4^U = \{0\} \\
S_5^R = S_5 \bigcap \left(\bigoplus_{i=1}^4 S_i^U\right) = \begin{bmatrix} \hat{\mathbf{s}} \\ \hat{\mathbf{s}} \times \mathbf{r}_5 \end{bmatrix} & S_5^U = \{0\} \\
S_6^R = S_6 \bigcap \left(\bigoplus_{i=1}^5 S_i^U\right) = \begin{bmatrix} \hat{\mathbf{s}} \\ \hat{\mathbf{s}} \times \mathbf{r}_6 \end{bmatrix} & S_6^U = \{0\}
\end{array}
\tag{2.49}
$$

The method leads to a direct physical interpretation of the mobilities of the chain. The mobilities are distributed among 4 screws systems; the 3-dimensional unrestricted screw system $S = \left\{S_1^U, S_2^U, S_3^U\right\}$ and the three restricted screw systems S_4^R, S_5^R, and S_6^R. Mobility variables can be assigned to each systems and can be distributed in a unique way on the unrestricted screw system S. Let us define the 3 mobilities of S as λ_1, λ_2, λ_3 and the three local mobilities of S_i^R as λ_4^R, λ_5^R,λ_6^R. There is always a unique way in which S_i^R projects on S such that

$$
\lambda_i^R S_i^R = -\rho_{1i} S_1^U - \rho_{2i} S_2^U - \rho_{3i} S_3^U \tag{2.50}
$$

The velocity of each joint $\dot{\theta}_i$ can be now be expressed as function of the global and local mobility variables

$$
\begin{bmatrix} \lambda_1^U \\ \lambda_2^U \\ \lambda_3^U \end{bmatrix} = \begin{bmatrix} -\rho_{14} & -\rho_{15} & -\rho_{16} \\ -\rho_{24} & -\rho_{25} & -\rho_{26} \\ -\rho_{34} & -\rho_{35} & -\rho_{36} \end{bmatrix} \begin{bmatrix} \lambda_4^R \\ \lambda_5^R \\ \lambda_6^R \end{bmatrix} \tag{2.51}
$$

In this case, $\dot{\theta}_i = \lambda_i^R$ and the velocity of the first three joints can be expressed as the function of the velocity of the last three joints when the end-effector is locked in position. The method can be applied to any type of serial chain. The division of each joint screw system into restricted and unrestricted screw systems lead to a distinction between the global mobility variables and the local mobility variables. In the next section, the exact same method is applied to find and distribute the constraints and overconstraints in a pure parallel mechanism.

2.3.4 Example 2: Distribution of the overconstraints in a 4-RRR Parallel Mechanism

The same 4-RRR mechanism that was used in Section 2.2.4 is used in this section to illustrate the calculation of the overconstraints with the proposed method. In the previous

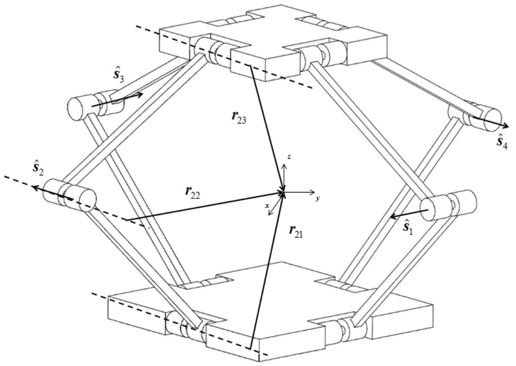


Figure 2.6: The 4-RRR parallel mechanism. Each joint of leg i is oriented in the $\hat{\mathbf{s}}_i$ direction and the position of the joint axis j is given by the vector $\mathbf{r}_{ij}$.

section, the constraints W between the base and the end-effector were calculated as $W = \sum_{i=1}^{4} W_i$ and the number of overconstraints was given by $R_C = \sum_1^4 dim\,(W_i) - dim\,(W)$. Little details were given about how to find a valid basis for W and it was not mentioned how to distribute the overconstraints in the mechanism. The general method presented in this section will be used to achieve this goal. The vectors used to represents the wrench systems of the mechanism are shown in Figure 2.6 .

The wrench system W_i of each leg was described by Equation 2.21. As mentioned in Section 2.2.5, the calculation of screw system intersections is often facilitated by the use of a appropriated coordinate system. Using a coordinate frame where $\mathbf{x} = \mathbf{s_1}$ and $\mathbf{y} = \mathbf{s_2}$, the 4 wrench systems can be expressed as

$$\begin{aligned}
W_1 &= \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{x} \\ \mathbf{0} \end{bmatrix} \right\} \\
W_2 &= \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{y} \\ \mathbf{0} \end{bmatrix} \right\} \\
W_3 &= \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{x} \\ \mathbf{0} \end{bmatrix} \right\} \\
W_4 &= \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{y} \\ \mathbf{0} \end{bmatrix} \right\}
\end{aligned} \tag{2.52}$$

The exact same procedure as the previous example is used to find the restricted and unrestricted wrench systems.

$$
\begin{aligned}
&W_1^R = \{0\} && W_1^U = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{x} \\ \mathbf{0} \end{bmatrix} \right\} \\
&W_2^R = W_2 \bigcap W_1^U = \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} && W_2^U = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix}, \begin{bmatrix} \mathbf{y} \\ \mathbf{0} \end{bmatrix} \right\} \\
&W_3^R = W_3 \bigcap \left(W_1^U \bigoplus W_2^U\right) = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{x} \\ \mathbf{0} \end{bmatrix} \right\} && W_3^U = \{0\} \\
&W_4^R = W_4 \bigcap \left(\bigoplus_{i=1}^3 W_i^U\right) = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{y} \\ \mathbf{0} \end{bmatrix} \right\} && W_4^U = \{0\}
\end{aligned}
\tag{2.53}
$$

From this analysis, it is clear that the unrestricted wrench system $W = W_1^U \oplus W_2^U$ is of dimension 5. The three restricted wrench systems W_1^R, W_2^R, and W_3^R are of dimension one, three, and three respectively, for a total of seven overconstraints. Each restricted wrench system W^R is expressed with a set of basis screws $W_i^R = \{\$_{i1}^R, \cdots, \$_{ij}^R, \cdots \$_{iq_i}^R\}$ where q_i is the dimension of the restricted screw system. Each basis screw of the restricted wrench system can be projected in a unique way on an unrestricted screw system $W = W_1^U \oplus W_2^U$ such that

$$
\lambda_{ij}^R \$_{ij}^R = -\sum_{j=1}^{3} \rho_{1j} \$_{1j}^U - \sum_{j=1}^{2} \rho_{2j} \$_{2j}^U
\tag{2.54}
$$

The seven independent internal stresses, corresponding to the screws $\lambda_{ij}^R \$_{ij}^R$ of the wrench systems W_2^R, W_3^R and W_4^R are distributed on W_1^U and W_2^U in the following way:

$$
\begin{bmatrix} \lambda_{11}^U \\ \lambda_{12}^U \\ \lambda_{13}^U \\ \lambda_{21}^U \\ \lambda_{21}^U \end{bmatrix}
=
\begin{bmatrix}
0 & -1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & -1 & 0 & 0 & -1 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1
\end{bmatrix}
\begin{bmatrix} \lambda_{21}^R \\ \lambda_{31}^R \\ \lambda_{32}^R \\ \lambda_{33}^R \\ \lambda_{41}^R \\ \lambda_{42}^R \\ \lambda_{43}^R \end{bmatrix}
\tag{2.55}
$$

where $\lambda_{ij}\$_{ij}$ is the internal stress on the wrench screw j of leg i. If the seven independent internal stresses $\lambda_{ij}^R \$_{ij}^R$ in the attached joint between the legs and the end-effector are measured, the matrix gives the internal stresses in the remaining five constraints so that the end-effector is in static equilibrium. Each leg must excert an opposite wrench at its end joints in order to stay in equilibrium and therefore the internal stresses in the base link are symmetrical to the internal stresses in the end-effector. The internal stresses within a leg is transmitted by each joint of the leg. This method can be applied to any pure parallel mechanism to calculate the number of independent overconstraints and to calculate the distribution of the internal stresses caused by these overconstraints over all joints and links of the mechanism.

2.4 Discussion and Summary

The first part of this chapter presented a general method based on screw theory for the mobility analysis of overconstrained pure parallel mechanisms that can be considered standard. The method is not based on one article in particular but instead uses elements and techniques based on the work of Hunt [37], Mohamed [55], and Phillips [65, 64] and uses the rules of twist addition for serial connections and twist intersection for parallel connections. More recently Gogu [22] discussed the concept of local mobilities and Dai [11] discussed the types of constraints and overconstraints in parallel mechanisms and made the distinction between the mechanism constraints system and the end-effector constraints system. Little was explained about how the local mobilities are distributed on the joints velocities or how the overconstraints are distributed as internal stress in the joints of the mechanisms.

The concepts of local mobilities and overconstraints are both related to sum of screw systems that must satisfy a set of homogeneous linear equations. In the second part of this chapter the concept of restricted screw systems was introduced and a framework for their calculation was presented. For each screw system that is included in the homogeneous equations, the procedure creates two subspaces, namely the restricted and unrestricted screw systems. In mobility analysis, the unrestricted screw systems represent the mobility of the end link in a serial chain and the restricted screw system represents the local mobilities. In constraints analysis, the unrestricted screw systems represent the constraints between the base and the end-effector and the restricted screw systems represent the overconstraints of the mechanism.

The method uses the vector space operations of intersection, complementary subspace and parallel projection of screw systems. These operations are often well known for usual screw systems, which facilitate the use of configuration-free analytical vector calculations which are valid for the full-cycle workspace instead of conguration-dependent numerical computations which are only valid locally. In addition, analytical vector calculation could be used to detect geometric conditions for serial singularities and constraint singularities.

Chapter 3

Mobility and Overconstraints Analysis of Parallel Mechanisms with Configurable Platforms

3.1 Introduction

In their article on the velocity analysis of interconnected chain mechanisms [85], Zoppi and al define a “pure” parallel mechanism as a parallel mechanism which has two terminal links; the base and the end-effector, connected in parallel by a set of independent serial chains. Several authors [49, 78, 18] have also proposed a similar definition for the term “pure” parallel mechanism although the term is not yet officially recognized by IFToMM. According to this definition, most closed-loop spatial mechanisms presented in the literature are purely parallel such as the well-known Gough Stewart platform[29]. The mobility and overconstraint analysis of any pure parallel mechanism can be computed with the method presented in Section 2. Less research has been done on robotic systems based on mechanisms that are not purely parallel. Among them are the hybrid robots [75] which consist of a number of serially connected parallel robots or the famous Delta robot [10], the legs of which are not formed of pure serial chains but contain a parallelogram mechanism. Mechanisms that are not purely parallel are often ignored by type synthesis methods such as in [42]. Yet, research on alternative classes of mechanisms can bring new possibilities in robot architecture design.

In general, parallel mechanisms with configurable platforms (PMCP) can bring two types of new possibilities for robot designers. On the first hand, they can offer additional kinematic solutions for a given motion requirement. As opposed to serial robots, for parallel robots a large number of different architectures can lead to the same motion pattern. In the first steps of robot design process, several architectures that fit the motion requirements are generally considered and one must be selected based on other requirements such has kinematic dexterity or structural stiffness. PMCPs can use their configurable platforms to generate additional motions of the end-effector and will offer different kinematic properties than pure parallel mechanisms for the same type of motion. On the second hand, as opposed to parallel robots, robots with a PMCP structure can have multiple end-effectors on the configurable platform while all the motors are located on the base. This is not possible with pure parallel robots. These multiple end-effectors can be used as grasping devices for example. From a structural stiffness point of view, it is more desirable to have all motors on the base than attaching a grasping hand at the top of a rigid platform. In the case of the PentaG robot, both of the advantages described above are present since the 2 DOF platform is used to provide both rotation and grasping capabilities to the robot.

Yet, very few researches have been done on parallel mechanisms with configurable platforms. The first example presented in the literature of a parallel robot in which the platform in not rigid (i.e. configurable) can be attributed to Yi and al in 2002[81]. Some specific robot architectures that include a configurable platform have been presented since then in [16, 62, 41] along with their specific kinematic analysis. In 2005, Gosselin and al. [56] highlighted the idea of using configurable platforms in parallel robots and presented a general method for their mobility analysis. However, the method they presented is not applicable for mechanisms that are overconstrained and the overconstraints analysis was not addressed.

In the first part of this chapter, the mobility and overconstraint analysis method that was presented in chapter 2 is extended to a much broader class of mechanisms. The concept of series-parallel mechanisms is introduced and the extended method is generalized to any series-parallel mechanisms. Examples are presented for hybrid mechanism, delta mechanisms and cross-jointing mechanisms. A general method is then proposed to perform the analysis of the mobilities and overconstraints of parallel mechanisms with configurable platforms. One key aspect of the proposed method is based on the concept of restricted and unrestricted screw systems that were used in Section 2.3 to compute the local mobility of serial chains and the overconstraints of pure parallel mechanisms.

A special type of duality between mobility and overconstraints of PMCPs is then introduced. It will be shown that unlike most of other classes of mechanisms, PMCPs have a particular self dual topology that allows the calculations of their mobility and overconstraints to be calculated using the same mathematically method. The general method is then applied to two different PMCPs and an analytical representation and interpretation of the complete distribution of their mobility and overconstraints is presented. Fundamental understanding of the mobility and overconstraints of PMCPs is a key element in the development of a broader type synthesis method for robot architecture that would include such mechanisms.

3.2 Mobility of Series-Parallel Mechanisms

3.2.1 Introduction

In this section, the mobility and overconstraints analysis developed in Chapter 2 will be extended and generalized to a broader class of mechanisms that are not purely parallel. The limitation of this extended method is defined by a single property of the corresponding graph of the mechanism, namely if the graph is series-parallel. Using the concepts of serial and parallel graph reductions, some already known classes of mechanisms that fall into this more general class of mechanisms, such as Delta mechanisms and hybrid mechanisms, are presented within the new general framework. Their formulas for the mobility and the overconstraint calculation are represented in terms of their different layers of serial and

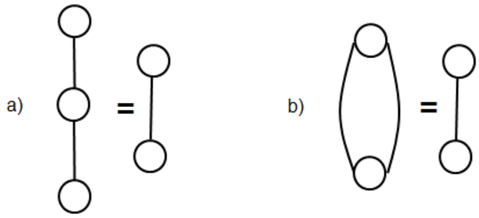


Figure 3.1: a) The serial reduction and b) the parallel reduction in Graph Theory

parallel connections between two terminal links and recursive formulas for the general case of any series-parallel mechanism are presented. Finally, some remarks on cross-jointing mechanisms and non series-parallel graph are mentioned.

3.2.2 Series-Parallel Graphs

The study of the instantaneous kinematics of mechanical chains was pioneered by Waldron [77] when he first introduced the series and parallel laws of instantaneous kinematics. These two rules are still used in the mobility and overconstraints analysis of pure parallel mechanisms presented in Chapter 2. The series and parallel laws for mobility analysis are quite similar to the serial and parallel graph reduction used in Graph Theory. Graphs are mathematical structures used to model pairwise relations between objects. A graph is a set of edges that connect pairs of vertices. They have been used in the study of various fields such as communication networks, chemistry, and electrical networks.

Mechanism architectures can be represented by a graph by assigning a vertex to each link and an edge to each joint. In this case, each edge connects strictly 2 vertices while vertices can be connected to multiple edges. The degree of a vertex is the number of edges that are connected to this vertex. A series-parallel graph is a graph with 2 distinguished vertices called terminal vertices that can be constructed by a sequence of series and parallel compositions. Any series-parallel graph can be reduced to a K_2 graph (a graph with only 2 vertices connected by a single edge) by a sequence of the following operations, shown on Figure 3.1.

- a) Replacement of a vertex of degree 2 and its pair of edges with a single edge.
- b) Replacement of a pair of parallel edges with a single edge that connects their common endpoints.

A series-parallel mechanism could be defined as a mechanism that can be represented with a series-parallel graph. In the standard mobility analysis method presented in section 2.2, the equivalent twist system of a serial leg was computed as the sum of the twist systems of its joints, and the equivalent twist system of parallel legs as the intersection of the leg twist systems. Those systems correspond to the graph reduction operations a) and b) respectively. For a pure parallel mechanism, a twist system is finally obtained between the 2 terminal links, the base and the end-effector as in Equation 2.16. The dimension of the twist system corresponds to the number of DOF between the base and the end-effector. Once the mobility is calculated, a global variable can be assigned to each DOF

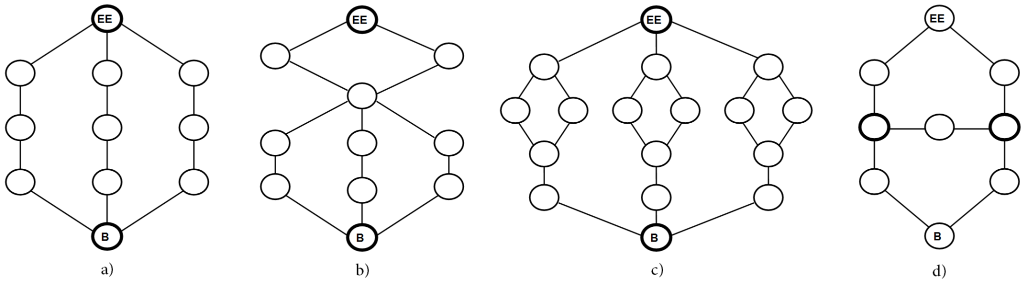


Figure 3.2: Examples of graph representation of a a) pure parallel, b) hybrid, c) Delta, and d) cross-jointing mechanism. B stands for the base, EE stands for the end-effector, and the bold vertices are the terminal links of the graph.

and distributed on all the joints of the mechanism in order to express each joint velocity as function of the end-effector velocity. In this process, the original series-parallel graph is reconstructed with the inverse of operation a) and b). If some legs have local mobilities, defined as R_M in Equation 2.5, those are independent of the mobility of the end-effector and increase the total number of DOF of the mechanism. A more detailed discussion of local mobilities and overconstrains is presented in Section 2.3.3. Using the serial and parallel laws, the method can be extended to mechanisms that are not purely parallel as long as they can be represented by a series-parallel graph. The definition of series-parallel graph ensures that it will be possible to reduce their corresponding graph to two terminal links using only the two reduction laws. In the next sections, general formulas are presented for the mobility analysis of some series-parallel mechanisms that are not purely parallel. Figure 3.2 shows an example for four classes of mechanisms that are series-parallel. In those graphs, B stands for the base, EE stands for the end-effector, and the bold vertices are the terminal vertices of the graph.

3.2.3 Hybrid Mechanisms

Hybrid mechanisms are formed by a serial sequence of parallel mechanisms. They often offer performance which are a compromise between serial mechanisms and pure parallel mechanisms. Figure 3.2 b) shows an example of a graph representation of an hybrid mechanism. From a graph theory point of view, hybrid mechanisms are formed by 3 layers of mechanical connections and their graph can be reduced therefore in a 3-steps sequence of series, then parallel, then series reductions. The first two levels correspond to the standard mobility analysis described by Equation 2.16 for each of the pure parallel mechanisms forming the hybrid mechanisms. The third step is the computation of the mobility of the serial sequence of parallel mechanisms. Given a hybrid mechanism formed by $i = 1..p$ pure parallel mechanisms, each having $j_i = 1..q_i$ parallel legs, each having $k_{ij} = 1..r_{ij}$ joints represented by the twist system T_{ijk}, the mobility between the base and the end-effector is given by

$$T = \sum_{i=1}^{p} \left(\bigcap_{j_i}^{q_i} \left(\sum_{k_{ij}}^{r_{ij}} T_{ijk} \right) \right) \tag{3.1}$$

An equivalent and more convenient computation can be obtained by using reciprocal screw systems as it was done in Equation 2.16.

$$T = \sum_{i=1}^{p} \left(\sum_{j_i=1}^{q} \left(\sum_{k_{ij}=1}^{r} T_{ijk} \right)^{\perp} \right)^{\perp} \tag{3.2}$$

The twist system obtained represents the mobility between the base and the end-effector. These mobilities are independent of local mobilities. As mentioned earlier, sum of twist systems can lead to local mobilities while the sum of wrench systems can lead to overconstraints. Since hybrid mechanisms have 2 layers of serial compositions, two distinct types of local mobilities are possible. The first type is the one encountered in pure parallel mechanisms when a serial leg has more DOF than the mobility between its two end-links. Using the same notation as Equation 3.2, the total number of first type local mobilities R_{M1} is given by

$$R_{M1} = \sum_{i=1}^{p} \sum_{j=1}^{q} \left(\sum_{k_{ij}=1}^{r_{ij}} dim\,(T_{ijk}) - dim\,(T_{ij}) \right) \tag{3.3}$$

The second type of local mobilities represents the redundancy of the serial sequence of parallel mechanism. As it is often the case in hybrid mechanism, the total number of DOF of all parallel mechanisms is larger than the number of DOF between the base and the end-effector. This means that the parallel mechanisms can still move while the end-effector is locked in position. The number of type 2 local mobilities R_{M2} for hybrid mechanisms is given by

$$R_{M2} = \sum_{i=1}^{p} dim\,(T_i) - dim\,(T) \tag{3.4}$$

The total number of local mobilities is then $R_M = R_{M1} + R_{M2}$. Hybrid mechanisms have one layer of parallel compositions and therefore there is one type of overconstraints possible. The number of overconstraints R_C is given by

$$R_C = \sum_{i=1}^{p} \left(\sum_{j=1}^{q} dim\,(W_{ij}) - dim\,(W_i) \right) \tag{3.5}$$

The end-effector mobility is given by $M = dim\,(T)$ while the total mobility of the hybrid mechanism is given by $M + R_{M1} + R_{M2}$. For a mechanism with a total of n links and m one-DOF joints, the mobility Equation 2.5 is still valid and $M = 6\,(n - m - 1) + \sum_{i=1}^{m} (f_i) + R_C - R_{M1} - R_{M2}$.

3.2.4 Delta Mechanisms

Delta robots were introduced as 3 translational DOF robots for the pick-and-place industry. Their high performance has made them widely used for high-speed applications.

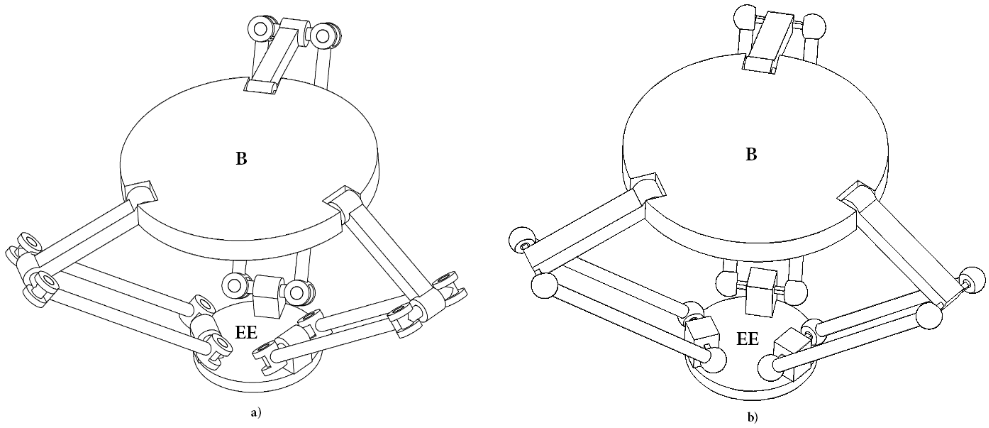


Figure 3.3: 3 DOF Delta mechanisms with a) revolute joints and b) spherical joints

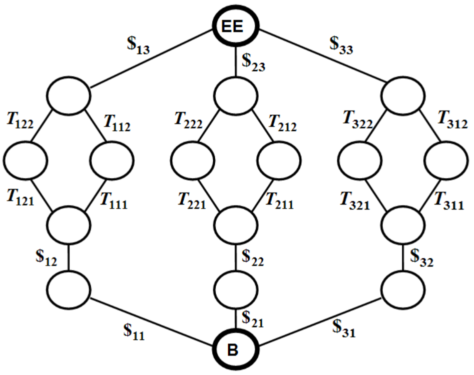


Figure 3.4: The graph representation of a 3 DOF Delta mechanism

One of their characteristics is the use of a parallelogram unit in each leg to constraint the orientation of the end-effector. The parallelogram unit is a four-bar mechanism that can be constructed with either revolute or spherical joints. Figure 3.3 shows a schematic representation of a 3 DOF Delta mechanism using revolute or spherical joints and Figure 3.4 shows a graph representation of such mechanism. When spherical joints are used in the parallelogram units, it is common practice to remove joints $\$_{11}$, $\$_{21}$, $\$_{31}$ and joints $\$_{13}$, $\$_{23}$, $\$_{33}$ as represented in Figure 3.3 b). From a graph theory point of view, due to the parallelogram units, a Delta mechanisms has four layers of serial and parallel compositions. The first two layers concerns the parallelogram units while the last two layers are the same as for the pure parallel mechanisms.

If T_{ikl} is the twist system of the joint $l = 1..2$ of leg $k = 1..2$ of the parallelogram unit i, the mobility of the parallelogram unit T_i is given by

$$T_i = \left(\sum_{k=1}^{2} \left(\sum_{l=1}^{2} T_{ikl} \right)^{\perp} \right)^{\perp} \tag{3.6}$$

For revolute joints, $dim\,(T_i) = 1$ and for spherical joints $dim\,(T_i) = 3$. In practice, the extra mobilities for the use of spherical joints are constrained when the legs of the Delta mechanism are assembled, and the full Delta mechanism has the same end-effector mobility regardless if revolute or spherical joints are used. However, the use of revolute or spherical joints will affect the number and distribution of local mobilities and overconstraints in the mechanism. A classic 3 DOF Delta mechanism has $i = 1..3$ legs and each legs has $j = 1..3$ revolute joints with parallel axis, noted T_{ij}, and one parallelogram unit. Using the four layers of series-parallel composition, the mobility of the end-effector is given by

$$T = \left(\sum_{i=1}^{3} \left(\sum_{j=1}^{3} T_{ij} + \left(\sum_{k=1}^{2} \left(\sum_{l=1}^{2} T_{ikl} \right)^{\perp} \right)^{\perp} \right)^{\perp} \right)^{\perp} \tag{3.7}$$

This mechanism has two layers of serial composition and two layers of parallel composition. The upper parallel composition represents the assembly of the three legs. The upper serial compositions represent the assembly of the three revolute joints with the parallelogram unit. The lower parallel compositions represent the closure of the parallelogram chain in the parallelogram units. Finally, the lower serial compositions represent the addition of the two revolute joints on one side of the parallelogram units. Each serial layer can lead to a type of local mobilities while each parallel layer can lead to a type of overconstraints. Those are quite different depending if T_{kij} represents a revolute or a spherical joint. The total local mobilities and the overconstraints in the parallelogram unit are noted R_{M1} and R_{C1} , and the local mobility and overconstraints in the Delta mechanism are noted R_{M2} and R_{C2}. The total number of rigid links is noted n, the total number of joints is given by m, and f_i represents the number of DOF of joint i. The values obtained for revolute or spherical configurations are summarised in Table 3.1. In both cases, the end-effector mobility is 3 translational DOF and the dimensions of end-effector mobility, overconstraints, and local mobilities satisfy the following relation:

$$M = 6\,(n - m - 1) + \sum_{i=1}^{m} (f_i) + R_{C1} + R_{C2} - R_{M1} - R_{M2} \tag{3.8}$$

In case of revolute joints, the parallelogram unit forms a planar mechanism and and each unit has therefore 3 overconstraints. Each leg constraints the rigid platform in two rotations. When assembling the three legs of the Delta mechanism, three of these constraints are redundant. In case of spherical joints, the two long bars of the parallelogram unit are free to rotate along their axis. Each parallelogram unit has two local mobilities and zero overconstraints. Each leg constraints the platform in one orientation and the combined action of the legs combine the three orientation without overconstraining the mechanism. Although the end-effector mobility is the same for these two versions of the Delta mechanism, the number and distribution of local mobilities and overconstraints on the different layers of series-parallel compositions is quite different.

3.2.5 General Series-Parallel Mechanisms

The principle of layer compositions can be extended to any general series-parallel mechanisms. As it was shown, the two types of local mobilities of hybrid mechanisms and the two

Distribution of Mobilities and overConstraints in a 3 DOF Delta mechanism

	T_{kij}: Revolute Joints	T_{kij}: Spherical Joints
R_{M1}	0	3x2=6
R_{C1}	3x3=9	0
R_{M2}	0	0
R_{C2}	3	0
n	17	11
m	21	15
$\sum_{i=1}^{m}(f_i)$	21	39
M	3	3

Table 3.1: Distribution of Mobilities and overconstraints in a 3 DOF Delta mechanism with revolute joints or spherical joints. In practice, the 6 local mobilities of the spherical joints version are often constrained with some rubber springs. The 12 overconstraints of the revolute joints version will introduce manufacturability tolerances that must be respected to limit the internal stresses due to the assembly.

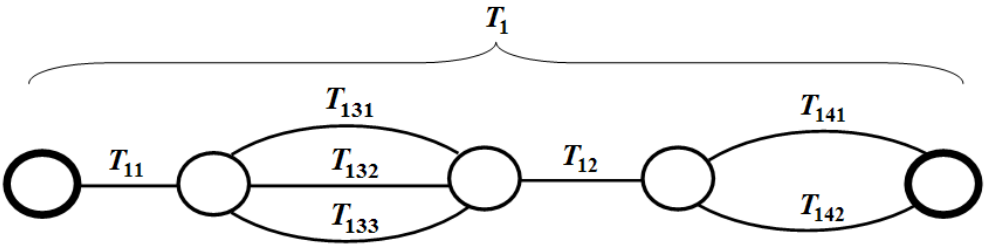


Figure 3.5: A Example of a general series-parallel chain described by Equation 3.9

type of overconstraints of Delta mechanisms with revolute joints occur on different layers of series-parallel compositions. They have to be computed separately and they distribute on different levels of the mechanism. One can consider a general chain i formed by a serial sequence of $j_i = 1..s_i$ joints represented by the twist system T_{ij} and $j_i = s_i..p_i$ parallel units. Each parallel unit $j_i = s_i..p_i$ is formed by parallel chains $k_{ij} = 1..q_{ij}$ represented by the twist system T_{ijk}. Figure 3.5 shows an example of such a chain. The twist system T_i of chain i is given by

$$T_i = \sum_{j_i=1}^{s_i} T_{ij} + \sum_{j_i=s_i}^{p_i} \left(\sum_{k_{ij}=1}^{q_{ij}} (T_{ijk})^{\perp} \right)^{\perp} \tag{3.9}$$

The twist systemsT_i, T_{ij}, and T_{ijk} belong to three different layers of composition. This formula can be used recursively to compute the twist system of any general series-parallel mechanism.

When analysing the first layer of a hybrid mechanism i as presented in Section 3.2.3, $s_i = 0$, p_i is the number of pure parallel mechanisms, q_{ij} is the number of leg of mechanism j and T_{ijk} is the twist system of this leg. In the first layer of a Delta mechanisms i, as presented in Section 3.2.4 $s_i = 0$, $p_i = 1$, $q_{ij} = 3$ and T_{ijk} is the twist system of a Delta leg.

Equation 3.9 can be used on any layer of composition. When considering only a single Delta leg i, then $s_i = 3$, $p_i = 1$, $q_{ij} = 2$, T_{ij} is the twist system of the revolute joint j in the leg and T_{ijk} is the twist system of 2 joints in the parallelogram unit. A last layer is needed to compute T_{ijk} , denoted now T_i, which can be done again using Equation 3.9 with $s_i = 2$, $p_i = 0$, $q_{ij} = 3$, and T_{ij} as the twist system of one joint in the parallelogram unit.

Given a chain i described with Equation 3.9, one layer of local mobilities can occur. Its dimension is given by

$$R_M = dim\,(T_i) - \sum_{j_i=1}^{s_i} dim\,(T_{ij}) - \sum_{j_i=s}^{p_i} dim \left(\sum_{k_{ij}=1}^{q_{ij}} (T_{ijk})^{\perp} \right)^{\perp} \tag{3.10}$$

One layer of overconstraint is also possible. Its dimension is given by

$$R_C = \sum_{j_i=s}^{p_i} \left(dim \left(\sum_{k_{ij}=1}^{q_{ij}} (T_{ijk})^{\perp} \right) - \sum_{k_{ij}=1}^{q_{ij}} dim\,(T_{ijk})^{\perp} \right) \tag{3.11}$$

This computation can be done on each layer of series-parallel composition to reveal all the local mobilities and overconstraints of any series-parallel mechanism.

3.2.6 Cross-Jointing Mechanisms

A cross-jointing mechanism has two legs between its base and its end-effector that are connected together by a joining chain somewhere on the legs. Figure 3.6 shows the graph representation of such a mechanism. T_{i1} and T_{i2} are the twist systems of the first part and the second part of leg i and T_j is the twist system of the joining chain. These twist systems can represent any layers of series-parallel compositions described by the recursive Equation 3.9 for general series-parallel mechanism. The top layer of composition, shown in Figure 3.6 is not a series-parallel graph between the base and the end-effector. It is however a series-parallel graph between the two jointing links shown in bold on the figure.

Since the graph is series-parallel, the serial and parallel laws can still be used to compute the mobility between the terminal links, to which global variables can be assigned. The mobility between the base and the end-effector is a function of these global variables, the local variables that are present in the base chain T_{11} , T_{21}, and the local variables that are present in the end-effector chain T_{12}, T_{22}. To compute the mobility between the base and the end-effector, these mobilities must be distributed on the base link and the end-effector link. Section 2.3 presents a method to solve screw systems under homogenous equation. This method can be used to distribute the mobilities and to find the twist system between the base and end-effector of cross-jointing mechanisms.

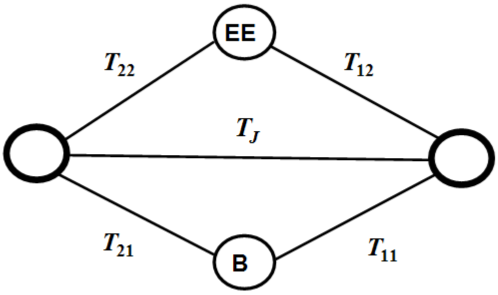


Figure 3.6: Graph Representation of a Cross-Jointing Mechanism

3.2.7 Non Series-Parallel Mechanisms

As it was seen in this section, it is possible to compute the equivalent twist system of serial chains by addition of twist systems, and the equivalent twist system of parallel chains by the intersection of twist systems. Those correspond to the graph reduction operations a) and b) respectively. For any series-parallel mechanisms, a twist system can be obtained between the 2 terminal links, generally the base and the end-effector. The dimension of the twist system corresponds to the number of DOF between the base and the end-effector. A global variable is assigned to each DOF of the mechanism, and by reconstruction of the original graph, these global variables are distributed on each joint of the mechanism.

For mechanisms that are not series-parallel, additional rules must be applied to compute and distribute their mobilities. As pointed out in [14], there is an analogy with series-parallel electrical circuits. If an electrical circuit can be represented as a series-parallel graph, the equivalent resistance between 2 terminal points can be computed using only the series law and parallel law for resistors. If the circuit is not series-parallel, one must use Kirchhoff's laws to find the equivalent resistance. An obvious way to determine if a graph is series-parallel is to verify that it is indeed possible to reduce it to a K_2 graph by a sequence of serial and parallel reductions. An alternative and sometime faster method is to ensure that the graph doesn't contain the complete graph K_4 as a minor. A graph B is a minor of a graph A if it is a subset of vertices and edges of A to which zero or more edge contractions has been performed. An edge contraction is the removal of an edge of a graph and the merging of the two vertices it previously connected. Figure shows a K_4 graph, which is also the smallest non series-parallel graph possible.

The corresponding graph of parallel mechanisms with configurable platforms always have the K_4 graph as a minor therefore, their mobility and overconstraints cannot be computed using the standard method of Chapter 2, or the extended method presented in this section.

3.3 Mobility Analysis of Parallel Mechanisms with Configurable Platforms

The concept behind parallel robots with configurable platforms is to replace the rigid (non-configurable) platform with a closed kinematic chain from which the mobility can

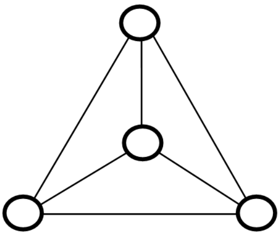


Figure 3.7: The complete K_4 graph. Non series-parallel graph always have the K_4 graph as a minor graph.

be used to perform additional tasks such as grasping or actuation of a tool. The platform is actuated from the base by a coupling action of additional legs which number depends on the mobility of the configurable platform.

Although they offer interesting advantages and different possibilities than parallel mechanisms, they have been barely addressed in the literature so far. One of the main reasons is maybe due to the fact that they have a non series-parallel architecture which make the standard method of mobility, overconstraints and synthesis non applicable to them. Mohamed and Gosselin [56] proposed in 2005 a first generalization of the concept of both planar and spatial PRCPs. In their article, they proposed a method to calculate their mobility using a matrix representation in which each row represents the constraints imposed by an independent closed-loop $i = 1..c$ and each column represents the velocity of a joint. In a PMCP, each independent closed-loop is formed by two legs and one platform segment. If the twist systems of the leg $i = 1..c$ are noted $T_{Li} = \{\$_{Li1}, \cdots \$_{Lij}, \cdots \$_{Liq_i}\}$, where $q_i = dim\left(T_{Li}\right)$and $\$_{Lij}$ is a basis screw of T_{Li} the twist systems of the platform segments $i = 1..c$ are noted $T_{Pi} = \left\{\$_{Pi1}, \cdots \$_{Pij}, \cdots \$_{Piq_j}\right\}$, the constraint equations can be represented as

$$\begin{bmatrix} T_{L1} & -T_{L2} & \cdots & \mathbf{0} \\ 0 & T_{L2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \mathbf{0} \\ -T_{L1} & \mathbf{0} & \mathbf{0} & T_{Lc} \end{bmatrix}_{6c,\sum_{i=1}^{n} dim(T_{Li})} \begin{bmatrix} \dot{\boldsymbol{\theta}}_{L1} \\ \dot{\boldsymbol{\theta}}_{L1} \\ \vdots \\ \dot{\boldsymbol{\theta}}_{Lc} \end{bmatrix} = \cdots$$

$$\begin{bmatrix} T_{P1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & T_{P2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & T_{Pc} \end{bmatrix}_{6c,\sum_{i=1}^{n} dim(T_{Pi})} \begin{bmatrix} \dot{\boldsymbol{\theta}}_{P1} \\ \dot{\boldsymbol{\theta}}_{P2} \\ \vdots \\ \dot{\boldsymbol{\theta}}_{Pc} \end{bmatrix} \tag{3.12}$$

where $\dot{\boldsymbol{\theta}}_{Li}$ and $\dot{\boldsymbol{\theta}}_{Pi}$ are the vectors representing the global variables of leg i and platform segment i respectively. The column variables $\$_{Lij}$ and $\$_{Pij}$ have been compressed into their respective screw systems T_{Li} and T_{Pi} in order to make the notation easier. This system of linear equations has $6c$ equations and $\sum_{i=1}^{n}\left(dim\left(T_{Li}\right) + dim\left(T_{Pi}\right)\right)$ unknowns. If the legs and the platform of the PMCP don't have local mobilities, the total number of unknowns in equal to the total number of joints m in the mechanism and $\sum_{i=1}^{n}\left(dim\left(T_{Li}\right) + dim\left(T_{Pi}\right)\right) = m$. Also, only if the mechanism doesn't present any overconstraints, the first Chebychev-Grübler-Kutzbach criterion of Equation 2.2 can be applied and the mobility of the mechanism is then given by $M = m - 6c$. Assuming that the mobility is controlled at each leg of the robot by a set of active joints $\dot{\boldsymbol{\theta}}_{Li,act}$ and that the set of leg passive (non-actuated) joints is noted $\dot{\boldsymbol{\theta}}_{Li,pas}$, Equation 3.12 can be rewrited as

$$\begin{aligned}
&\begin{bmatrix} T_{L1,act} & \mathbf{0} & \cdots & \mathbf{0} \\ 0 & T_{L2,act} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & T_{Lc,act} \end{bmatrix}_{6c,M} \begin{bmatrix} \dot{\boldsymbol{\theta}}_{L1,act} \\ \dot{\boldsymbol{\theta}}_{L2,act} \\ \vdots \\ \dot{\boldsymbol{\theta}}_{Ln,act} \end{bmatrix} = \cdots \\
&\begin{bmatrix} T_{L1} & -T_{L2} & \cdots & \mathbf{0} & T_{P1} & \mathbf{0} & \cdots & \mathbf{0} \\ 0 & T_{L2} & \cdots & \mathbf{0} & \mathbf{0} & T_{P2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \mathbf{0} & \vdots & \vdots & \ddots & \mathbf{0} \\ -T_{L1} & \mathbf{0} & \mathbf{0} & T_{Lc} & \mathbf{0} & \mathbf{0} & \mathbf{0} & T_{Pc} \end{bmatrix}_{6c,m-M} \begin{bmatrix} \dot{\boldsymbol{\theta}}_{L1,pas} \\ \dot{\boldsymbol{\theta}}_{L2,pas} \\ \vdots \\ \dot{\boldsymbol{\theta}}_{L3,pas} \\ \dot{\boldsymbol{\theta}}_{P1,pas} \\ \dot{\boldsymbol{\theta}}_{P2,pas} \\ \vdots \\ \dot{\boldsymbol{\theta}}_{Pn,pas} \end{bmatrix}
\end{aligned} \tag{3.13}$$

Since $6c = m - M$ in mechanisms that doesn't present local mobilities and overconstraints, the second matrix is a rectangular matrix. Using numerical values, it is possible to invert this matrix in order to obtain the velocity of each passive joints as function of the velocity of the active joints.

As it is shown, this framework is not valid for overconstrained mechanisms, since they have a non-rectangular matrix of passive joints and and the case of overconstrained PRCPs was not mentioned in the article. In addition, the procedure requires the inversion of a matrix which, given its dimension, is not invertible using analytical vector representation. In this section, a general method is presented for the mobility analysis of overconstrainted parallel robots with configurable platforms that leads to an analytical vector representation of their mobility.

3.3.1 Graph Reduction

The first step of the method is to perform the graph reduction by series and parallel laws, so that the local mobilities and overconstraints that may occur in the lower layers are taken into acount. The graph reduction of the mechanism is performed until a non series-parallel 3-connected graph with a twist system for each edge is obtained. The reduced graph of any parallel mechanisms with configurable platforms is a wheel graph such as the graphs shown in Figure 3.8. In those graphs, the centre vertex represents the base, the spokes of the wheel represents the legs and the outer cycle represents the configurable platform. Any wheel graph has the complete graph K_4 as a minor and is therefore non series-parallel.

Either the legs edges or the platform edges can be formed by unlimitedly large and complex mechanisms as long as those are series-parallel. In such a case, it is always possible to

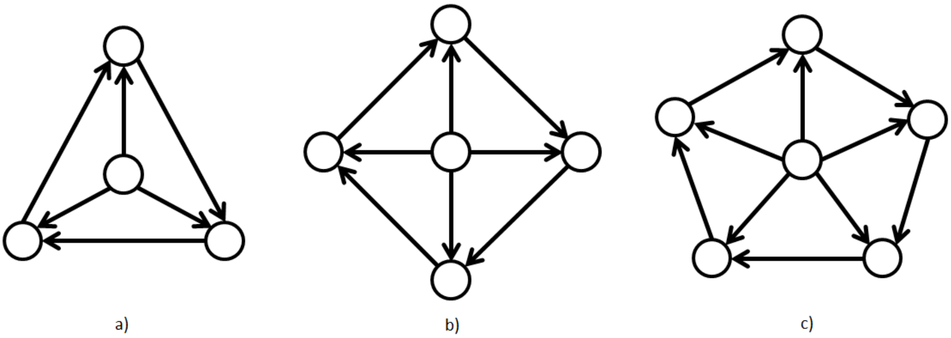


Figure 3.8: Directed wheel graphs a) $W_4 = K_4$ b) W_5 and c) W_6. The centre vertex represents the base, the spokes of the wheel represents the legs and the outer cycle represents the configurable platform. The order i of the graph W_i represents the number of independent legs. The directions of the edges are chosen arbitrarily and determine the direction for positive velocity.

reduce the whole mechanism into a wheel graph. At the end of the graph reduction, a twist system is obtained for each leg and for each platform segments. When during the graph reduction process, some local mobilities or overconstraints appears, such as described in Section 2.3.3, they can be computed separately since they will not affect the mobility between the top layer links represented by the wheel graph vertices. For leg i, the leg twist system is noted $T_{Li} = \{\$_{Li1}, \cdots \$_{Lij}, \cdots \$_{Lin}\}$, where $n = dim\,(T_{Li})$and $\$_{Lij}$ is a basis screw of T_{Li}.

3.3.2 Legs Matrix of Mobilities

The wheel graph and the twist systems associated with each edge are the inputs for the method. Each twist system represents the global mobility of the edge when disconnected from the remainder of the mechanism. As opposed to classical parallel mechanisms, PMCPs don't have a single end-effector but instead have one end-effector for each leg. The mobility of these end-effectors are not independent and are function of the global DOF of the mechanism. The first step is to create the initial leg matrix of mobility. In this matrix, each row of screws represents the mobility of one leg, i.e. one end-effector, and each column represents an independent variable associated to the global DOF of the whole system. The initial matrix ${}^tG$ represents the mechanism without connections to the configurable platform, i.e that each leg are independent. For a mechanism with n legs, the matrix is

$$
{}^0G = \begin{bmatrix} T_{L1} & 0 & 0 & 0 \\ 0 & T_{L2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & T_{Ln} \end{bmatrix}_{6n,\sum_{i=1}^{n} dim(T_{Li})} \tag{3.14}
$$

Each row of G represents the motion of the last link of each leg relatively to the base. It is possible to perform any linear operations on the columns of G to obtain a valid and equivalent representation of the DOF of the system with a different physical meaning. Such linear operations on the matrix columns will be used in the following steps to compute the dependency of the leg mobility introduced by the configurable platform.

3.3.3 Connection of the Platform to the Legs

The mobility of a platform chain $P1$ is represented by the twist system $T_{P1} = \{\$_{P11}, \cdots \$_{P1i}, \cdots \$_{P1n}\}$. When $P1$ is connected, it constraints the mobility of leg $L2$ relatively to leg $L1$. If a variable λ is assigned to each column of the initial leg mobility matrix in Equation 3.14, the connection with the first platform chain requires the following equations to be satisfied:

$$\sum_{j=1}^{dim(T_{L2})} \$_{L2i}\lambda_{L2i} - \sum_{i=1}^{dim(T_{L1})} \$_{L1i}\lambda_{L1i} = \sum_{i=1}^{dim(T_{P1})} \$_{P1i}\lambda_{P1i} \tag{3.15}$$

As pointed out in Section 2.3 the entire solution set can be represented by a restricted and an unrestricted screw systems. The unrestricted screw system representing the set of particular solutions is given in this case by $T_{P1} \bigcap (T_{L1} + T_{L2})$ and correspond to the allowed mobility of the platform segment. The restricted screw system representing the set of solution to the corresponding homogeneous equations is given by $T_{L1} \bigcap T_{L2}$ and corresponds to mobility of the platform relatively to the base when the joints of the platform are locked i.e. when all $\lambda_{P1i} = 0$.

Using this principle, the dependency between the mobility of each leg by the connections of the platform chains can be computed from the leg matrix. The method consists of three steps for each platform chains. If the platform chain T_{Pi} connects the legs T_{Li} and $T_{L(i+1)}$, the 3 steps are then

1) Compute the mobility of leg $L\,(i-1)$ relatively to Li by substracting rows $(i-1)$ from row i in matrix G. The result is a set of screws which are not necessarily independent and span a screw system. From this set of screws, compute a restricted and an unrestricted screw systems such as presented in section 2.3

2) Perform linear operations on the columns of G such that the restricted S^R and unrestricted screw systems S^U depend on separated variables λ to obtain a new matrix G'.

3) Compute the intersection of the platform mobility T_{Pi} with the unrestricted screw system S^U . Perform a change of variables such that the columns of S^U correspond to the intersection $T_{Pi} \bigcap S^U$. The remaining columns can be deleted.

For example, when connecting leg $L1$ and leg $L2$ with platform segment $P1$, the first step is to compute the mobility of leg $L2$ relatively to leg $L1$ by substracting row 1 from row 2 in the matrix G.

$$^{0}G_{(2-1)} = \begin{bmatrix} -T_{L1} & T_{L2} & 0 & 0 \end{bmatrix} \tag{3.16}$$

The restricted screw system of this set of basis screws is given by $S_1^R = T_{L1} \bigcap T_{L2}$ and represent the mobility of the two legs when they are connected together with a rigid platform. The unrestricted screw system S_1^U is the complement of S_1^R and is defined by

$S_1^R \bigoplus S_1^U = T_{L1}$. In the second step, a change of variables is performed so that S_1^R will be canceled in 3.16. After this change of variable, ${}^0G_{(2-1)}$ and ${}^0G'$ become

$$ {}^0G'_{(2-1)} = \left[\begin{array}{c|cccc} 0 & -S_1^U & T_{L2} & 0 & 0 \end{array} \right] \tag{3.17}$$

$$ {}^0G' = \left[\begin{array}{c|cccc} S_1^R & S_1^U & 0 & 0 & 0 \\ S_1^R & 0 & T_{L2} & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & T_{Ln} \end{array} \right]_{6n, \sum_{i=1}^{n} dim(T_{Li})} \tag{3.18}$$

where $S_1^U \cap T_{L2} = \{0\}$ by definition. The third step is to compute the motion allowed by the platform segment, which in this case is given by $T_{P1}^R = T_{P1} \bigcap \left(S_1^U + T_{L2} \right)$. All basis screws in S_1^U and T_{L2} are independent but not necessarily orthogonal. A unique non-orthogonal projection of T_{P1}^R on S_1^U parallel to T_{L2}, noted $T_{P1}^R / S_1^U // T_{L2}$, and T_{P1}^R on T_{L2} parallel to S_1^U, noted $T_{P1}^R / T_{L2} // S_1^U$, can be obtained. Finally, the matrix ${}^1G$ is obtained, which corresponds to the mobility of the mechanism when the platform segment $P1$ is connected.

$$ {}^1G = \begin{bmatrix} S_1^R & T_{P1}^U / S_1^U // T_{L2} & 0 & 0 & 0 \\ S_1^R & 0 & T_{P1}^U / T_{L2} // S_1^U & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & T_{Ln} \end{bmatrix} \tag{3.19}$$

Physically, the first columns represents the motion of leg $L1$ and leg $L2$ when they are connected by a rigid end-effector. The second and third columns represent the distribution of the motion of the platform chain $P1$ on leg L_1 and L_2 respectively.

The key idea behind this method is that the global variables, i.e the columns of the matrix G, that are removed by the connection of the platform chain between two legs are independent of the global variables describing the common mobility of these two legs. Indeed, the connection of a configurable platform chain cannot constraint DOF of the mechanism that would have been present if the platform chain would have been rigid. This method ensures that the proper mobility is calculated each time a platform segment is connected to the legs. The use of the generative matrix G can be related to the work presented in [20], although the vector space operations used here are different. Compared to [20], the use of the leg edges as an initial input results in less vector space operations for parallel mechanisms with configurable platforms and the remaining operations are particularly suitable to perform their mobility analysis with analytical vector representations. The three steps must be repeated for each platform segment until the final matrix is obtained. The number of columns of the final matrix corresponds to the number of DOF of the parallel mechanism with configurable platform. The motion of each leg as function of the global DOF of the mechanism is given by each row of G. Any linear operations on the matrix columns are still possible at this stage to obtain a more convenient representation. It is, for example interesting to separate the DOF of the platform mobility and the DOF of the platform configuration.

3.4 Overconstraints Analysis of Parallel Mechanisms with Configurable Platforms

3.4.1 Self Dual Topogoly of Parallel Mechanisms with Configurable Platforms

The input of the mobility analysis method for parallel mechanisms with configurable platforms described in the previous section is a directed wheel graph with a twist system associated to each edge. Using the principle of restricted and unrestricted screw systems, the mobility of each leg relatively to the base is computed as function of a set of global independent variables of the system, i.e. DOF. The mobility of each platform chain can be computed as the relative mobility between the two connected legs as function of the global variables.

In a more mathematical abstraction, the method gives the solutions to screw systems that are restricted by homogeneous equations described by the cycles of a wheel graph. Any physical phenomena that correspond to this mathematical description can be analyzed with this procedure. As presented in Section 2.2.2, screw systems can be used to describe mobilities between rigid bodies, in which case they are called twist systems, but are also used to describe constraints between rigid bodies, in which case they are called wrench systems.

Overconstraints may occur in closed-loop mechanism and can be defined as special conditions that should be met to close loops. If those conditions are not satisfied, links must slightly deform, generating internal stress on the link and reaction forces at the joint interfaces. Those internal forces are present in the mechanism even when no external forces act on the mechanism and the mechanism is not moving. By consequence, the sum of all internal stresses due to overconstraints on a single link must be equal to zero.

The homogeneous conditions that the wrench systems must satisfy on a rigid body are analogous to the homogeneous conditions that the twist systems must satisfy within a closed-loop. In graph theory, the dual graph B of an original graph A is a graph in which each vertex of B corresponds to a loop of A and vice-versa. Edges that are incident to a vertex in the original graph are in the same direction as the corresponding loop of the dual graph and vice versa. If the twist systems of the dual edges are defined as the reciprocal of the twist systems of the original edges, the mechanism corresponding to the dual graph is dual to the original mechanism. In 2007[13], Davies used the concept of dual graphs to show that the dual mechanism of a planar four-bar linkage is a parallel mechanism with 4 parallel SU legs. The article highlighted the fact that mobilities are often easier to imagine than overconstraints and that the methods used to calculate the mobility of a given dual mechanism can be used to calculate the overconstraints of the original mechanism. Actually, the dual mechanism of a pure parallel mechanism is always a single closed-loop and vice versa. Fig. 1 shows an example of the graph representation of a pure parallel mechanism with 5 legs and its dual single closed-loop.

The mobility of the pure parallel mechanism on the left is $\bigcap S_i$ and the overconstraints of the single-loop on the left is also $\bigcap S_i$. The overconstraints of pure parallel mechanisms corresponds to the restricted wrench systems of their leg which are given by $S_j^{\perp} \bigcap \left(\sum_{i=1}^{j-1} S_i^{\perp} \right)$ for $j = 2..5$ for a pure parallel mechanisms with 5 legs. In a dual way, the mobility of the single closed-loop is also described by the set of twist systems

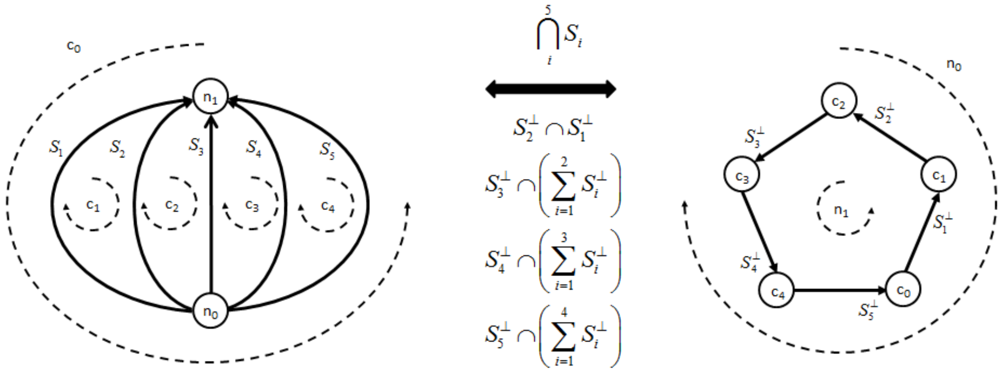

c0
n1
n0
S_1
S_2
S_3
S_4
S_5
c1
c2
c3
c4
$\bigcap_{i}^{5} S_i$
$S_2^\perp \cap S_1^\perp$
$S_3^\perp \cap \left(\sum_{i=1}^{2} S_i^\perp\right)$
$S_4^\perp \cap \left(\sum_{i=1}^{3} S_i^\perp\right)$
$S_5^\perp \cap \left(\sum_{i=1}^{4} S_i^\perp\right)$
n0
c2
c1
c3
c4
c0
n1
$S_1^\perp$
$S_2^\perp$
$S_3^\perp$
$S_4^\perp$
$S_5^\perp$

Figure 3.9: Graph representation of a parallel mechanism with five legs and its dual single closed-loop. The screw system over the big arrow represents the mobility of the parallel mechanism and the overconstraints of the single closed-loop. The screw systems below the arrow represent the overconstraints of the parallel mechanism and the mobility of the single closed-loop.

$S_j^\perp \bigcap \left(\sum_{i=1}^{j-1} S_i^\perp\right)$ for $j = 2..5$. Similar dual relations can be found for the dual mechanisms of an hybrid mechanism and a Delta mechanism shown in Figure 3.10 a) and b) respectively.

It is clear that, for most of mechanisms, the mathematical procedure to calculate its mobility differs from the procedure to calculate its overconstraints. It was proposed in Section 3.3.1 that the reduced graph of a PMCP is always a wheel graph. A remarkable property of all wheel graphs is that they are self-dual. The dual mechanism of a PMCP is therefore always a PMCP. This self duality can be exploited to our advantage since, for PMCP, any general method for the mobility analysis can also be used directly for the overconstraint analysis.

Figure 3.11 shows the dual graphs for PMCPs with 3, 4 and 5 legs. The principle can be extended to mechanisms with more legs. In the dual wrench graphs, vertices represent closed-loop while cycles represent rigid links. For each edge, twist systems T are replaced by their reciprocal wrench systems $W = T^\perp$. The consequence of this is that in the wrench graph, the configurable platform is now represented by the spokes of the wheel while the legs are represented by the outer cycle.

Thanks to the self dual topology of PMCPs, the same mathematical procedure can be applied for both their mobility and overconstraint analysis. The method presented in Section 3.3 for the mobility analysis is then used again in the overconstraint analysis. While the mobility and overconstraint methods are mathematically equivalent, their physical interpretation is different. The next section presents the physical interpretation of the mobility calculation method of PMCPs when it is applied to the calculation of the overconstraints.

3.4.2 Platform Matrix of Overconstraints

As it was the case for the mobility analysis, the first step is to create an initial matrix from the wrench systems of the spokes of the wheel graph. For a mechanism with n legs,

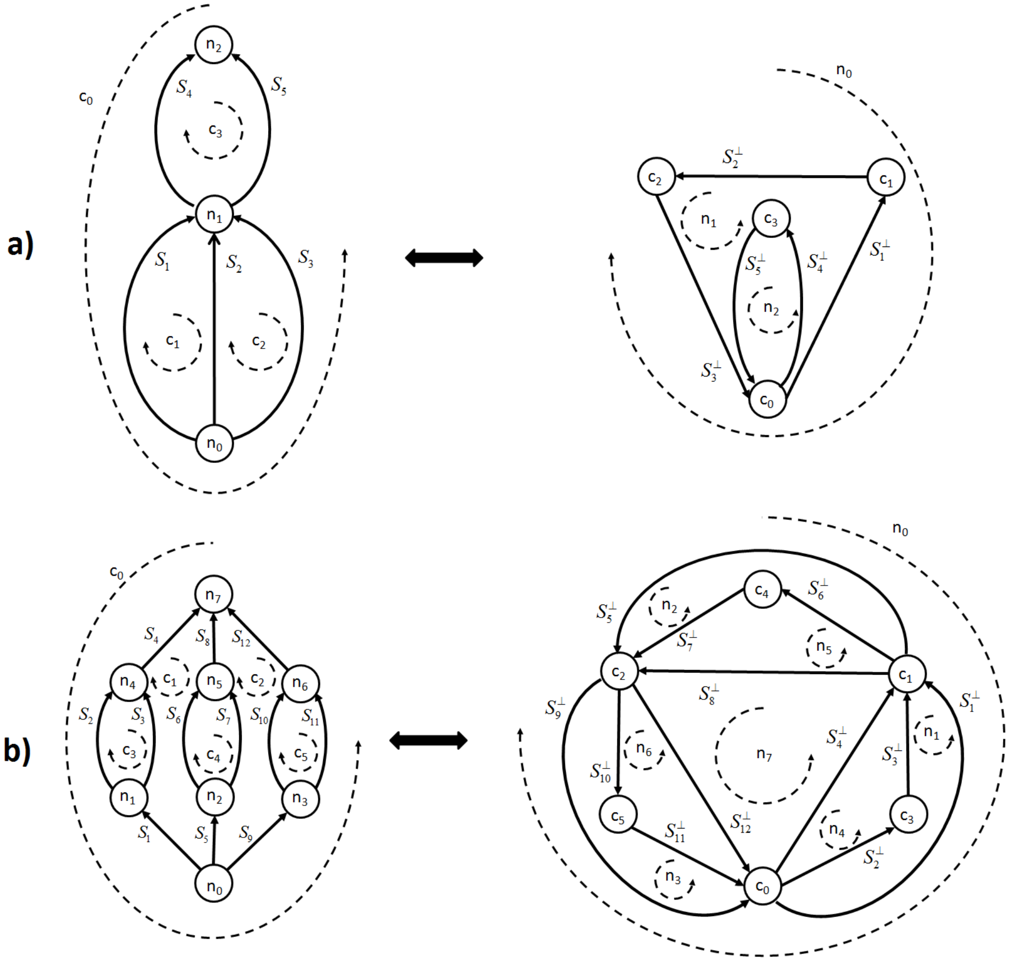


Figure 3.10: Graph representation of the dual mechanisms of a) a hybrid mechanism and b) a Delta mechanism. The distribution of the mobilities in the mechanisms in the left column corresponds to the distribution of the overconstraints in the mechanisms of the right column and vice-versa.

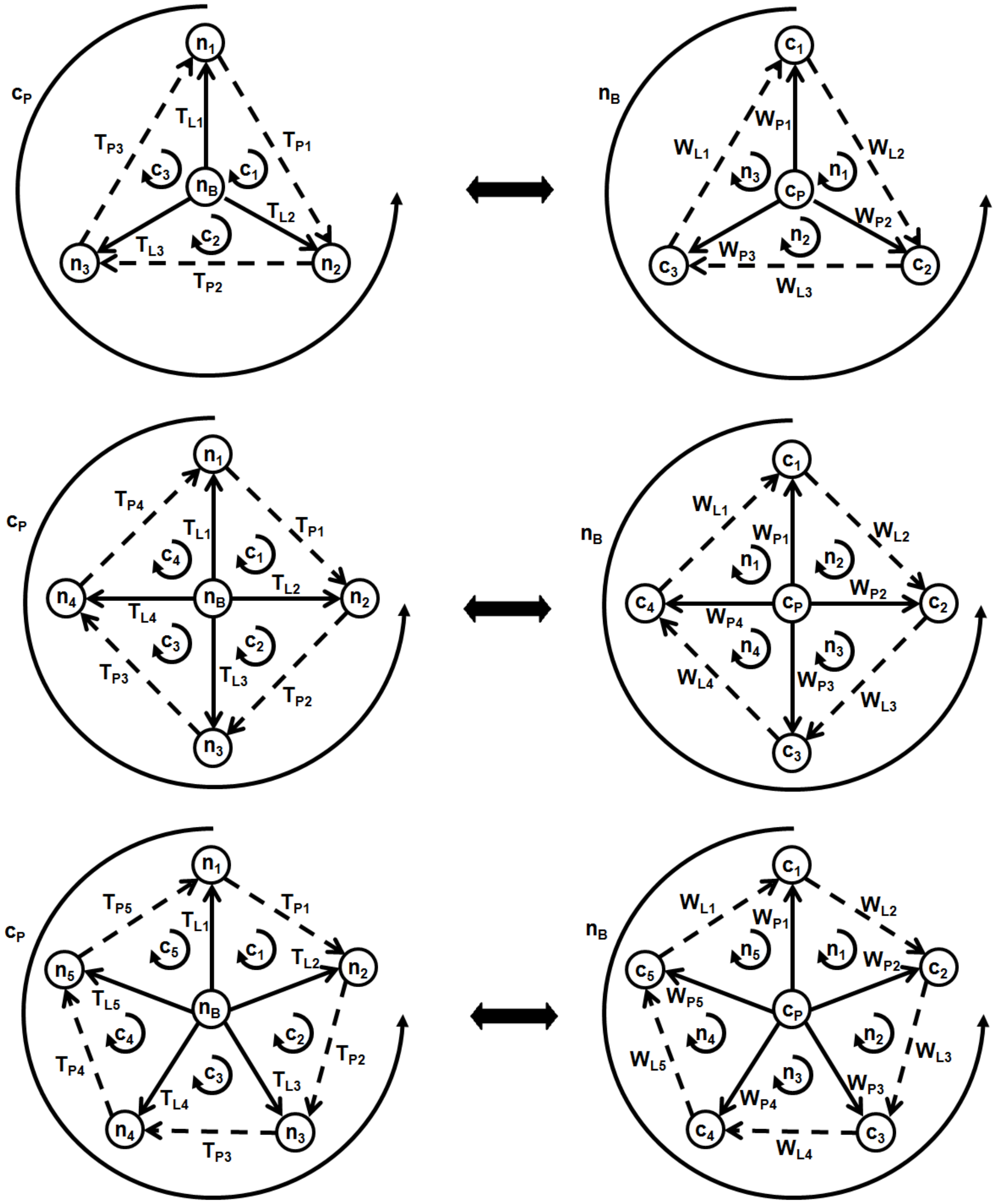


Figure 3.11: Dual twist graphs and wrench graphs for parallel mechanisms with configurable platforms with 3, 4 and 5 legs

the initial platform wrench matrix ${}^tH$ is

$$
{}^tH = \begin{bmatrix} W_{P1} & 0 & 0 & 0 \\ 0 & W_{P2} & 0 & 0 \\ & & \ddots & \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & W_{Pn} \end{bmatrix}_{6n, \sum_{i=1}^{n} dim(W_{Pi})} \tag{3.20}
$$

Physically, this matrix represents the overconstraints of the mechanism when all the joints of all legs are locked. In this situation, any constraint that a platform segment imposes between two legs is an independent overconstraint since the legs are already fully constraint. As it was the case for the mobility matrix, any linear operations on the columns of H are possible.

3.4.3 Connection of the legs to the Platform

The same mathematical procedure than was presented in section 3.3.3 is used to determine the overconstraints of the mechanism. In the initial matrix ${}^tH$, every wrench of the platform segments correspond to an independent overconstraint since all the leg joints are locked. The physical procedure consists of unlocking each leg and compute the remaining overconstraints present in the platform.

The three steps procedure is explained here again when applied to overconstraint computation. Considering that leg W_{Li} connects with platform segments W_{Pi} and $W_{P(i-1)}$, the precedure is

1) Compute the constraints that can be applied on the leg by platform segment P_i and P_{i-1} by substracting the row $i-1$ from row i in matrix H. The results is a set of screws which are not necessarily independent and span a screw system. Compute the restricted and unrestricted screw systems such as presented in section 2.3

2) Perform linear operations on the columns of H such that the restricted S^R and unrestricted screw systems S^{UR} depends on separated variables λ to obtain a new matrix H'.

3) Compute the intersection of the leg constraints W_{Li} with the unrestricted screw system S^{UR} . Perform a change of variables such that the columns of S^U contract into the intersection $W_{Li} \bigcap S^U$. The remaining columns can be deleted.

The global variables, i.e the columns of the matrix H, that are removed in this process correspond to wrenches that could have been applied by the platform segments on the leg but cannot be balanced by the leg wrench system. These variables are independent of the variables describing the common wrench system between the two platform segments. The three steps must be repeated for each platform link until the final matrix H is obtained. The number of columns of the final matrix represents the number of overconstraints of the mechanism. The rows represent the internal forces introduced by the overconstraints into the platform segments.

3.5 Summary of the General Method

In a parallel mechanism with configurable platform, each leg and each platform segment is formed by a series-parallel mechanism. In section 3.2.5, it was shown that the series and parallel rules used in parallel mechanism analysis can be extended to any mechanism as long as their associate graph is series-parallel. This results in a computation of a twist system and its reciprocal wrench between the two terminal links of the series-parallel mechanism. During this analysis, local mobilities and local overconstraints can be revealed. They can be treated separately since they do not affect the global mobility or overconstraints distributed in the top layer of the parallel mechanism with configurable platform. The top layer of a PMCP corresponds to a wheel graph in which the central vertex is the base, the spokes represents the legs, the rim vertices represents the platform connectors that are connected to both the leg and the platform, and the edges of the rim represent the platform segments.

The input of the method is a directed wheel graph with a twist system associated to each edge. The output is the distribution of a set of independent variables on each screw system of the spoke edges such that the sum of the finite screws in each cycle of the wheel graph is zero. The three basic screw systems operations that are used during the process are the intersection of screw systems, calculation of complementary screw systems and parallel projection of screw systems. One consists in the computation of the restricted and unrestricted screw systems for two adjacent spoke edges. The second consists in the computation of the intersection of the rim edges with the unrestricted screw system of the two spokes they connect. Linear changes of variables are performed from the calculation of the restricted and unrestricted screw systems for two adjacent spoke edges. The independent variables that were lost in the intersection operations between the rim edges and the unrestricted screw system of the spoke edges correspond to the dependency created by the rim edges.

In mobility analysis of closed-loop mechanisms, the sum of the velocities of the joints forming a closed-loop must be zero. In the overconstraint analysis of closed-loop mechanisms, the sum of the internal forces created by the overconstraints on any rigid links must be equal to zero. The dual graph of a given original planar graph is a graph that has a vertex corresponding to each plane region of the original graph, and an edge joining two neighbouring regions for each edge in the original graph. Wheel graphs are always planar and have the property to be self-dual.

By using as input to the method the original graph of twist systems or the corresponding dual graph of wrench systems, both the mobility and the overconstraints can be computed with the same mathematical method. The physical interpretations of the mathematical concepts involved in the method are of course quite different. Table 3.2 presents a comparison of the physical interpretation of the method for the mobility and the overconstraint analysis.

PMCPs have the remarkable property of having a self dual topology. This is due to the fact that pure parallel mechanisms are dual to a single closed-loop and vice-versa. Combining a single closed-loop and a parallel mechanism together results therefore in a self dual topology and this property can be used to facilitate their general analysis. The next section describes the mobility and overconstraints of two PMCPs using the proposed method. It will be shown that the methods lead to the correct description of the mobility and overconstraint of the mechanism, but also that the method can be used to obtain an analytical vector representation of the solutions.

Mathematical Concept	Physical interpretation in mobility analysis	Physical interpretation in overconstraint analysis
Vertices of the wheel graph	Terminal links	Closed-loops of the mechanism
Cycles of the wheel graph	Closed-loops of the mechanism	Terminal links
Spokes of the wheel graph	Leg twist systems	Platform wrench systems
Outer edges of the wheel graph	Platform twist systems	Leg wrench systems
rows of the matrix	Velocity of leg attach points relatively to the base	Internal forces transmitted by a platform segment due to overconstraints
Columns of the matrix	Independent mobilities for the whole mechanism	Independent overconstraints of the whole mechanism
Initial matrix	All leg are free and are not connected to the platform	All joints of all leg are locked and are connected to the platform
Restricted screw systems	Commom mobility of 2 legs when connected to a rigid platform	Overconstraint present in the platform when not connected to the leg
Unrestricted screw systems	Relative mobility of 2 legs when they are not connected	Overconstraints imposed the 2 platform segments when they are rigidly connected to the base
Intersection operations	Relative mobility permited by the platform segment between two legs	Overconstraints permited by the legs
Final matrix	Distribution of the global DOF of the mechanism on the mobility of each leg	Distribution of the global overconstraints of the mechanism on the internal stress of platform segment

Table 3.2: The particular self dual topology of PMCPs allows the use of the same mathematical method for the calculation of their mobility and the calculation of their overconstraints. Each mathematical concept of the proposed method has then a dual physical interpretation.

3.6 Examples

To illustrate the proposed method, this section presents the mobility and overconstraints analysis of two parallel mechanisms with configurable platforms. The first example is a 4-RRR parallel mechanism with a planar 8R configurable platform for which the analysis is performed numerically. Numerical computations are sometime less tedious to perform and could be theoretically automated in a computer algorithm. A problem concerning automated algorithms is that the method repeatedly involves the computation of rank of matrices so round-off errors must not introduce errors in the computation of the dimensions of the various screw systems involved.

Numerical computations have also the disadvantage that they can only be performed for a given configuration, and the resulting velocities and internal forces cannot be generalized to other configuration of the mechanisms. If the mechanism is not already in a singular configuration, the number of DOF and the number of overconstraints is constant for any non-singular configuration.

The second example presented in this section is the PentaG mechanism, which serves as architecture for the PentaG robot presented in the following chapters of this thesis. In this example, the analysis is computed using a vector representation of the screw systems that is not dependent on the mechanism configuration. Although somehow more tedious to perform, the vector based representation obtained is valid for any configuration and gives much more insights in the nature of the mobility and overconstraints present in the PentaG mechanism.

3.6.1 Example 1: a 4-RRR Parallel mechanisms with a planar 8R configurable platform

The mobility and overconstraints analysis of a pure 4-RRR parallel mechanism with a rigid platform was presented in Section 2.2.4. The example used in this section is a modified version of this mechanism in which the rigid platform is replaced by a planar chain made of 8 revolute joints, as shown in Figure 3.12. Since the mechanism is not anymore represented by a series-parallel mechanism, the extended method presented previously in this chapter in Section 3.2 cannot be used to analyze the mobility and overconstraints of this mechanism. However, the proposed method for parallel mechanisms with configurable platforms can be used directly here for this purpose.

3.6.1.1 Graph Reduction and Edge Screws Systems

The first step of the method is to reduce the corresponding graph of the mechanism to a wheel graph where the centre vertex represents the base and the outside rim represents the configurable platform. For each edge of this graph, a twist system and the reciprocal wrench system are computed. For the 4-RRR robot with a 8R planar configurable platform, the reduced graph of twist systems and the corresponding dual graph of wrench systems is presented in Figure 3.13. In Table 3.3, a screw system basis is given for both the twist system and the reciprocal wrench system of each leg $L1$, $L2$, $L3$ and $L4$ and each platform segment $P1$, $P2$, $P3$ and $P4$. In the notation used, $\mathbf{s}_{Lij}$ and , $\mathbf{r}_{Lij}$ are respectively the axis of rotation and the position of joint j of leg i and $\mathbf{s}_{Pij}$ and $\mathbf{r}_{Pij}$ are respectively the axis of rotation and the position of joint j of the platform segment i.

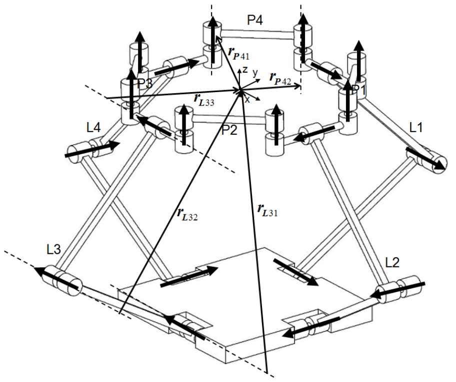


Figure 3.12: A 4-RRR parallel robot with a 8R configurable platform

For a given screw system, the set of independent basis screws that span the screw system is not unique, and any convenient set of basis screws can be used as inputs to the method. In Equations 3.21, linear combinations of the basis screws that represent the 3 joints of leg $L1$ are performed so that the new basis screws represent now a translation in the y direction, a translation in the z translation, and a rotation around the x axis. Equation 3.23 shows similar transformations applied to the twist system of platform segment T_{P1}. A basis for each leg and platform twist system is shown in Table 3.3.

$$T_{L1} = T_{L3} = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{x} \\ \mathbf{r}_{Li3} \times \mathbf{x} \end{bmatrix} \right\} i = 1, 3 \tag{3.21}$$

$$T_{L2} = T_{L4} = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{Li3} \times \mathbf{x} \end{bmatrix} \right\} i = 2, 4 \tag{3.22}$$

$$T_{Pi} = \left\{ \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{P12} - \mathbf{r}_{P11}) \times \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{z} \\ \mathbf{r}_{P12} \times \mathbf{z} \end{bmatrix} \right\} \tag{3.23}$$

3.6.1.2 Mobility Analysis

The first step is to construct the initial leg matrix of Mobility G for the situation where all the legs are free and are not connected to the platform. Each rows of screws represent the mobility of a leg relatively to the base and each column represents an independent variable associated with a DOF of the mechanism. This matrix is useful to keep track of the influence of changes of variables and dependencies of variables introduced by the connection of the platform on the legs.

$$^{t}G = \begin{bmatrix} T_{L1} & 0 & 0 & 0 \\ 0 & T_{L2} & 0 & 0 \\ 0 & 0 & T_{L3} & 0 \\ 0 & 0 & 0 & T_{L4} \end{bmatrix}_{24,12} \tag{3.24}$$

The first platform connection T_{P1} connects leg 2 to leg 1. The first step consists in a change of variables $\lambda \to \lambda'$ so that the screws motion from leg 1 to leg 2 for the restricted

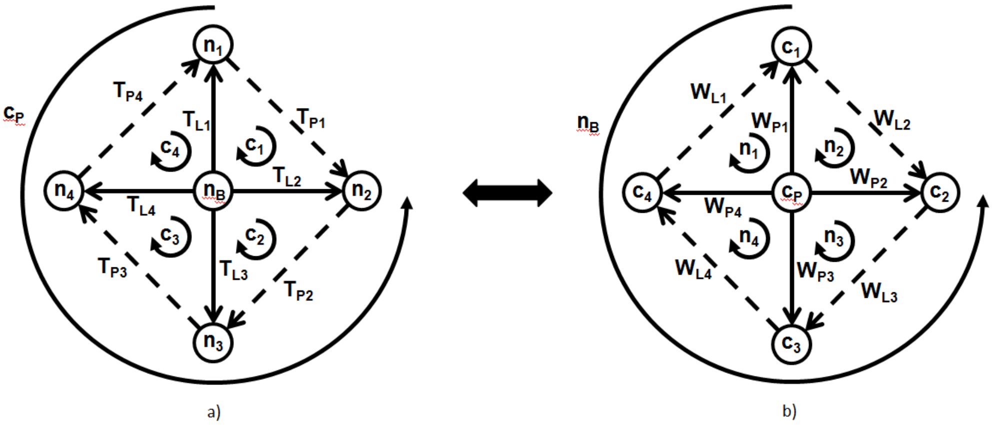


Figure 3.13: The dual reduced graphs of a) the twist systems and b) the wrench systems of the 4-RRR+8-R parallel mechanism with configurable platform

Legs and Platform Twist Systems
$T_{L1} = T_{L3} = \left\{ \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix}, \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix}, \begin{bmatrix}\mathbf{x}\\ \mathbf{0}\end{bmatrix} \right\} i = 1,3$
$T_{L2} = T_{L4} = \left\{ \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix}, \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix}, \begin{bmatrix}\mathbf{y}\\ \mathbf{0}\end{bmatrix} \right\} i = 2,4$
$T_{Pi} = \left\{ \begin{bmatrix}\mathbf{0}\\ (\mathbf{r}_{Pi2} - \mathbf{r}_{Pi1}) \times \mathbf{z}\end{bmatrix}, \begin{bmatrix}\mathbf{z}\\ \mathbf{r}_{Pi2} \times \mathbf{z}\end{bmatrix} \right\} i = 1..4$

Legs and Platform Wrench Systems
$W_{L1} = W_{L3} = \left\{ \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix}, \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix}, \begin{bmatrix}\mathbf{y}\\ \mathbf{0}\end{bmatrix} \right\} i = 1,3$
$W_{L2} = W_{L4} = \left\{ \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix}, \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix}, \begin{bmatrix}\mathbf{x}\\ \mathbf{0}\end{bmatrix} \right\} i = 2,4$
$W_{Pi} = \left\{ \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix}, \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix}, \begin{bmatrix}(\mathbf{r}_{Pi2} - \mathbf{r}_{Pi1})\\ \mathbf{r}_{Pi1} \times \mathbf{r}_{Pi2}\end{bmatrix}, \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} \right\} i = 1..4$

Table 3.3: Sets of basis screws for the legs and the platform screw systems of the 4RRR+8R mechanism

and unrestricted screw systems depend on separated variables. In this case, the change of variables is given by $\lambda_1' = \lambda_1 + \lambda_5$.

$$
{}^{t}G' = \left[\begin{array}{c|cccccccc}
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{x}\\\mathbf{0}\end{bmatrix} & 0 & 0 & 0 & 0 & 0 \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & 0 & 0 & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \begin{bmatrix}\mathbf{y}\\\mathbf{0}\end{bmatrix} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & T_{L3} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & T_{L4}
\end{array}\right] \tag{3.25}
$$

The second step is to compute the intersection of the leg connection T_{P1} with the unrestricted 5 DOF screw system of row 2 and row 1. This represents the motion allowed by the platform segment from leg 1 to leg 2. In this case, the five variables λ_2 to λ_5 combine into a 1 DOF motion $[\mathbf{0}\,(\mathbf{r}_{P12} - \mathbf{r}_{P11}) \times \mathbf{z}]^T$ of the platform segment $P1$.

$$
{}^{1}G = \begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & -\rho_1 \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & 0 & 0 \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_2 \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & 0 & 0 \\
0 & 0 & T_{L3} & 0 \\
0 & 0 & 0 & T_{L4}
\end{bmatrix} \tag{3.26}
$$

where $\rho_2\mathbf{x} + \rho_1\mathbf{y} = (\mathbf{r}_{P12} - \mathbf{r}_{P11}) \times \mathbf{z}$. The matrix ${}^{1}G$ represents the mobility of the mechanism when only the platform segment P1 is connected between leg 1 and 2. The first column shows that the two legs can move together in the z direction. The second column shows that they have also 1 DOF of motion relative to each other in the xy plane. The motion of the platform connection $P1$ can be obtained by substracting row 1 from row 2 in ${}^{1}G$. Using now ${}^{1}G$ as input, the procedure is repeated again for platform segments $P2$, $P3$ and $P4$ to obtain ${}^{2}G$, ${}^{3}G$ and ${}^{4}G$. For each segment connection, the first step is to compute the restricted and unrestricted screw systems between the two connected legs and proceed to a change of variable accordingly to obtain ${}^{i}G'$. The second step is to compute the intersection of the platform segment motion with the unrestricted screw system representing the motion between the two connected legs to obtain the next matrix. The procedure for each platform connection is shown in the following equations

$$
{}^{1}G' = \left[\begin{array}{c|ccccc}
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & -\rho_1 \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & 0 & 0 & 0 & 0 \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_2 \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & 0 & 0 & 0 & 0 \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & 0 & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \begin{bmatrix}\mathbf{x}\\\mathbf{0}\end{bmatrix} & 0 \\
0 & 0 & 0 & 0 & 0 & T_{L4}
\end{array}\right]
\quad
{}^{2}G = \begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & -\rho_1 \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & 0 \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_2 \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & 0 \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_1 \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & 0 \\
0 & 0 & T_{L4}
\end{bmatrix} \tag{3.27}
$$

The change of variables applied are ${}^{1}\lambda_1' ={}^{1} \lambda_1 + \lambda_4^1$ and ${}^{2}\lambda_2 ={}^{1} \lambda_2 + \rho_1 \cdot{}^{1} \lambda_3$ where $\rho_1\mathbf{y} - \rho_2\mathbf{x} = (\mathbf{r}_{P22} - \mathbf{r}_{P21}) \times \mathbf{z}$.

$$
{}^{2}G' = \left[\begin{array}{c|cccc}
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & -\rho_1 \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & 0 & 0 & 0 \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_2 \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & 0 & 0 & 0 \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_1 \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & 0 & 0 & 0 \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & 0 & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \begin{bmatrix}\mathbf{y}\\\mathbf{0}\end{bmatrix}
\end{array}\right]
\quad
{}^{3}G = \begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & -\rho_1 \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_2 \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_1 \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & -\rho_2 \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix}
\end{bmatrix} \tag{3.28}
$$

where $-\rho_2\mathbf{x} - \rho_1\mathbf{y} = (\mathbf{r}_{P32} - \mathbf{r}_{P31}) \times \mathbf{z}$.

The motion from leg 4 to leg 1 corresponding to row 1 minus row 4 of ${}^3G$ is given by the single screw $[\mathbf{0}\,(-\rho_1\mathbf{y} - \rho_2\mathbf{x})]^T$. This screw system is by definition unrestricted since it has only 1 DOF, so there is no need to proceed to a change of variables. One of the basis screw of the lst platform segment T_{P4} is $[\mathbf{0}\,(\mathbf{r}_{P42} - \mathbf{r}_{P41}) \times \mathbf{z}]^T$, which intersect perfectly with $[\mathbf{0}\,(-\rho_1\mathbf{y} - \rho_2\mathbf{x})]^T$. The final result is

$$
{}^4G = {}^3G = \begin{bmatrix} \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & -\rho_1\begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} \\ \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_2\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} \\ \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho_1\begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} \\ \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & -\rho_2\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} \end{bmatrix} \tag{3.29}
$$

This parallel mechanism with configurable platform has 2 DOF of mobility. The first column shows that the platform can move in the z direction, even if the platform is rigid. The second column shows that the platform can be configured in 1 DOF, which create a motion in each of the 4 legs.

3.6.1.3 Overconstraints Analysis

Due to the duality between mobilities and overconstraints analysis of PMCPs, the exact same method is applied in a dual way to compute the overconstraints of this mechanism. The dual graph of wrench systems is used as input to the method. The first step is to construct the initial platform wrench matrix ${}^tH$ of overconstraints which corresponds to the overconstraints in the mechanism if all of the joints of all legs are locked. In this situation, any constrain imposed by the platform segments on two legs corresponds to an independent overconstraint. Each row of screws represents the internal forces present in a single platform segment due to manufacturing misalignment and each column represents an independent misalignment.

$$
{}^0H = \begin{bmatrix} W_{P1} & 0 & 0 & 0 \\ 0 & W_{P2} & 0 & 0 \\ 0 & 0 & W_{P3} & 0 \\ 0 & 0 & 0 & W_{P4} \end{bmatrix} \tag{3.30}
$$

The leg $L2$ connects platform segment $P2$ to $P1$. The first step consists of a change of variables so that the screws constraints from P1 to P2 for the restricted and unrestricted screw systems depend on separated variables. In this case, the change of variables is given by $\lambda_1' = \lambda_1 + \lambda_5$, $\lambda_2' = \lambda_2 + \lambda_6$ and $\lambda_3' = \lambda_3 + \lambda_7$. These 3 variables correspond to overconstraints in the configurable platform and are not dependent of the leg constraints.

$$
{}^0H' = \left[\begin{array}{ccc|ccccccc} \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}(\mathbf{r}_{P12} - \mathbf{r}_{P11})\\\mathbf{r}_{P11} \times \mathbf{r}_{P12}\end{bmatrix} & 0 & 0 & 0 & 0 & 0 & 0 \\ \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & 0 & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}(\mathbf{r}_{P22} - \mathbf{r}_{P21})\\\mathbf{r}_{P21} \times \mathbf{r}_{P22}\end{bmatrix} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & W_{P3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & W_{P4} \end{array}\right] \tag{3.31}
$$

The second step is to compute the intersection of the leg connection W_{L2} with the unrestricted 5 DOF wrench system of row 2 and row 1. In this case, the five variables λ_4 to λ_8 combine into two constraints of W_{L2}.

$$
{}^{1}H=\begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & 0 & \begin{bmatrix}(\mathbf{r}_{P12}-\mathbf{r}_{P11})\\\mathbf{r}_{P11}\times\mathbf{r}_{P12}\end{bmatrix} & 0 & 0\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & -\begin{bmatrix}(\mathbf{r}_{P22}-\mathbf{r}_{P21})\\\mathbf{r}_{P21}\times\mathbf{r}_{P22}\end{bmatrix} & 0 & 0\\
0 & 0 & 0 & 0 & 0 & W_{P3} & 0\\
0 & 0 & 0 & 0 & 0 & 0 & W_{P4}
\end{bmatrix}
\tag{3.32}
$$

where $\rho_2\left(\mathbf{r}_{P22}-\mathbf{r}_{P21}\right)-\rho_1\left(\mathbf{r}_{P12}-\mathbf{r}_{P11}\right)=\mathbf{x}$. The matrix ${}^{1}H$ represents the overconstraints of the mechanism when only the joints of leg $L2$ are unlocked and the joints of the remaining legs are still locked. The first three columns show that the two platform segments share planar overconstraints that are not transmitted to the leg. The last two columns show that their combined constraint also produce 2 overconstraints on leg $L2$. The overconstraints in leg $L2$ can be obtained by substracting row 1 from row 2 in ${}^{1}H$. Using now ${}^{1}H$ as input, the procedure is repeated again for leg $L2$, $L3$ and $L4$ to obtain ${}^{2}H$, ${}^{3}H$ and ${}^{4}H$. For each leg connection, the first step is to compute the restricted and unrestricted screw systems between the two connected platform segment and proceed to a change of variable accordingly to obtain ${}^{i}H'$. The second step is to compute the intersection of the leg constraints with the unrestricted screw system representing the total constraints imposed by the two platform segments on the unlocked leg to obtain the next matrix. The procedure for each leg connection is shown in the following equations

$$
{}^{1}H'=\left[\begin{array}{cccc|cccccc}
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & 0 & \begin{bmatrix}(\mathbf{r}_{P12}-\mathbf{r}_{P11})\\\mathbf{r}_{P11}\times\mathbf{r}_{P12}\end{bmatrix} & 0 & 0 & 0 & 0 & 0\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & -\begin{bmatrix}(\mathbf{r}_{P22}-\mathbf{r}_{P21})\\\mathbf{r}_{P21}\times\mathbf{r}_{P22}\end{bmatrix} & 0 & 0 & 0 & 0 & 0\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & 0 & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}(\mathbf{r}_{P32}-\mathbf{r}_{P31})\\\mathbf{r}_{P31}\times\mathbf{r}_{P32}\end{bmatrix} & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & W_{P4}
\end{array}\right]
\tag{3.33}
$$

$$
{}^{2}H=\begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & 0 & 0 & \begin{bmatrix}(\mathbf{r}_{P12}-\mathbf{r}_{P11})\\\mathbf{r}_{P11}\times\mathbf{r}_{P12}\end{bmatrix} & 0\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & 0 & -\begin{bmatrix}(\mathbf{r}_{P22}-\mathbf{r}_{P21})\\\mathbf{r}_{P21}\times\mathbf{r}_{P22}\end{bmatrix} & 0\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}(\mathbf{r}_{P32}-\mathbf{r}_{P31})\\\mathbf{r}_{P31}\times\mathbf{r}_{P32}\end{bmatrix} & 0\\
0 & 0 & 0 & 0 & 0 & 0 & W_{P4}
\end{bmatrix}
\tag{3.34}
$$

$$
{}^{2}H'=\left[\begin{array}{ccccc|ccccc}
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & 0 & 0 & \begin{bmatrix}(\mathbf{r}_{P12}-\mathbf{r}_{P11})\\\mathbf{r}_{P11}\times\mathbf{r}_{P12}\end{bmatrix} & 0 & 0 & 0 & 0\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & 0 & -\begin{bmatrix}(\mathbf{r}_{P22}-\mathbf{r}_{P21})\\\mathbf{r}_{P21}\times\mathbf{r}_{P22}\end{bmatrix} & 0 & 0 & 0 & 0\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}(\mathbf{r}_{P32}-\mathbf{r}_{P31})\\\mathbf{r}_{P31}\times\mathbf{r}_{P32}\end{bmatrix} & 0 & 0 & 0 & 0\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & 0 & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}(\mathbf{r}_{P42}-\mathbf{r}_{P41})\\\mathbf{r}_{P41}\times\mathbf{r}_{P42}\end{bmatrix}
\end{array}\right]
\tag{3.35}
$$

$$
{}^{3}H=\begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & 0 & 0 & 0 & \begin{bmatrix}(\mathbf{r}_{P12}-\mathbf{r}_{P11})\\\mathbf{r}_{P11}\times\mathbf{r}_{P12}\end{bmatrix}\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & 0 & 0 & -\begin{bmatrix}(\mathbf{r}_{P22}-\mathbf{r}_{P21})\\\mathbf{r}_{P21}\times\mathbf{r}_{P22}\end{bmatrix}\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & 0 & \begin{bmatrix}(\mathbf{r}_{P32}-\mathbf{r}_{P31})\\\mathbf{r}_{P31}\times\mathbf{r}_{P32}\end{bmatrix}\\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\\mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & -\begin{bmatrix}(\mathbf{r}_{P42}-\mathbf{r}_{P41})\\\mathbf{r}_{P41}\times\mathbf{r}_{P42}\end{bmatrix}
\end{bmatrix}
\tag{3.36}
$$

$$
{}^{3}H' = \left[\begin{array}{cccc|ccc}
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & 0 & 0 & 0 & \begin{bmatrix}(\mathbf{r}_{P12}-\mathbf{r}_{P11})\\ \mathbf{r}_{P11}\times\mathbf{r}_{P12}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & 0 & 0 & -\begin{bmatrix}(\mathbf{r}_{P22}-\mathbf{r}_{P21})\\ \mathbf{r}_{P21}\times\mathbf{r}_{P22}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & 0 & \begin{bmatrix}(\mathbf{r}_{P32}-\mathbf{r}_{P31})\\ \mathbf{r}_{P31}\times\mathbf{r}_{P32}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & 0 & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & -\begin{bmatrix}(\mathbf{r}_{P42}-\mathbf{r}_{P41})\\ \mathbf{r}_{P41}\times\mathbf{r}_{P42}\end{bmatrix}
\end{array}\right] \tag{3.37}
$$

$$
{}^{4}H = \begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & 0 & 0 & \begin{bmatrix}(\mathbf{r}_{P12}-\mathbf{r}_{P11})\\ \mathbf{r}_{P11}\times\mathbf{r}_{P12}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & 0 & -\begin{bmatrix}(\mathbf{r}_{P22}-\mathbf{r}_{P21})\\ \mathbf{r}_{P21}\times\mathbf{r}_{P22}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}(\mathbf{r}_{P32}-\mathbf{r}_{P31})\\ \mathbf{r}_{P31}\times\mathbf{r}_{P32}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & 0 & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & -\begin{bmatrix}(\mathbf{r}_{P42}-\mathbf{r}_{P41})\\ \mathbf{r}_{P41}\times\mathbf{r}_{P42}\end{bmatrix}
\end{bmatrix} \tag{3.38}
$$

This parallel mechanism with configurable platform has 6 overconstraints. Each column represents an independent overconstraint i.e. a geometrical condition that needs to be met in order to assemble the mechanism. When these conditions are not perfectly satisfied, the links must stretch or bend to permit the assembly and this will introduce internal stresses in the rigid links. Several overconstraints can be present at the same time and the internal stress that they generate can be added linearly for each link. For this reason it is always possible to perform linear operations on the columns of H and still obtain a valid representation with a different but equivalent physical meaning. Some linear changes of variables are performed on column 4 and 5 of ${}^{4}H$ to obtain a matrix from which it is easier to make a physical interpretation.

$$
{}^{4}H = \begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ -\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ -\mathbf{y}\end{bmatrix} & \begin{bmatrix}(\mathbf{r}_{P12}-\mathbf{r}_{P11})\\ \mathbf{r}_{P11}\times\mathbf{r}_{P12}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ -\mathbf{y}\end{bmatrix} & -\begin{bmatrix}(\mathbf{r}_{P22}-\mathbf{r}_{P21})\\ \mathbf{r}_{P21}\times\mathbf{r}_{P22}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}(\mathbf{r}_{P32}-\mathbf{r}_{P31})\\ \mathbf{r}_{P31}\times\mathbf{r}_{P32}\end{bmatrix} \\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{z}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ -\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & -\begin{bmatrix}(\mathbf{r}_{P42}-\mathbf{r}_{P41})\\ \mathbf{r}_{P41}\times\mathbf{r}_{P42}\end{bmatrix}
\end{bmatrix} \tag{3.39}
$$

The first three columns are due to the fact that the configurable platform is formed by a planar loop. Those types of overconstraints are common to all planar linkages. In a planar linkage made of revolute joints, the sum of the difference in the orientation of the joint axis must be zero. In addition, the sum of differences in height of the joint attachment must also be zero. Columns 4 shows that the base joint of leg 2 and leg 4 must lie in the same plane. Columns 5 shows that the base joint of leg 1 and leg 3 must also be in the same plane. Note that the four base joints don't have to be in the same plane, but only the opposite legs taken 2 by 2. The last column shows that only a certain combination of lengths is possible for the platform segments. If one imagine that the last platform segment $P4$ is a bit shorter than $P1$, $P2$ and $P3$. This would create stretching in $P2$ and $P4$ and compression in $P1$ and $P3$.

All the information needed to compute internal forces due to overconstraints is contained in the matrix ${}^{4}H$. Each row represents the internal stresses produced on platform segments $P1$, $P2$, $P3$ and $P4$. rows can be subtracted to obtain the stress inside the last link of each leg. For example, row 2 minus row 1 gives the stress present in the last link of $L1$. The stress present on the base link is given by the summation of the stress of the

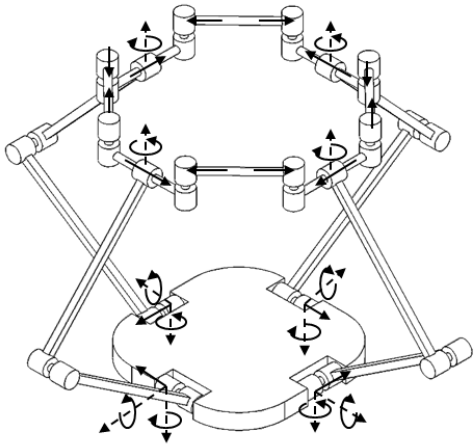


Figure 3.14: Distribution of the internal stresses due to one of the overconstraint of the 4RRR+8R mechanism. The plain arrows represents linear forces and the dot arrows represent torques. The sum of the internal stresses on each rigid body is zero. All the internal stresses are dependent on a single variable. This variable is represented by the 6th column of the platform matrix of overconstraints ${}^4H$, which shows linear compression in platform segments P_1 and P_3 and linear traction in platform segments P_2 and P_4.

four leg. If some additional information are available concerning the actual stiffness of the links, it is possible to compute a relation between all the manufacturing tolerances and the internal stresses and derive some tolerance requirements depending on the maximal stress that are permitted by the links. Figure 3.14 shows the distribution of the internal stresses corresponding to the column 6 of the matrix H_4 in the whole PMCP. Figure 3.15 shows a representation of the physical meaning of the intermediate matrices for the mobility and overconstraint analysis of the 4RRR+8R mechanism.

3.6.2 Example 2: PentaG Mechanism

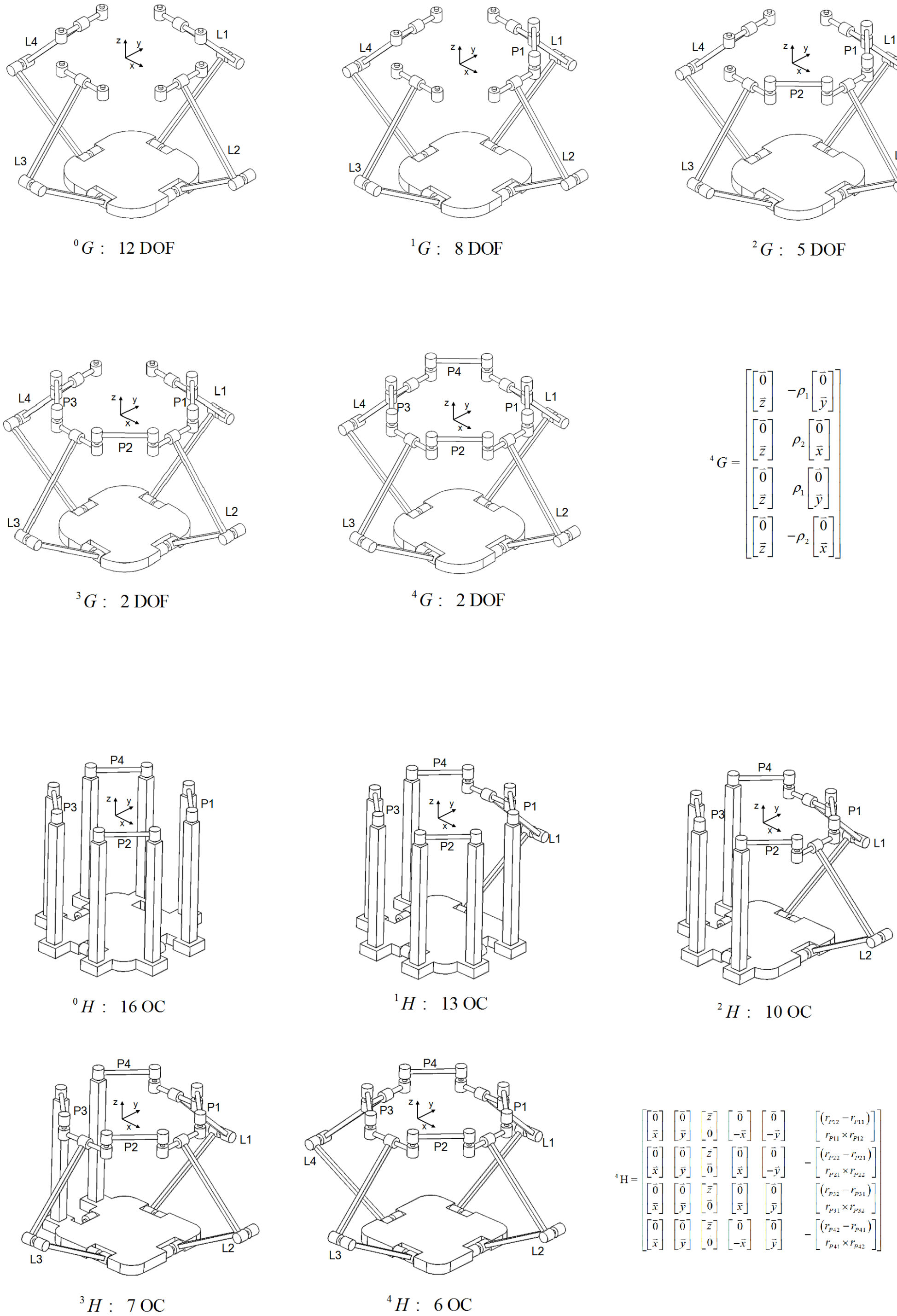
3.6.2.1 Graph Reduction and Screw Systems

The first step of the method is to reduce the graph representing the PentaG into a wheel graph representing the terminal links of the parallel robot with configurable platform using the two standard reduction laws. Figure 3.16 shows the original graph of the PentaG mechanism. In this graph, each link is denoted as n_{ij} where $i = 1..5, p$ represent a leg or the platform and j is the number of the link. The joints are noted m_{ij} in a similar way.

All the joints of the PentaG mechanism are revolute joint with 1 DOF. The screw $\$_{ij}$ corresponding to joint m_{ij} is expressed as

$$T_{ij} = \begin{bmatrix} \hat{\mathbf{s}}_{ij} \\ \mathbf{r}_{ij} \times \mathbf{s}_{ij} \end{bmatrix} \tag{3.40}$$

where $\hat{\mathbf{s}}_{ij}$ is a unit vector oriented in the direction of the axis of rotation of the joint and $\mathbf{r}_{ij}$ is the vector from the origin of the global coordinate frame to the position of the joint.



${}^{0}G$: 12 DOF

${}^{1}G$: 8 DOF

${}^{2}G$: 5 DOF

${}^{3}G$: 2 DOF

${}^{4}G$: 2 DOF

$$
{}^{4}G=\begin{bmatrix}
\begin{bmatrix}\vec{0}\\ \vec{z}\end{bmatrix} & -\rho_1\begin{bmatrix}\vec{0}\\ \vec{y}\end{bmatrix}\\
\begin{bmatrix}\vec{0}\\ \vec{z}\end{bmatrix} & \rho_2\begin{bmatrix}\vec{0}\\ \vec{x}\end{bmatrix}\\
\begin{bmatrix}\vec{0}\\ \vec{z}\end{bmatrix} & \rho_1\begin{bmatrix}\vec{0}\\ \vec{y}\end{bmatrix}\\
\begin{bmatrix}\vec{0}\\ \vec{z}\end{bmatrix} & -\rho_2\begin{bmatrix}\vec{0}\\ \vec{x}\end{bmatrix}
\end{bmatrix}
$$

${}^{0}H$: 16 OC

${}^{1}H$: 13 OC

${}^{2}H$: 10 OC

${}^{3}H$: 7 OC

${}^{4}H$: 6 OC

$$
{}^{4}\mathrm{H}=\begin{bmatrix}
\begin{bmatrix}\vec{0}\\ \vec{x}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ \vec{y}\end{bmatrix} & \begin{bmatrix}\vec{z}\\ 0\end{bmatrix} & \begin{bmatrix}\vec{0}\\ -\vec{x}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ -\vec{y}\end{bmatrix} & \begin{bmatrix}(r_{P12}-r_{P11})\\ r_{P11}\times r_{P12}\end{bmatrix}\\
\begin{bmatrix}\vec{0}\\ \vec{x}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ \vec{y}\end{bmatrix} & \begin{bmatrix}\vec{z}\\ \vec{0}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ \vec{x}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ -\vec{y}\end{bmatrix} & -\begin{bmatrix}(r_{P22}-r_{P21})\\ r_{P21}\times r_{P22}\end{bmatrix}\\
\begin{bmatrix}\vec{0}\\ \vec{x}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ \vec{y}\end{bmatrix} & \begin{bmatrix}\vec{z}\\ \vec{0}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ \vec{x}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ \vec{y}\end{bmatrix} & \begin{bmatrix}(r_{P32}-r_{P31})\\ r_{P31}\times r_{P32}\end{bmatrix}\\
\begin{bmatrix}\vec{0}\\ \vec{x}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ \vec{y}\end{bmatrix} & \begin{bmatrix}\vec{z}\\ 0\end{bmatrix} & \begin{bmatrix}\vec{0}\\ -\vec{x}\end{bmatrix} & \begin{bmatrix}\vec{0}\\ \vec{y}\end{bmatrix} & -\begin{bmatrix}(r_{P42}-r_{P41})\\ r_{P41}\times r_{P42}\end{bmatrix}
\end{bmatrix}
$$

Figure 3.15: Mobility and overconstraints of the 4RRR+8R mechanism

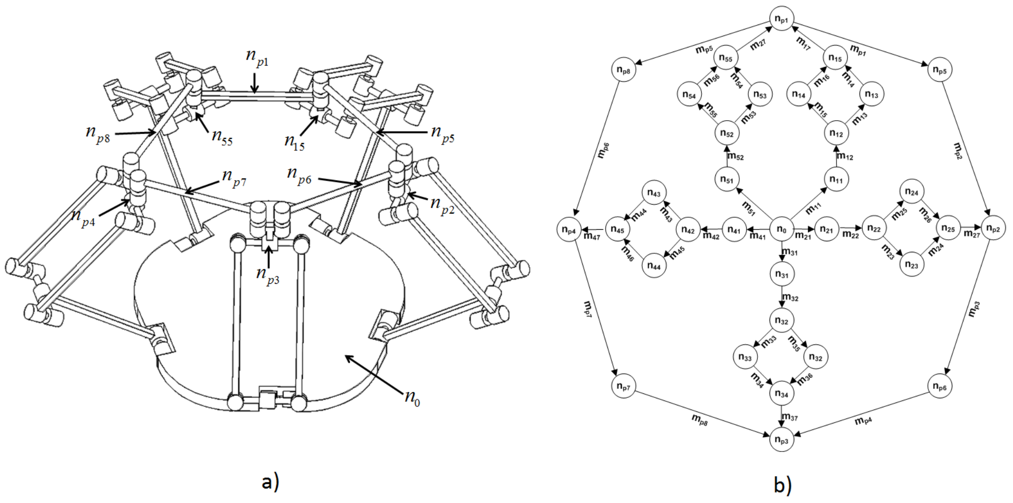


Figure 3.16: a) Shematic representation of the PentaG mechanism and b) its Graph Representation.

For numerical computation, $\mathbf{r}_{ij}$ can be any vector from the origin to any point on the row corresponding to the axis of rotation of the joint since the result of the cross product $\mathbf{r}_{ij} \times \mathbf{s}_{ij}$ is the same regardless the point on the row used to compute $\mathbf{r}_{ij}$.

The first reduction possible is a serial reduction that applies on links n_{i3} and n_{i4} of legs $i = 1..5$ and on links n_{p5}, n_{p6}, n_{p7}, and n_{p8} of the configurable platform. The graph resulting from this reduction is presented in figure 3.17 a). The screw systems obtained from those serial reductions are:

$$T_{ia} = T_{i3} + T_{i4} = \left\{ \begin{bmatrix} \hat{\mathbf{s}}_{ic} \\ \mathbf{r}_{i3} \times \hat{\mathbf{s}}_{ic} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{ic} \\ \mathbf{r}_{i4} \times \hat{\mathbf{s}}_{ic} \end{bmatrix} \right\} = \left\{ \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i4} - \mathbf{r}_{i3}) \times \hat{\mathbf{s}}_{ic} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{ic} \\ \mathbf{r}_{i4} \times \hat{\mathbf{s}}_{ic} \end{bmatrix} \right\} \quad (3.41)$$

$$T_{ib} = T_{i5} + T_{i6} = \left\{ \begin{bmatrix} \hat{\mathbf{s}}_{ic} \\ \mathbf{r}_{i5} \times \hat{\mathbf{s}}_{ic} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{ic} \\ \mathbf{r}_{i6} \times \hat{\mathbf{s}}_{ic} \end{bmatrix} \right\} = \left\{ \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i5} - \mathbf{r}_{i6}) \times \hat{\mathbf{s}}_{ic} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{ic} \\ \mathbf{r}_{i6} \times \hat{\mathbf{s}}_{ic} \end{bmatrix} \right\} \quad (3.42)$$

$$\begin{aligned} T_{pj} = T_{p,2j-1} + T_{p,2j} &= \left\{ \begin{bmatrix} \hat{\mathbf{s}}_{p} \\ \mathbf{r}_{p,2j-1} \times \hat{\mathbf{s}}_{p} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{p} \\ \mathbf{r}_{p,2j} \times \hat{\mathbf{s}}_{p} \end{bmatrix} \right\} \\ T_{pj} &= \left\{ \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{p,2j} - \mathbf{r}_{p,2j-1}) \times \hat{\mathbf{s}}_{p} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{p} \\ \mathbf{r}_{i4} \times \hat{\mathbf{s}}_{p} \end{bmatrix} \right\} \end{aligned} \quad (3.43)$$

Following this serial reduction, a parallel reduction is now possible in each parallelogram unit. The resulting graph is shown in Figure 3.17 b). In a parallel reduction, the resulting twist system represents the intersection of the original twist systems. Information concerning the overconstraints of the mechanism may be lost during the intersection procedure and it is therefore better to proceed to the addition of the wrench system. The wrench systems of the parallelogram units are given by

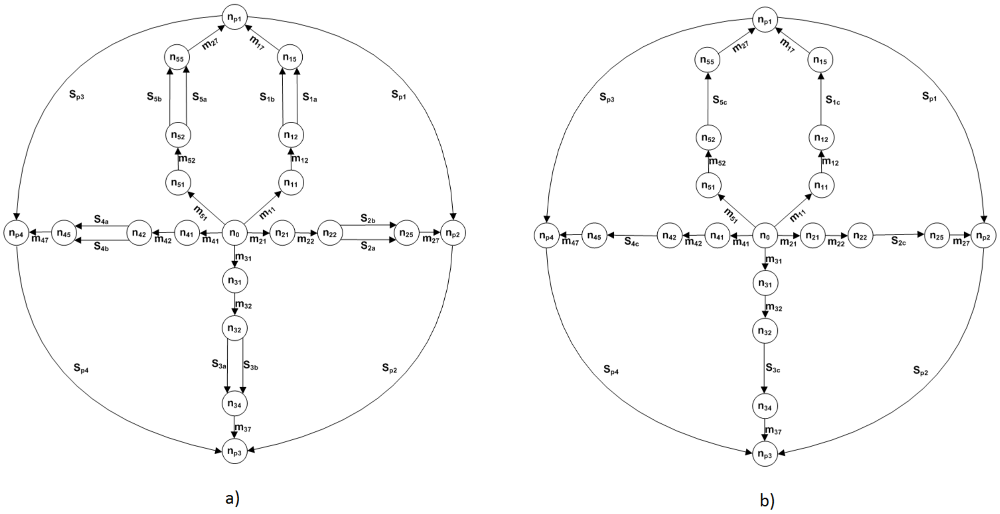


Figure 3.17: Graph representation of the PentaG mechanism after a) the first serial reduction and b) the first parallel reduction

$$W_{ia} = \left\{ \begin{bmatrix} (\mathbf{r_{i4}} - \mathbf{r_{i3}}) \\ \mathbf{r_{i3}} \times \mathbf{r_{i4}} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{ic} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r_{i4}} - \mathbf{r_{i3}}) \times \hat{\mathbf{s}}_{\mathbf{ic}} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i4} - \mathbf{r}_{i3}) \end{bmatrix} \right\} \tag{3.44}$$

$$W_{ib} = \left\{ \begin{bmatrix} (\mathbf{r_{i6}} - \mathbf{r_{i5}}) \\ \mathbf{r_{i5}} \times \mathbf{r_{i6}} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{ic} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r_{i6}} - \mathbf{r_{i5}}) \times \hat{\mathbf{s}}_{\mathbf{ic}} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i6} - \mathbf{r}_{i5}) \end{bmatrix} \right\} \tag{3.45}$$

The overconstraints can be obtained from the restricted wrench system of W_{ia} and W_{ib} as defined in Section 2.3. Since $(\mathbf{r}_{i4} - \mathbf{r}_{i3})$ and $(\mathbf{r}_{i6} - \mathbf{r}_{i5})$ are both perpendicular to $\hat{\mathbf{s}}_{ic}$ the restricted screw system is

$$W_{ic}^{R} = W_{ia} \bigcap W_{ib} = \left\{ \begin{bmatrix} \hat{\mathbf{s}}_{ic} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r_{i4}} - \mathbf{r_{i3}}) \times \hat{\mathbf{s}}_{\mathbf{ic}} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i4} - \mathbf{r}_{i3}) \end{bmatrix} \right\} \tag{3.46}$$

For that reason, there are therefore 3 overconstraints for each parallelogram units -as it is the case for most planar mechanisms- and therefore 15 overconstraints for the complete mechanism. The twist system of the parallelogram units is given by

$$T_{ic} = T_{ia} \bigcap T_{ib} = \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i4} - \mathbf{r}_{i3}) \times \hat{\mathbf{s}}_{ic} \end{bmatrix} \tag{3.47}$$

which shows that each parallelogram unit has translational DOF.

In the next graph reduction step, the twist systems of the legs are computed. The resulting graph is shown in Figure 3.18 a). The twist system T_{id} of each leg is given by

$$T_{id} = T_{i1} + T_{i2} + T_{ic} + T_{i7} = \left\{ \begin{bmatrix} \hat{\mathbf{s}}_{id} \\ \mathbf{r}_{i1} \times \hat{\mathbf{s}}_{id} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{id} \\ \mathbf{r}_{i2} \times \hat{\mathbf{s}}_{id} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i4} - \mathbf{r}_{i3}) \times \hat{\mathbf{s}}_{ic} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{id} \\ \mathbf{r}_{i7} \times \hat{\mathbf{s}}_{id} \end{bmatrix} \right\} \tag{3.48}$$

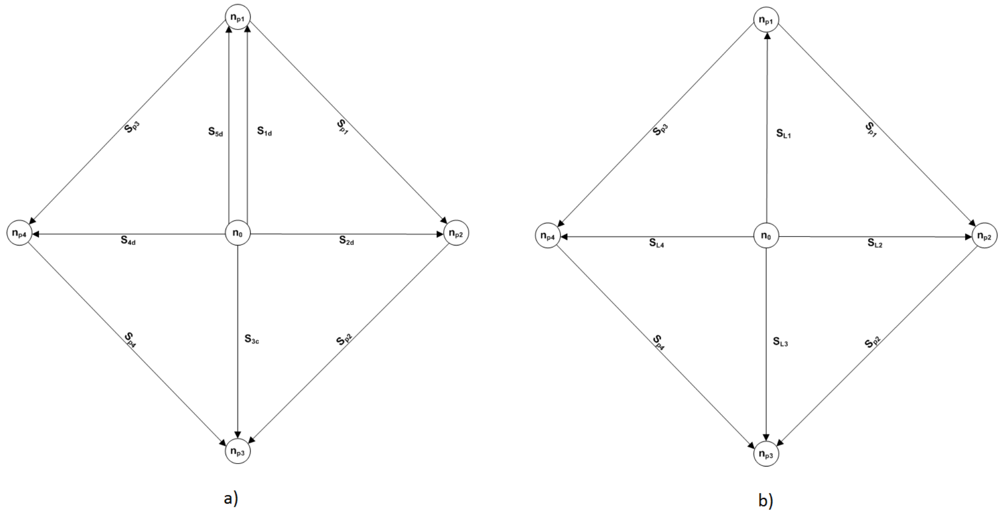


Figure 3.18: Graph representation of the PentaG mechanism after a) the second serial reduction and b) the second parallel reduction

$$T_{id} = \left\{ \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i2} - \mathbf{r}_{i1}) \times \hat{\mathbf{s}}_{id} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i7} - \mathbf{r}_{i2}) \times \hat{\mathbf{s}}_{id} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i4} - \mathbf{r}_{i3}) \times \hat{\mathbf{s}}_{ic} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{id} \\ \mathbf{r}_{i7} \times \hat{\mathbf{s}}_{id} \end{bmatrix} \right\} \tag{3.49}$$

The four twist systems don't intersect so there is no local mobility in the legs. A linear transformation of the basis screws of those system are performed Equation 3.48 to Equation 3.49 to facilitate the next computations. The last possible reduction is the parallel reduction between leg $i = 1$ and leg $i = 5$. The resulting graph is shown in Figure 3.18 b). Since this is a parallel reduction, the wrench system is first computed to reveal possible local overconstraints.

$$W_{1d} = T_{1d}^{\perp} = \left\{ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_{1c} \times \hat{\mathbf{s}}_{1d} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{14} - \mathbf{r}_{13}) \times \hat{\mathbf{s}}_{1d} \end{bmatrix} \right\} \tag{3.50}$$

$$W_{5d} = T_{5d}^{\perp} = \left\{ \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_{5c} \times \hat{\mathbf{s}}_{5d} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{54} - \mathbf{r}_{53}) \times \hat{\mathbf{s}}_{5d} \end{bmatrix} \right\} \tag{3.51}$$

The restricted system obtained represents the overconstraint of this parallel construction.

$$W_{L1}^{R} = W_{1d} \bigcap W_{5d} = \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{s}}_{1d} \times \hat{\mathbf{s}}_{5d} \end{bmatrix} \tag{3.52}$$

One overconstraint occurs when these two legs are assembled. Considering the overconstraints already revealed for the parallelogram units, the total of local overconstraints is 16. The twist system T_{L1} is given by

$$T_{L1} = T_{1d} \bigcap T_{5d} = \left\{ \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i2} - \mathbf{r}_{i1}) \times \hat{\mathbf{s}}_{id} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i7} - \mathbf{r}_{i2}) \times \hat{\mathbf{s}}_{id} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{i4} - \mathbf{r}_{i3}) \times \hat{\mathbf{s}}_{ic} \end{bmatrix} \right\} \tag{3.53}$$

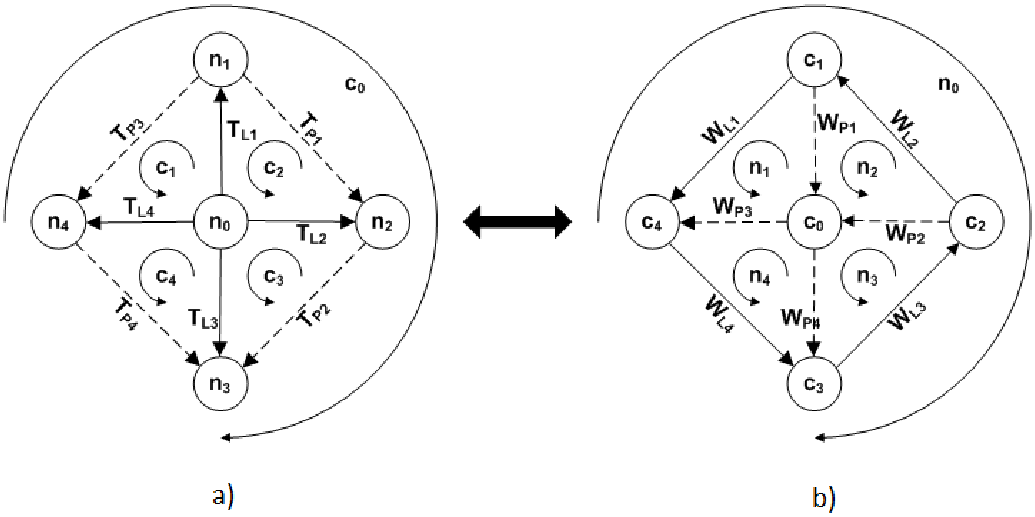


Figure 3.19: Dual a) Twist Graph and b) Wrench Graph of the PentaG

This complete the graph reduction procedure. By now, no more serial or parallel reductions are possible due to the non series-parallel nature of the PentaG mechanism and the result of the graph reduction procedure is a wheel graph with corresponding edges twist systems. No local mobilities were present in the legs or the platform. A total of 16 local overconstraints in the wheel edges has been revealed, 3 in each parallelogram unit and 1 between leg 1 and leg 5.

From the graph reduction process, a wheel graph of twist systems and a dual wheel graph of wrench systems shown in Figure 3.19 is finally obtained. A set of basis screws for each screw system is given in Table 3.4.

Legs and Platform Twist Systems
$T_{L1} = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} \right\}$
$T_{Li} = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \hat{\mathbf{s}}_{id} \\ \mathbf{r}_{i7} \times \hat{\mathbf{s}}_{id} \end{bmatrix} \right\} i = 2..4$
$T_{Pi} = \left\{ \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{p,2i} - \mathbf{r}_{p,2i-1}) \times \hat{\mathbf{s}}_p \end{bmatrix}, \begin{bmatrix} \mathbf{z} \\ \mathbf{r}_{p,2i} \times \mathbf{z} \end{bmatrix} \right\} i = 1..4$

Legs and Platform Wrench Systems
$W_{L1} = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} \right\}$
$W_{Li} = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \times \hat{\mathbf{s}}_{id} \end{bmatrix} \right\} i = 2..4$
$W_{Pi} = \left\{ \begin{bmatrix} \hat{\mathbf{z}} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} (\mathbf{r_{p,2i}} - \mathbf{r_{p,2i-1}}) \\ \mathbf{r}_{p,2i} \times \mathbf{r}_{p,2i-1} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{x}} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \hat{\mathbf{y}} \end{bmatrix} \right\} i = 1..4$

Table 3.4: Sets of basis screws for the legs and the platform screw systems of the PentaG mechanism

3.6.2.2 Mobility Analysis

The inputs of the mobility method is the wheel graph of twist systems of Figure 3.19 a). First, the initial leg matrix of mobility ${}^{t}G$ is built. This matrix has 15 columns representing the 15 DOF of the legs when the configurable platform is not connected. For each DOF, i.e for each column, a global variable λ_i can be assigned

$$ {}^{t}G = \begin{bmatrix} T_{L1} & 0 & 0 & 0 \\ 0 & T_{L2} & 0 & 0 \\ 0 & 0 & T_{L3} & 0 \\ 0 & 0 & 0 & T_{L4} \end{bmatrix}_{24,15} \tag{3.54} $$

The first platform connection $P1$ connects leg 2 to leg 1. The mobility of leg 2 relatively to leg 1 is obtained from the subtraction of row 2 and row 1 in ${}^{t}G$.

$$ {}^{t}G_{(2-1)} = [-T_{L1}\, T_{L2}\, 0\, 0]_{6,15} \tag{3.55} $$

$$ {}^{t}G_{(2-1)} = \left[\begin{bmatrix} \mathbf{0} \\ -\mathbf{x} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ -\mathbf{y} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ -\mathbf{z} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{s}}_{id} \\ \mathbf{r}_{i7} \times \hat{\mathbf{s}}_{id} \end{bmatrix} \mathbf{0}_{6,4}\, \mathbf{0}_{6,4} \right]_{6,15} \tag{3.56} $$

A change of variable is then performed so that the variables of the restricted screw systems of ${}^{t}G_{(2-1)}$ are canceled. The change of variables performed are ${}^{t}\lambda^{\backprime}_1 = {}^{t}\lambda_1 + {}^{t}\lambda_4$, ${}^{t}\lambda^{\backprime}_2 = {}^{t}\lambda_2 + {}^{t}\lambda_5$ and ${}^{t}\lambda^{\backprime}_3 = {}^{t}\lambda_3 + {}^{t}\lambda_6$, leading to $S_1^R = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix}, \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} \right\}$ and $S_1^U = \{\emptyset\}$. The new generative matrix ${}^{t}G^{\backprime}$ is

$$ {}^{t}G' = \left[\begin{array}{ccc|cccccc} \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} & 0 & 0 & 0 & 0 & 0 & 0 \\ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} & \begin{bmatrix} \hat{\mathbf{s}}_{id} \\ \mathbf{r}_{i7} \times \hat{\mathbf{s}}_{id} \end{bmatrix} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & T_{L3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & T_{L4} \end{array} \right]_{24,15} \tag{3.57} $$

The second step is to compute the intersection of the platform segment $P1$ with the unrestricted 4 DOF twist system between leg 2 and leg 1.

$$ T_{P1}^{U} = T_{P1} \bigcap \left(S_1^U + T_{L2} \right) = T_{P1} \bigcap T_{L2} = \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{p2} - \mathbf{r}_{p1}) \times \hat{\mathbf{s}}_p \end{bmatrix} \tag{3.58} $$

The leg mobility matrix obtained after the connection of platform segment $P1$ is

$$ {}^{1}G = \left[\begin{array}{cccccc} \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{y} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} & \begin{bmatrix} \mathbf{0} \\ (\mathbf{r}_{p2} - \mathbf{r}_{p1}) \times \hat{\mathbf{s}}_p \end{bmatrix} & \mathbf{0} & \mathbf{0} \\ \multicolumn{3}{c}{\mathbf{0}} & \mathbf{0} & T_{L3} & \mathbf{0} \\ \multicolumn{3}{c}{\mathbf{0}} & \mathbf{0} & \mathbf{0} & T_{L4} \end{array} \right]_{6,12} \tag{3.59} $$

The procedure to connected platform edge $P2$ and $P3$ is very similar to the procedure to connect P_1. The generative matrix after connection of platform edge $P1$, $P2$ and $P3$ is

$$
{}^{3}G = \begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\(\mathbf{r_{p2}}-\mathbf{r_{p1}})\times\hat{\mathbf{s}}_p\end{bmatrix} & \mathbf{0} & \mathbf{0} \\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p2}-\mathbf{r}_{p1})\times\hat{\mathbf{s}}_p\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\(\mathbf{r_{p4}}-\mathbf{r_{p3}})\times\hat{\mathbf{s}}_p\end{bmatrix} & \mathbf{0} \\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p2}-\mathbf{r}_{p1})\times\hat{\mathbf{s}}_p\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p4}-\mathbf{r}_{p3})\times\hat{\mathbf{s}}_p\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\(\mathbf{r_{p6}}-\mathbf{r_{p5}})\times\hat{\mathbf{s}}_\mathbf{p}\end{bmatrix}
\end{bmatrix}_{6,6} \tag{3.60}
$$

The last platform edge to close is P_4. The twist system of P_4 must be the same than $L_4 - L_1$.

$$
{}^{3}G_{(4-1)} = \begin{bmatrix}\mathbf{0}_{6,3} \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p2}-\mathbf{r}_{p1})\times\hat{\mathbf{s}}_p\end{bmatrix} \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p4}-\mathbf{r}_{p3})\times\hat{\mathbf{s}}_p\end{bmatrix} \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p6}-\mathbf{r}_{p5})\times\hat{\mathbf{s}}_p\end{bmatrix}\end{bmatrix}_{6,15} \tag{3.61}
$$

Since all vectors $\mathbf{r_{pi}}$ are in the same plane, the 3 screws representing the motion of leg 1 relatively to leg 4 are not independent. In order to separate the restricted and unrestricted screw systems, a change of variable ${}'\lambda_4 = \lambda_4 - \rho_1\lambda_5 - \rho_2\lambda_6$ is applied where ρ_1 and ρ_2 are defined such that

$$
\begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p2}-\mathbf{r}_{p1})\times\hat{\mathbf{s}}_p\end{bmatrix} = \rho_1 \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p4}-\mathbf{r}_{p3})\times\hat{\mathbf{s}}_p\end{bmatrix} + \rho 2 \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p6}-\mathbf{r}_{p5})\times\hat{\mathbf{s}}_p\end{bmatrix} \tag{3.62}
$$

The leg mobility matrix becomes

$$
{}^{3}G'_{(4-1)} = \begin{bmatrix}\mathbf{0}_{6,3}\, 0 \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p4}-\mathbf{r}_{p3})\times\hat{\mathbf{s}}_p\end{bmatrix} \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p6}-\mathbf{r}_{p5})\times\hat{\mathbf{s}}_p\end{bmatrix}\end{bmatrix}_{6,15} \tag{3.63}
$$

$$
{}^{3}G' = \left[\begin{array}{cccc|cc}
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\(\mathbf{r_{p2}}-\mathbf{r_{p1}})\times\hat{\mathbf{s}}_p\end{bmatrix} & \mathbf{0} & \mathbf{0} \\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & \rho 2\begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p6}-\mathbf{r}_{p5})\times\hat{\mathbf{s}}_p\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\(\mathbf{r_{p4}}-\mathbf{r_{p3}})\times\hat{\mathbf{s}}_p\end{bmatrix} & \mathbf{0} \\
\begin{bmatrix}\mathbf{0}\\\mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\\mathbf{z}\end{bmatrix} & 0 & \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p4}-\mathbf{r}_{p3})\times\hat{\mathbf{s}}_p\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\(\mathbf{r_{p6}}-\mathbf{r_{p5}})\times\hat{\mathbf{s}}_\mathbf{p}\end{bmatrix}
\end{array}\right]_{6,6} \tag{3.64}
$$

Finally, the intersection of the twist system of the last platform segment $P4$ with the unrestricted screw system of Equation 3.63 is performed. The 2 last columns combine into a single screw representing the motion of the platform segment $P4$. The change of variable is ${}^{4}\lambda_5 = \rho_3\lambda_5 + \rho_4\lambda_6$ with ρ_3 and ρ_4 such that

$$
\begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p8}-\mathbf{r}_{p7})\times\hat{\mathbf{s}}_p\end{bmatrix} = \rho_3 \begin{bmatrix}\mathbf{0}\\(\mathbf{r}_{p4}-\mathbf{r}_{p3})\times\hat{\mathbf{s}}_p\end{bmatrix} + \rho_4 \begin{bmatrix}\mathbf{0}\\(\mathbf{r_{p6}}-\mathbf{r_{p5}})\times\hat{\mathbf{s}}_\mathbf{p}\end{bmatrix} \tag{3.65}
$$

The final matrix is then

$$
{}^{4}G = \begin{bmatrix}
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix} & \mathbf{0} & \mathbf{0}\\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ (\mathbf{r_{p2}} - \mathbf{r_{p1}}) \times \hat{\mathbf{s}}_p\end{bmatrix} & \mathbf{0}\\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix} & \rho 2 \begin{bmatrix}\mathbf{0}\\ (\mathbf{r}_{p6} - \mathbf{r}_{p5}) \times \hat{\mathbf{s}}_p\end{bmatrix} & \rho_3 \begin{bmatrix}\mathbf{0}\\ (\mathbf{r_{p4}} - \mathbf{r_{p3}}) \times \hat{\mathbf{s}}_p\end{bmatrix}\\
\begin{bmatrix}\mathbf{0}\\ \mathbf{x}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{y}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix} & \mathbf{0} & \begin{bmatrix}\mathbf{0}\\ (\mathbf{r}_{p8} - \mathbf{r}_{p7}) \times \hat{\mathbf{s}}_p\end{bmatrix}
\end{bmatrix}_{6,5} \tag{3.66}
$$

This matrix shows that the PentaG mechanism has 5 DOF. The first three columns correspond to the mobility of the legs when the platform is rigid. The fourth column is the motion of platform segment $P2$ and the fifth column is the motion of platform segment $P4$. Each row represents the mobility of the legs relatively to the base.

3.6.2.3 Overconstraints Analysis

Because of the duality between mobilities and overconstraints analysis, the same method is applied to compute the overconstraints. The inputs of the method is the dual graph of wrench systems shown in Figure 3.19 b). The first step is to construct the initial platform wrench matrix ${}^{t}H$

$$
{}^{t}H = \begin{bmatrix}
W_{P1} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & W_{P2} & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & W_{P3} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & W_{P4}
\end{bmatrix}_{24,16} \tag{3.67}
$$

Mathematically, the overconstraint analysis of the PentaG is identical to the mobility analysis presented in section 3.6.2.2. The initial matrix corresponds to the overconstraints of the mechanism when all leg joints are locked. The three steps method is applied to each leg to unlock them. The first step is to compute the total forces exerted by $P2$ and $P1$ using a subtraction of row 1 to row 2 in 3.67.

$$
{}^{t}H_{(2-1)} = \begin{bmatrix} - \begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} - \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} - \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} - \begin{bmatrix}(\mathbf{r_{p2}} - \mathbf{r_{p1}})\\ \mathbf{r}_{p2} \times \mathbf{r}_{p1}\end{bmatrix} \begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} \begin{bmatrix}(\mathbf{r_{p4}} - \mathbf{r_{p3}})\\ \mathbf{r}_{p4} \times \mathbf{r}_{p3}\end{bmatrix} \mathbf{0}\,\mathbf{0} \end{bmatrix}_{24,16} \tag{3.68}
$$

A change of variable is performed so that the dependent screws in ${}^{t}H_{(2-1)}$ vanish. The new matrices are

$$
{}^{t}H'_{(2-1)} = \begin{bmatrix} \mathbf{0}\mathbf{0}\mathbf{0} - \begin{bmatrix}(\mathbf{r_{p2}} - \mathbf{r_{p1}})\\ \mathbf{r}_{p2} \times \mathbf{r}_{p1}\end{bmatrix} \begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} \begin{bmatrix}(\mathbf{r_{p4}} - \mathbf{r_{p3}})\\ \mathbf{r}_{p4} \times \mathbf{r}_{p3}\end{bmatrix} \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} \mathbf{0}\,\mathbf{0} \end{bmatrix}_{24,16} \tag{3.69}
$$

$$
{}^{t}H' = \left[\begin{array}{ccc|ccccccc}
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}(\mathbf{r_{p2}} - \mathbf{r_{p1}})\\ \mathbf{r}_{p2} \times \mathbf{r}_{p1}\end{bmatrix} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \mathbf{0} & \begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}(\mathbf{r_{p4}} - \mathbf{r_{p3}})\\ \mathbf{r}_{p4} \times \mathbf{r}_{p3}\end{bmatrix} & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & W_{P3} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & W_{P4}
\end{array}\right]_{24,16} \tag{3.70}
$$

The restricted screw system on the left of ${}^{t}H'$ corresponds to the wrenches that can be transmitted directly from $P1$ to $P2$. When leg $L2$ is unlocked, the unrestricted screw system on the right part of ${}^{t}H'$ must intersect with the wrench system of $L2$. The new matrix ${}^{2}H$ is

$$
{}^{2}H = \begin{bmatrix}
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & W_{P3} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & W_{P4}
\end{bmatrix}_{24,12} \tag{3.71}
$$

Unlocking leg $L3$ and $L4$ leads to similar operations. The platform wrench matrix ${}^{234}H$ obtained after those operations is

$$
{}^{234}H = \begin{bmatrix}
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} & \mathbf{0} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{3d}\end{bmatrix} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{3d}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{4d}\end{bmatrix}
\end{bmatrix}_{24,6} \tag{3.72}
$$

Leg $L1$ is connecting platform segment $P4$ to $P1$. Subtraction of row 4 from row 1 gives

$$
{}^{234}H_{(1-4)} = \begin{bmatrix}\mathbf{0}\mathbf{0}\mathbf{0} - \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} - \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{3d}\end{bmatrix} - \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{4d}\end{bmatrix}\end{bmatrix} \tag{3.73}
$$

After a change of variables so that columns 4 of ${}^{234}H_{(1-4)}$ vanish, the matrix ${}^{234}H'$ becomes

$$
{}^{234}H' = \left[\begin{array}{cccc|cc}
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} & \mathbf{0} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \rho_2\begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{4d}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{3d}\end{bmatrix} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \mathbf{0} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{3d}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{4d}\end{bmatrix}
\end{array}\right]_{24,6} \tag{3.74}
$$

where the scalar ρ_2 is defined such that the equation

$$
\begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} = \rho_1\begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{3d}\end{bmatrix} + \rho_2\begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{3d}\end{bmatrix} \tag{3.75}
$$

is respected.

The last two columns of ${}^{234}H'_{(1-4)}$ are included in the leg constraints W_{L1}, so no change occur in the overconstraints of the mechanism when the last leg $L1$ is unlocked. This means that the final platform constraint matrix ${}^{1234}H$ is the same as ${}^{234}H$ or ${}^{234}H'$ if the change of variables is kept.

$$
{}^{1234}H = {}^{234}H = \begin{bmatrix}
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} & \mathbf{0} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{3d}\end{bmatrix} & \mathbf{0}\\
\begin{bmatrix}\hat{\mathbf{z}}\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{x}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{y}}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{2d}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{3d}\end{bmatrix} & \begin{bmatrix}\mathbf{0}\\ \mathbf{z}\times\hat{\mathbf{s}}_{4d}\end{bmatrix}
\end{bmatrix}_{24,6} \tag{3.76}
$$

Physically, the first 3 columns of ${}^{1234}H$ represent the planar overconstraints of the planar configurable platform, which occur in any planar linkage. The last three columns show that the base joints of all legs must be in the same plane. During the graph reduction operation, 15 planar local overconstraints due to the parallelogram units and 1 local overconstraint due to the alignment of base joints in leg 1 and 5 were revealed. No local mobilities were found. The total of overconstraints for the PentaG mechanism is therefore 15+1+6=22. The mobility of this mechanism was computed in Section 3.6.2.2 and it was found that the number of DOF is 5. In total, the PentaG mechanism has 34 rigid links and 43 revolute joints. Those numbers respect the modified Chebychev–Grübler–Kutzbach criterion for quick mobility calculation presented in Section 2.2.1.

$$M = 6\,(n - m - 1) + \sum_{i=1}^{m}(f_i) + R_C - R_M = 6\,(34 - 43 - 1) + 43 + 22 - 0 = 5 \tag{3.77}$$

3.7 Summary

In this chapter, the concept of series-parallel mechanism was introduced and it was shown that the analysis method for pure parallel mechanisms presented in Chapter 2 can be extended for the analysis of any general series-parallel mechanism, applying the serial and parallel laws for twist and wrench systems. The concept of series-parallel mechanisms was defined based on the properties of their associated graph. A variety of already known classes of mechanisms, such as Delta robots and hybrid mechanisms, fall into this more general category of mechanisms and their different layers of serial and parallel connections have been presented within this new general framework using the concept of graph reduction. For each class, the standard method of mobility and overconstraints analysis can be extended using this framework as long as the mechanism is series-parallel. General recursive formulas and methods have then been presented that are applicable to any series-parallel mechanism.

It was seen that the series-parallel reduction of a parallel mechanism with configurable platform (PMCP), as defined in this thesis, always results in a wheel graph. Since wheel graphs fall into the category of non series-parallel graphs, the mobilities and overconstraints analysis of PMCPs cannot be computed with the extended standard method for series-parallel mechanisms and different methods must be used. A general method was proposed to perform the analysis of the mobility and overconstraints of parallel mechanisms with configurable platforms (PMCPs). One key aspect of the new proposed method is based on the concept of restricted and unrestricted screw systems that were used in section 2.3 to compute the local mobility of serial chains and the overconstraints of pure parallel mechanisms. A physical interpretation is provided for each step of the method.

The method underlines the separation between the global variables that are associated with the rigid motion of the platform and the global variables associated with the configuration of the platform. In the mobility analysis of PMCPs, the restricted screw system between two legs corresponds to the mobility of the legs relatively to the base if they are connected with a rigid platform. The unrestricted screw system corresponds to their relative mobility when they are not connected. Using an appropriated change of variables, the constraints imposed by the configurable platform on unrestricted the screw systems of the legs is computed. The process is repeated for each platform connection and a matrix representing the velocity of each leg as function of the global DOF of the mechanism is finally obtained.

Wheel graphs have the particular property of being self-dual and unlike most of other classes of mechanisms, PMCPs have a self dual topology. This is due to the fact that pure parallel mechanisms are dual to a single closed loop and vice versa. Combining a single closed-loop and a parallel mechanism together results in a self dual mechanism. The particular self dual topology of PMCPs can be exploited in their analysis. General methods developed for calculating the mobility of PMCPs can be reused directly to calculate their overconstraints and vice versa. The proposed mobility analysis method for PMCPs was then applied directly to the analysis of overconstraints of PMCPs, and the dual physical interpretation was presented.

The general method was then applied to two different parallel mechanisms with configurable platforms and an analytical solution and an interpretation of the complete distribution of their mobility and overconstraints was obtained. From the method, it was shown that the 4-RRR+8R mechanism has 2 DOF and 6 overconstraints while the PentaG has 5 DOF and 22 overconstraints. Those numbers are in agreement with the modified Chebychev–Grübler–Kutzbach criterion.

Very few parallel mechanisms with configurable platforms have been presented in the literature or used in the industry. Such mechanisms are ignored by type synthesis methods of mechanisms such as in [42]. They are promising since they offer alternative kinematic solutions for existing robots and can also bring new possibilities in robot architecture design, as it is the case for the PentaG, which offer grasping capabilities while all actuators are located on the base. A better understanding of the fundamentals of their mobility and overconstraints, as presented in this chapter, will help robot designers to consider them as a valid option in their choice of a robot architecture. Fundamentals of mobility and overconstraints analysis of PMCPs is an essential aspect in the future development of a broader type synthesis method that includes those mechanisms.

Part II

Parallel Robots with Configurable Platforms

Chapter 4

Kinematics of the PentaG Robot

This chapter presents the kinematics analysis of the PentaG robot, as a logical step after the design of the PentaG mechanism. The *International Federation for the Promotion of Mechanism and Machine Science* (Iftomm) has defined kinematics to be “the branch of theoretical mechanics dealing with the geometry of motion, irrespective of the causes that produce the motion”. For each configuration, the motion of the end-effector of a robot corresponds to certain motion of the motors controlling the robot. The robot motion can be expressed either in the joint space or in the end-effector Cartesian space and the kinematic mapping describes the relation between these two spaces. In robot kinematics, the links of the robot are modeled as perfectly rigid bodies so their movement can be analyzed with the use of geometry.

The position kinematic relations are highly non-linear while the velocity kinematic relations can be linearised locally for a given configuration. Furthermore, the direct kinematics problem, which consists of mapping the joint motion into the Cartesian motion, must be distinguished from the inverse kinematic problem, which consists of mapping the end-effector motion into the joint motion. These two problems are very different by nature and complexity. For a serial robot, the *direct position kinematics* (DPK) is relatively simple and has a single solution, while the *inverse position kinematics* (IPK) is more complex and has in general multiple solutions. For parallel robots, the inverse position kinematic problem is of equal complexity than for serial robots, and the direct position kinematic problem is even more complex and has challenged kinematicians in the last decades. Direct position kinematics is one of the most active fields of research in parallel robotics and there are many parallel robots for which no analytical solutions for the DPK have been found yet and for which numerical solutions must be used, despite their limitations.

The velocity mapping from the joint space to the Cartesian space can be expressed locally in a linear way by a the *Jacobian* matrix. The value of the Jacobian is generally dependent on the configuration of the robot, with the exception of the isotropic robots, which have the remarkable property of a constant value Jacobian for any configuration over the workspace. As it will be shown in this chapter, many configuration-dependent properties of the robot, for instance accuracy, static, dexterity and stiffness, can be derived from the value of the Jacobian matrix in the corresponding configuration. An important concept in velocity kinematics is the presence of singular configurations. For those particular configurations, the Jacobian matrix is not of full rank and the end-effector will have

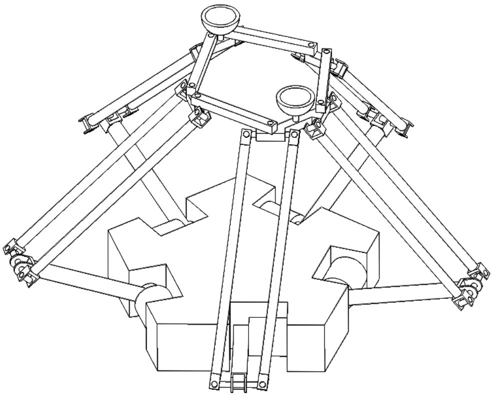


Figure 4.1: A schematic representation of the PentaG robot: A 5 DOF parallel robot with a configurable platform

uncontrollable degrees of freedom. This subject has also been extensively studied by kinematicians over the last two decades, especially for parallel robots, since parallel robot may possess a type of singularity that is not encountered in serial robots and has been called parallel singularities. Many open questions remain about singularity analysis of parallel robots and a general method to describe the degeneracy of the Jacobian, i.e. the singular configuration, as a function of the design parameters of the robot was never been fully addressed.

Aside from revealing the specific kinematics of the PentaG robot, an original contribution of this chapter is the modification of the classical method of kinematic analysis of parallel robot to allow the analysis of parallel robots with configurable platforms. In first place, the inverse and direct position kinematic relations are described followed by the derivation of the inverse Jacobian. The problem of static balancing is then addressed. Section 4.4 deals with the serial and parallel singularities of the PentaG. New types of singularities, related to the configurable platform and absent in classical robot are analyzed.

4.1 Position Kinematics

Direct position kinematics is concerned with the mapping of the joint positions into end-effector positions while inverse position kinematics is concerned with the mapping of end-effector position into joint position. On the contrary of serial robots, for which the direct position kinematic is easier to analyse than the inverse kinematics, the direct kinematic analysis of parallel robot is much more complex than its inverse kinematics. For parallel robots, both direct and inverse position kinematic mapping have in general multiple solutions. The multiple solutions of the inverse kinematics are called *working modes* while the multiple solutions of the direct kinematics are called *assembly modes.*

4.1.1 Inverse Position Kinematics

The inverse position kinematics problem is to establish the values of the position of the actuators as function of the position of the end-effector. This problem is very similar to the IPK problem encountered in serial robots. For a serial robot, one is interested in the value of each joint of the serial chain as a function of the end-effector position. A parallel mechanism can be considered as a collection of serial chains that share the same end-effector. In order to calculate the IPK of a parallel robot, the IPK of each leg can be calculated independently. The IPK solutions for serial robots can therefore almost be used directly for the IPK of parallel robots.

Finding an analytical expression of the IPK of a general serial chain has been the subject of many researches in the beginning 90s [45, 44, 50, 67]. One significant advance on this problem was to show that the IPK of a general 6 joints serial chain has at most 16 solutions corresponding to the roots of a polynomial of order 16[66]. In practice, some geometrical conditions must be satisfied so that all the solutions are real. For many robots, some of the solutions to the polynomial are always imaginary and the IPK problem of these serial robots has less than 16 real solutions. In some industrial serial robot, iterative methods based on numerical techniques are used to compute the IPK in real-time. In this section, the analytical solutions of the IPK of the PentaG are presented based on geometrical arguments. Analytical solutions have the advantage of being computer efficient for the real-time control of the robot. The problem is to find the following relation f:

$$\begin{aligned} \mathbf{q} &= f(\chi) \\ \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \end{bmatrix} &= f\left(\begin{bmatrix} x \\ y \\ z \\ \theta \\ \rho \end{bmatrix}\right) \end{aligned} \tag{4.1}$$

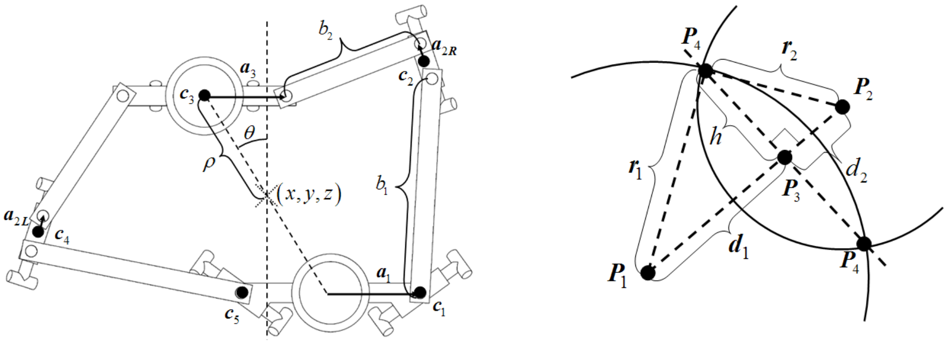
where q_i is the angle of the actuator i with the horizontal plane, χ is the position of the end-effector, and f represent the non-linear relation of the IPK. The IPK of the PentaG differs from the IPK of classical parallel robots due to the configurable platform. Because of this platform, the problem can be divided into two sub-problems. The first is the computation of the IPK of the platform, which consist of finding the position of the leg attach point $\mathbf{c_i}$ of leg i as function of the end-effector position χ, and the second is the computation of the IPK for the actuator position q_i as function of the leg attach points $\mathbf{c_i}$.

$$\begin{aligned} \mathbf{c_i} &= f_p(\chi) \\ q_i &= f_l(\mathbf{c_i}) \end{aligned} \; i = 1..5 \tag{4.2}$$

The techniques used here are based on simple geometric calculations. It will be shown that both the IPK of the platform and the IPK of the legs have multiple solutions. Each solution corresponds to a working mode. Robots are normally designed to operate only in one working mode. The solution corresponding to the working mode used for the PentaG will be presented in detail. The other working modes are briefly discussed in Section 4.1.1.3.

4.1.1.1 Inverse Position Kinematic of the Configurable Platform

The IPK of the configurable platform corresponds to the non-linear function f_p that maps the 5-dimensional vector χ, which corresponds to the position, orientation and grasping of the two finger tips, into five 3-dimensional vectors $\mathbf{c_i} = [x_i \, y_i \, z_i]^T$, each of



a) b)

Figure 4.2: Inverse Position Kinematic of the configurable platform. The position of the leg attach points c_2 and c_4 in a) corresponds to the solution of the intersection of two circles in b).

which corresponding to the position of the attach point of leg i. Since $z - z_i$ is constant for each leg i, the problem is reduced to a 2D geometric problem in the horizontal plane. Figure 4.1 shows a schematic representation of the complete 5 DOF PentaG robot. Figure 4.2 a) shows the position of the end-effector $\chi = [x\,y\,z\,\theta\,\rho]^T$, the attach points $\mathbf{c_i}$ and the geometric parameters used in the IPK of the configurable platform. By the construction of the robot, the last link of each leg is constrained in rotation around the vertical axis. The vector $\mathbf{a}_1$, $\mathbf{a}_{2R}$, $\mathbf{a}_{2L}$ and $\mathbf{a}_3$, which determine the position of the platform joints relatively to the leg attach points $\mathbf{c_i}$ are therefore constant for any configuration. Since those vectors are constant, the position of the leg attach point of leg #1, #3 and #5 are easily computed as

$$\mathbf{c}_1 = \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \rho \begin{bmatrix} sin\,(\theta) \\ -cos\,(\theta) \\ -h \end{bmatrix} + \mathbf{a}_1 \tag{4.3}$$

$$\mathbf{c}_3 = \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \rho \begin{bmatrix} -sin\,(\theta) \\ cos\,(\theta) \\ -h \end{bmatrix} \tag{4.4}$$

$$\mathbf{c}_5 = \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \rho \begin{bmatrix} sin\,(\theta) \\ -cos\,(\theta) \\ -h \end{bmatrix} - \mathbf{a}_1 \tag{4.5}$$

For the attach points of legs #2 and #4, the intersections of the platform links need to be computed. The lengths of the platform links are symmetric on each side of the finger tips. The positions $\mathbf{c_2}$ and $\mathbf{c_4}$ are located at the intersection of two circles with radius b_1 and b_2. The closure equations for the attach point $\mathbf{c_2}$ are given by

$$\begin{aligned} \|\mathbf{c}_2 - \mathbf{a}_{2R} - \mathbf{c}_1\| &= b_1 \\ \|\mathbf{c}_3 + \mathbf{a}_3 - \mathbf{c}_2 - \mathbf{a}_{2R}\| &= b_2 \end{aligned} \tag{4.6}$$

And the closure equations for the attach point $\mathbf{c_4}$ are given by

$$
\begin{aligned}
\|\mathbf{c}_4 - \mathbf{a}_{2L} - \mathbf{c}_5\| &= b_1 \\
\|\mathbf{c}_3 - \mathbf{a}_3 - \mathbf{c}_4 - \mathbf{a}_{2L}\| &= b_2
\end{aligned}
\tag{4.7}
$$

The point $\mathbf{c_2}$ corresponds to the intersection point of a circle centered at $(\mathbf{c}_1 + \mathbf{a}_{2R})$ with a radius b_1 and a circle centered at $(\mathbf{c}_3 + \mathbf{a}_3 - \mathbf{a}_{2R})$ with a radius b_2. The point $\mathbf{c}_4$ can be computed in a similar way and corresponds to the intersection point of a circle centered at $(\mathbf{c}_5 - \mathbf{a}_{2L})$ with radius b_1 and a circle centered at $(\mathbf{c}_3 - \mathbf{a}_3 - \mathbf{a}_{2L})$ with radius b_2. Figure 4.2 b) shows the intersection points of two circles centered in $\mathbf{p}_1$ and $\mathbf{p}_2$ with radius r_1 and r_2 respectively. If $|r_1 - r_2| < \|\mathbf{p}_2 - \mathbf{p}_1\| < (r_1 + r_2)$, intersection of circles has two solutions which correspond to two different working modes for the configurable platform.

The intersection points are computed in the following way. Considering two right triangles $\mathbf{p}_1\mathbf{p}_3\mathbf{p}_4$ and $\mathbf{p}_2\mathbf{p}_3\mathbf{p}_4$. one can write

$$
\begin{aligned}
d_1^2 + h^2 &= r_1^2 \\
d_2^2 + h^2 &= r_2^2 \\
d_1 + d_2 &= \|\mathbf{p}_2 - \mathbf{p}_1\|
\end{aligned}
\tag{4.8}
$$

to obtain a system of three non linear equations and three unknowns. Solving for d_1, d_2 and h gives

$$
\begin{aligned}
d_1 &= \tfrac{\|\mathbf{p}_2 - \mathbf{p}_1\|^2 + r_1^2 - r_2^2}{2\|\mathbf{p}_2 - \mathbf{p}_1\|} \\
d_2 &= \tfrac{\|\mathbf{p}_2 - \mathbf{p}_1\|^2 + r_2^2 - r_1^2}{2\|\mathbf{p}_2 - \mathbf{p}_1\|} \\
h &= \sqrt{r_1^2 - d_1^2}
\end{aligned}
\tag{4.9}
$$

The points $\mathbf{p}_4$ are finally given by

$$
\mathbf{p}_4 = \mathbf{p}_1 + \begin{bmatrix} d_1 & \pm h \\ \mp h & d_1 \end{bmatrix} \frac{(\mathbf{p}_2 - \mathbf{p}_1)}{\|\mathbf{p}_2 - \mathbf{p}_1\|}
\tag{4.10}
$$

The plus-minus signs reflect the fact that the problem has two solutions. To compute $\mathbf{c_2}$ and $\mathbf{c_4}$, the terms in equation 4.10 should be replaced by the terms of the closures equations 4.6 and 4.7. The solution corresponding to the working mode used for the platform is then selected. The positions of the leg attach points $\mathbf{c_2}$ and $\mathbf{c_4}$are finally given by

$$
\mathbf{c}_2 = (\mathbf{c}_1 + \mathbf{a}_{2R}) + \begin{bmatrix} d_2 & h_2 \\ -h_2 & d_2 \end{bmatrix} \frac{(\mathbf{c}_3 + \mathbf{a}_3 - 2\mathbf{a}_{2R} - \mathbf{c}_1)}{\|\mathbf{c}_3 + \mathbf{a}_3 - 2\mathbf{a}_{2R} - \mathbf{c}_1\|}
\tag{4.11}
$$

$$
d_2 = \frac{\|\mathbf{c}_3 + \mathbf{a}_3 - 2\mathbf{a}_{2R} - \mathbf{c}_1\|^2 + b_1^2 - b_2^2}{2\,\|\mathbf{c}_3 + \mathbf{a}_3 - 2\mathbf{a}_{2R} - \mathbf{c}_1\|}
\tag{4.12}
$$

$$
h_2 = \sqrt{b_1^2 - d_2^2}
\tag{4.13}
$$

and

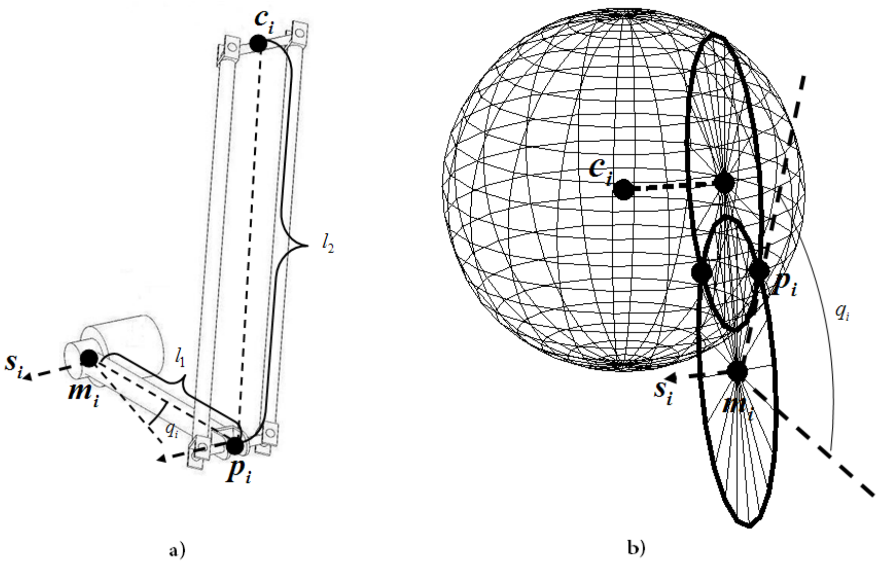


Figure 4.3: The IPK solutions of the legs using geometrical intersections. The position of the leg attach point $\mathbf{c}_i$ is given by the solution of the IPK of the configurable platform. The axis of rotation $\mathbf{s}_i$ of the grounded actuator is also known. The two possible solutions correspond to the intersection points of a sphere with a circle.

$$\mathbf{c}_4 = (\mathbf{c}_5 + \mathbf{a}_{2L}) + \begin{bmatrix} d_4 & -h_4 \\ h_4 & d_4 \end{bmatrix} \frac{(\mathbf{c}_3 - \mathbf{a}_3 - 2\mathbf{a}_{2L} - \mathbf{c}_5)}{\|\mathbf{c}_3 - \mathbf{a}_3 - 2\mathbf{a}_{2L} - \mathbf{c}_5\|} \tag{4.14}$$

$$d_4 = \frac{\|\mathbf{c}_3 - \mathbf{a}_3 - 2\mathbf{a}_{2L} - \mathbf{c}_5\|^2 + b_1^2 - b_2^2}{2\,\|\mathbf{c}_3 - \mathbf{a}_3 - 2\mathbf{a}_{2L} - \mathbf{c}_5\|} \tag{4.15}$$

$$h_4 = \sqrt{b_1^2 - d_4^2} \tag{4.16}$$

4.1.1.2 Leg Inverse Kinematics

In the previous section, the position of the five leg attach points $\mathbf{c}_i$ was described as function of the end-effector position χ. The second step of the IPK problem is to find the angle of the ground-based actuators as function of the leg attach points. The results of the first equation in 4.2 are used as input for the second equation in 4.2. Each leg has the same structure, so the same type of solution applies to each leg. Figure 4.3 a) shows a schematic representation of a leg. In order to compute q_i, the actuator position, the position of the leg mid-point $\mathbf{p}_i$ needs to be computed. The actuator position $\mathbf{m}_i$ is constant for any configuration and the leg attach point $\mathbf{c}_i$ is computed from the IPK of the platform. From the structure of the leg, one can see that the mid-point $\mathbf{p}_i$ must be located on the surface of a sphere centered in $\mathbf{c}_i$, and on a circle that is centered on $\mathbf{m}_i$ and perpendicular to the axis of rotation of the actuator $\mathbf{s}_i$, as shown in Figure 4.3 b).

To compute the sphere-circle intersection points, the leg attach point $\mathbf{c}_i$ must first be expressed in a reference frame that is centered on $\mathbf{m}_i$, with the Z axis along the vertical and the X perpendicular to $\mathbf{s}_i$. The leg attach point $\mathbf{f}_i$, which corresponds to the leg attach point $\mathbf{c}_i$ expressed in the new reference frame can be expressed as

$$\mathbf{f}_i = \begin{bmatrix} cos\,(\phi - \pi) & -sin\,(\phi - \pi) & 0 \\ sin\,(\phi - \pi) & cos\,(\phi - \pi) & 0 \\ 0 & 0 & 1 \end{bmatrix} (\mathbf{c}_i - \mathbf{m}_i) \tag{4.17}$$

where ϕ is the angle between $\mathbf{s}_i$ and the Y axis of the global reference frame. In this new reference frame, the mid-point $\mathbf{p}_i$ is given by the intersection of two circles in the XZ plane. The centre and radius of the second circle are given by $[f_{ix}\,0\,f_{iz}]$ and $\sqrt{l_2^2 - f_{iy}^2}$ respectively. The circle-circle intersection Equation 4.10 is then used to compute the intersection point in the XZ plane and actuator angle q_i.

$$q_i = tan^{-1}\,(p_z/p_x) \tag{4.18}$$

$$\mathbf{p}_i = \begin{bmatrix} d_i & \pm h_i \\ \mp h_i & d_i \end{bmatrix} \frac{\begin{bmatrix} f_{ix} \\ f_{iz} \end{bmatrix}}{\sqrt{(f_{ix}^2 + f_{iz}^2)}} \tag{4.19}$$

$$d = \frac{\left(f_x^2 + f_z^2\right) + l_1^2 - l_2^2}{2\sqrt{(f_x^2 + f_z^2)}} \tag{4.20}$$

$$h = \sqrt{l_1^2 - d^2} \tag{4.21}$$

4.1.1.3 Working Modes

Since circles in the same plane can intersect on two points and that circles intersection was used in both the IPK of the platform and the IPK of the legs, the IPK of the whole PentaG robot has several solutions. Those solutions are referred as the *working modes* of the robot and one must choose the correct solution corresponding to the working mode in which the robot operates. Normally, robots are designed to work in a single working mode. The poses which are at the limit of two working modes correspond to a serial singularity or type I singularities [26]. For a classic parallel robot, these singularities occur when one of the leg reaches the limits of the workspace corresponding to its serial chain. From the IPK analysis of the PentaG, and according to the definition of serial singularity of parallel robots, it is shown for the first time that parallel robots with configurable platforms have a different type of serial singularities that are not due to the workspace boundaries of the leg but are due to the workspace boundaries of the configurable platform.

This phenomenon can be understood with the help of Figure 4.2 b) representing the intersection of two circles. Consider a planar serial chain made of 3 revolute joints, with the base joint located in $\mathbf{p}_1$ and the end-effector joint located on $\mathbf{p}_2$. The second joint is at one of the two positions define by $\mathbf{p}_4$, which corresponds to one of the two working modes and the only way to change working mode for a fixed end-effector position $\mathbf{p}_2$ is to disassemble and reassemble the mechanism. If the position of the end-effector is now moved until $\|\mathbf{p}_2 - \mathbf{p}_1\| = |r_1 \pm r_2|$, the two solutions degenerate into a single solution. This specific pose is a serial singularity, which is located at the boundary of the configurable platform workspace and can be used as a transition between two working modes. Serial singularities and other type of singularities will be described in detail in Section 4.4.

In the case of the PentaG, two circle-circle intersections have been used for the IPK of the platform and there are therefore $2^2 = 4$ working modes for the platform. Each leg has also 2 solutions and there are therefore $2^5 = 32$ working modes for the legs given a certain platform working mode. The total number of working modes of the PentaG is then $2^{2+5} = 4 \times 32 = 128$. This means that given a certain end-effector position χ, there are 128 sets of actuator positions $\mathbf{q}$ possible.

4.1.2 Direct Position Kinematics

The direct position kinematics (DPK) is the expression of the position of the end-effector as function of the position of the actuator. The DPK is of great interest in control since the relation is needed to compute the position of the end-effector from the position data of the motor sensors. In a serial robot, all the joints are actuated and the calculation of the DPK of serial robots is generally straight forward. This is a very different situation for parallel robots, for which most of the joints are not actuated. In general, the DPK problem has multiple solutions and it is generally extremely difficult to obtain an analytical expression of the position of the end-effector as a function of the actuators position. Due to their complexity and importance, DPK problems are one of the major focuses of current research on parallel robots. For example, although introduced in the 60's and used in various applications since decades, it's only around the mid 90's that kinematicians found out that the DPK of general Gough platform has up to 40 different solutions [79] and found some analytical formulations of the DPK using Study's parameters [38]. In most parallel robots, the DPK is solved numerically using initial knowledge of the position of the end-effector. This section presents the DKP problem of the PentaG and explores some analytical and numerical methods to obtain solutions that can be used in real-time control.

4.1.2.1 Analytical Methods

Since the IPK of the PentaG was solved using intersections of geometrical objects, it is tempting to use the same approach for the DPK of the PentaG. As the actuator positions q_i is known from the motor sensors, the coordinates of points p_i (see Figure 4.3 a)) at the base of the parallelogram unit of each leg i can be calculated. The leg attach joints c_i on the configurable platform can freely rotate around point p_i, forming five spheres with radius ℓ_2 centered on p_i. The DPK problem can now be expressed as finding centre position, rotation and grasping values $\chi = [x\, y\, z\, \theta\, \rho]^T$ for the configurable platform such that each of the five leg attach points c_i are located somewhere on the surface of those five spheres.

This problem was solved analytically for the PentaG robot if the height of the platform is initially known but it was unfortunately not possible to obtain an analytical expression for the complete direct kinematic of the PentaG for an unknown height. In the absence of an analytical solution, numerical methods were investigated in order to solve the direct position kinematic problem.

4.1.2.2 Numerical Methods

It wasn't possible to obtain an analytical expression of the DPK of the PentaG, at least not something that can be directly implemented into an algorithm for real-time control. The computation of the DPK is nevertheless mandatory to extract the position of the end-effector from the data of the actuator sensors. In this section, a numerical procedure is presented, which is based on the Jacobian matrix J, that converges to the right solution of the DPK problem, given that a initial guess solution that is close enough to the real solution is used as input. The Jacobian matrix J of the PentaG will be presented in detail in section 4.2.1 as the locally linear function of the velocity of the end-effector as function of the angular velocity of the actuators.

$$\dot{\chi} = J_{5,5}\left(\chi\right)\dot{q} \tag{4.22}$$

Multiplying both sides of 4.22 by an infinitesimal change in time $\triangle t$, one see that the Jacobian can also be applied to small displacements.

$$\triangle\chi = J\triangle q \tag{4.23}$$

The numerical method used here can be related to Newton numerical method for finding the roots of a function. The procedure uses as input the actuator position q, an estimate χ_0 of the end-effector position, the jacobian $J\left(\chi\right)$ and inverse kinematic relation $\mathbf{q} = f\left(\chi\right)$. The procedure is as follows:

- Compute the inverse kinematic relation of the first end-effector position estimate to find a first actuator position estimate: $\mathbf{q_0} = f\left(\chi_\mathbf{0}\right)$
- Compute the difference between the estimated actuator position and the actual actuator position: $\triangle\mathbf{q_0} = \mathbf{q_0} - \mathbf{q}$
- Apply the Jacobian to obtain the change in end-effector position. $\triangle\chi_0 = J\left(\chi_0\right)\triangle\mathbf{q_0}$
- Find a new end-effector position estimate: $\chi_1 = \triangle\chi_0 + \chi_0$
- Repeat the procedure by replacing χ_{i-1} with χ_i until $\triangle q_i$ is smaller than the sensor resolutions.

The useful workspace of the robot was designed to contain no singularities (see Section 4.4), so the Jacobian matrix is a smooth function in this region where the robot is normally used. Implementation of this algorithm on the prototype shows that as long as the mechanism remains in the useful workspace, the iteration needs in most cases between 1 and 3 steps to obtain an accuracy of 10^{-6} rad.

4.1.2.3 Assembly Modes

The several solutions to the DPK are called assembly modes. If the robot is in a certain configuration with the actuators locked, one could disassemble the joints and reassemble the robot in a different configuration with the same actuator values $\overrightarrow{q}$. Since it was not possible to express the direct position kinematic in an analytical form, it is difficult to evaluate the actual number of assembly modes of the PentaG. In parallel robots, the boundaries between assembly modes are configurations where parallel singularity occurs, see Section 4.4. From the singularity analysis, it was concluded that at least 3 assembly modes exist in the workspace of the PentaG. However, the robot should always operate inside its useful workspace (see Chapter 5), which is free of singularities, and therefore shouldn't change its assembly mode.

4.2 Velocity Kinematics and Static Analysis

This section deals with velocity relationships between the various links and joints of the PentaG robot. On the contrary of position kinematics, velocity kinematics is locally linear, which implies that it can be described with linear algebra for a particular configuration. The analytical solutions in this section are presented as function of variable position vectors representing robot configurations. In a particular configuration, one can assign numerical values to these vectors using the solutions to the position kinematic problem presented in Section 4.1.1.

Several fundamental properties of a robot can be deduced from its velocity kinematic analysis. The linear velocity equations can be assembled into a matrix called the Jacobian matrix, from which it is possible to derive the mobility, accuracy, statics, stiffness and the singular configurations of the robot. Due to the presence of the configurable platform, it is not possible to apply classical and standard methods for the computation of the Jacobian matrix. In this section, alternative techniques are developed and some kinematic phenomenons such as platform singularities, which are unknown to classical parallel robots, are described.

4.2.1 The Jacobian Matrix

In velocity kinematics, the Jacobian matrix J represents the linear relation between the actuator velocity $\dot{\mathbf{q}}$ and the end-effector velocity $\dot{\chi}$ for a given configuration of the robot. If the robot doesn't have actuation redundancy, the Jacobian matrix is a square configuration-dependent $n \times n$ matrix where n is the number of DOF of the robot. For the PentaG, the Jacobian is a 5X5 matrix.

$$\dot{\chi} = J\dot{\mathbf{q}} \tag{4.24}$$

Assuming that the Jacobian is full rank, it can be inverted numerically to obtain the inverse Jacobian which gives us the actuator velocities as function of the end-effector velocities.

$$\dot{\mathbf{q}} = J^{-1}\dot{\chi} \tag{4.25}$$

As it was the case for position kinematics, the inverse Jacobian matrix is easier to compute analytically than the direct Jacobian matrix. The Jacobian can also be used to describe the static force relation between the actuator and the end-effector. For a small displacement of the end-effector, the instantaneous power transmitted, with no loss hypothesis, by the actuators must equal the power delivered by the end-effector. The power delivered by the actuators P_a and at the end-effector P_e are given by

$$P_a = \tau^T \dot{\mathbf{q}} = [\tau_1\, \tau_2\, \tau_3\, \tau_4\, \tau_5] \begin{bmatrix} \dot{q}_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \end{bmatrix} \tag{4.26}$$

$$P_e = \mathbf{F}^T \dot{\chi} = [f_x\, f_y\, f_z\, \tau_\theta\, \tau_\rho] \begin{bmatrix} \dot{q}_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \end{bmatrix} \tag{4.27}$$

From power conservation principle, the static force relation is then given by

$$P_a = P_e \tag{4.28}$$

$$\tau^T \dot{\mathbf{q}} = \mathbf{F}^T \dot{\chi} \tag{4.29}$$

$$\tau^T \dot{\mathbf{q}} = \mathbf{F}^T J \dot{\mathbf{q}} \tag{4.30}$$

$$\tau^T = \mathbf{F}^T J \tag{4.31}$$

$$\tau = J^T \mathbf{F} \tag{4.32}$$

Since the Jacobian matrix is dependent on the configuration of the robot, an analytical expression of the elements of the Jacobian matrix is needed, so that the numerical values corresponding to a particular position can be computed directly from this analytical formulation into the control algorithm. Since the Jacobian matrix is the derivative of the position kinematic, it may be tempting to obtain those solutions from the derivative of the expressions for the IPK found in Section 4.1.1. It should be noted that the terms in the actuator position Equation 4.18 are themselves solutions from several equations, also involving terms which are solutions of more equations. For these reasons, a direct single formula for the expression of actuator position as function of the end-effector position will lead to a very large equation difficult to manipulate. The derivative of such equation, for each term inside the Jacobian matrix, would be a very tedious process that would lead to an even longer equation.

Several standard method based on screw theory have been developed to compute the Jacobian of fully parallel robots. One of the most common procedure used for Jacobian analysis was developed by Mohamed and Duffy[55] and use the concept of reciprocal screws. In the case of the PentaG, it is unclear if this procedure can be used due to the presence of the configurable platform of the robot. The strategy used for the Jacobian computation of the PentaG is quite different and is based on the particular geometry of the robot and on static analysis. In order to obtain a analytical expression of the Jacobian, a more direct approach based on the velocity/static relation described by Equation 4.32 is used.

4.2.1.1 The Jacobian Matrix of the leg

As it was the case for the IPK, the problem can be divided in two parts; the Jacobian analysis of the configurable platform and the Jacobian analysis of the legs. Also in a similar way to position kinematics it is generally easier to compute the inverse of the Jacobian than the Jacobian matrix itself. The direct Jacobian can be obtained by inverting the matrix formed by the numerical values found in the inverse Jacobian. Since static analysis is often more intuitive to compute than velocity analysis, a first goal is to find the transpose of the inverse Jacobian that satisfies the following relation.

$$\mathbf{f}_{c,i} = J^{-T}\tau_i \tag{4.33}$$

where $\mathbf{f}_{c,i}$ is a 3 dimensional vector representing the forces exerted on the leg attach point $\mathbf{c}_i$ of leg i and τ_i is the torque force exerted by the actuator i. This relation is dependent on the configuration of the leg, which can be represented by the vectors $\mathbf{s}_i$, $\mathbf{l}_{1,i}$, and $\mathbf{l}_{2,i}$ as shown on Figure 4.4. Those vectors can be computed from the results obtained in the inverse position kinematics analysis developed in Section 4.1.1.2. From static analysis, the force on the leg middle point l_i is

$$\mathbf{f}_{l,i} = \frac{\left(\hat{\mathbf{s}}_{1,i} \times \hat{\mathbf{l}}_{1,i}\right)}{|\mathbf{l}_{1,i}|}\tau_i \tag{4.34}$$

Due to the construction of the parallelogram unit, the only linear forces that can be transmitted from the point l_i to the point c_i are in the direction of the vector $\hat{\mathbf{l}}_{2,i}$. Therefore, the force $\mathbf{f}_{c,i}$ has the same direction as the unit vector $\hat{\mathbf{l}}_{2,i}$ and its components in the direction of $\mathbf{f}_{l,i}$ must cancel the force applied on the point l_i .

$$\mathbf{f}_{c,i} = \frac{\hat{\mathbf{l}}_{2,i}}{\hat{\mathbf{l}}_{2,i} \cdot (\hat{\mathbf{s}}_{i,1} \times \mathbf{l}_{1,i})}\tau_i \tag{4.35}$$

From the static relation and the power conservation principle, the inverse Jacobian $J_{l,i}^{-1}$ for leg i is computed as

$$\mathbf{f}_{c,i} = J_{l,i}^{-T}\tau_i \tag{4.36}$$

$$\dot{q}_i = J_{l,i}^{-1}\mathbf{v}_{c,i} \tag{4.37}$$

$$\dot{q}_i = \frac{\hat{\mathbf{l}}_{2,i}^T}{\hat{\mathbf{l}}_{2,i} \cdot (\hat{\mathbf{s}}_{i,1} \times \mathbf{l}_{1,i})}\mathbf{v}_{c,i} \tag{4.38}$$

where $\dot{q}_i$ is the velocity of actuator i, and $\mathbf{v}_{c,i}$ is the 3D vector representing the velocity of the leg attach point c_i. The velocity relation is linear locally and dependent on the orientation of the leg vectors $\mathbf{l}_{1,i}$ and $\mathbf{l}_{2,i}$.

4.2.1.2 The Jacobian matrix of the configurable platform

In this section, the velocity of the leg attached points $\mathbf{v}_{c,i}$ as function of the end-effector velocity $\dot{\chi}$ is presented. Since the leg attach points for leg $i = 1, 3, 5$ is rigidly connected to the finger tips, the velocity relation is simple for these three legs. The computation of the velocity of the leg attach points for leg $i = 2, 4$ requires more careful analysis. From the construction of the mechanism, it is known that the links that form the attach

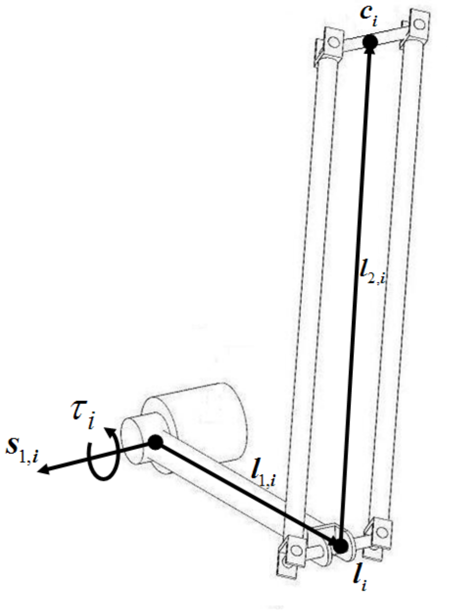


Figure 4.4: Static of one leg of the PentaG

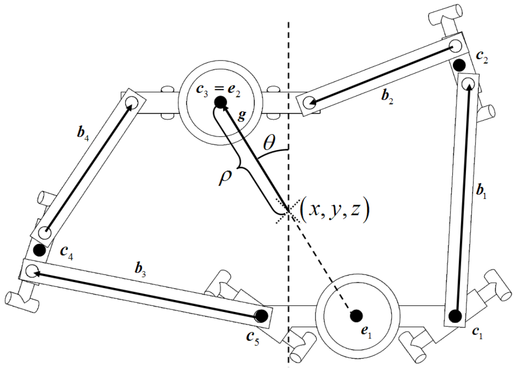


Figure 4.5: Vector notation used in the calculation of the inverse Jacobian of the configurable platform

points of the platform are constraint in rotation. Figure 4.5 shows the vectors used in the computation of the Jacobian matrix of the configurable platform. Only the links represented by the vectors $\mathbf{b}_1$, $\mathbf{b}_2$, $\mathbf{b}_3$, and $\mathbf{b}_4$ can rotate around the vertical axis.

The velocity of the end-effector is expressed with a 5 dimensional vector $\dot{\chi} = \left[\dot{x}\,\dot{y}\,\dot{z}\,\dot{\theta}\,\dot{\rho}\right]^T$. From this vector representation, the velocities $\dot{\mathbf{e}}_1$ and $\dot{\mathbf{e}}_2$ of the two finger tips can be calculated as

$$\dot{\mathbf{e}}_1 = [I_{3X3} \; -\hat{\mathbf{z}} \times \mathbf{g} \; -\hat{\mathbf{g}}] \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{\rho} \end{bmatrix} \tag{4.39}$$

$$\dot{\mathbf{e}}_2 = [I_{3X3} \; \hat{\mathbf{z}} \times \mathbf{g} \; \hat{\mathbf{g}}] \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{\rho} \end{bmatrix} \tag{4.40}$$

Since the legs $i = 1, 3, 5$ have their attach point c_i rigidly connected to the finger tip position, their velocity is easily given by $\dot{\mathbf{c}}_1 = \dot{\mathbf{c}}_5 = \dot{\mathbf{e}}_1$and $\dot{\mathbf{c}}_3 = \dot{\mathbf{e}}_2$. The velocity vectors $\dot{\mathbf{c}}_2$ and $\dot{\mathbf{c}}_4$ are computed from a certain combination of velocities $\dot{\mathbf{e}}_1$ and $\dot{\mathbf{e}}_2$. The component of the velocity that are due to the movements of the platform and the component of the velocity that are due to the deformation of the platform can be separated linearly. This is done by virtually considering the point e_1 as the base of the configurable platform. The velocity due to the angular component of the end-effector velocity $\dot{\theta}$ is first computed. When the point e_1 is considered fixed, the point c_2 can only move in the direction $\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1$. The velocity of the point e_2 relatively to point e_1 as function of the angular velocity $\dot{\theta}$ is given by

$$\dot{\mathbf{e}}_2 - \dot{\mathbf{e}}_1 = (\hat{\mathbf{z}} \times 2\mathbf{g})\,\dot{\theta} \tag{4.41}$$

The power conservation principle can be used again. If the point e_2 is constrained to move in the $\hat{\mathbf{z}} \times \hat{\mathbf{g}}$ direction, and the point c_2 is constrained to move in the $\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1$ direction, both relatively to point e_1, the linear relation between the force f_{e2} applied on point e_2 and the force f_{c2}, applied on point c_2 is given by

$$f_{e2} = \frac{(\hat{\mathbf{z}} \times \hat{\mathbf{g}}) \cdot \mathbf{b}_2}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) \cdot \mathbf{b}_2} f_{c2} \tag{4.42}$$

In the velocity domain, relatively to the velocity of point e_1, this gives

$$\dot{\mathbf{c}}_2 = \frac{(\hat{\mathbf{z}} \times \hat{\mathbf{g}}) \cdot \mathbf{b}_2}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) \cdot \mathbf{b}_2} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) |\dot{\mathbf{e}}_2| \tag{4.43}$$

using equation 4.41 to compute the magnitude of $\dot{\mathbf{e}}_2$ and adding the term $(-\hat{\mathbf{z}} \times \mathbf{g})\,\dot{\theta}$, so the velocity is expressed relatively to the velocity end-effector centre point $\dot{\mathbf{p}} = [\dot{x}\,\dot{y}\,\dot{z}]^T$

$$\dot{\mathbf{c}}_2 = \left(\frac{(\hat{\mathbf{z}} \times 2\mathbf{g}) \cdot \mathbf{b}_2}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) \cdot \mathbf{b}_2} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) - \hat{\mathbf{z}} \times \mathbf{g} \right) \dot{\theta} \tag{4.44}$$

A similar procedure is used to compute the velocity of c_2 as a function of the end-effector grasping velocity $\dot{\rho}$. In this case, the velocity of e_2 relatively to the velocity of e_1 is given by $\dot{\mathbf{e}}_2 - \dot{\mathbf{e}}_1 = (2\mathbf{g})\,\dot{\rho}$ and the velocity of e_1 relatively to the centre point p is given by $-\hat{\mathbf{g}}\dot{\rho}$. The same reasoning is used to adapt Equation 4.44 to the grasping velocity $\dot{\rho}$

$$\dot{\mathbf{c}}_2 = \left(\frac{2\hat{\mathbf{g}} \cdot \mathbf{b}_2}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) \cdot \mathbf{b}_2} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) - \hat{\mathbf{g}} \right) \dot{\rho} \tag{4.45}$$

Equations 4.45 and 4.44 give the velocity of point c_2 relatively to the centre point p_1. The centre point of velocity need to be added to obtain the velocity relatively to the base. The full Jacobian of point c_2 as a function of the end-effector position is given by

$$\dot{\mathbf{c}}_2 = \left[I_{3X3} \; \tfrac{(\hat{\mathbf{z}} \times 2\mathbf{g}) \cdot \mathbf{b}_2}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) \cdot \mathbf{b}_2} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) - \hat{\mathbf{z}} \times \mathbf{g} \; \tfrac{2\hat{\mathbf{g}} \cdot \mathbf{b}_2}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) \cdot \mathbf{b}_2} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) - \hat{\mathbf{g}} \right] \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{\rho} \end{bmatrix} \tag{4.46}$$

The same reasoning can be applied to find the relation between the velocity of point c_4 and the velocity of the end-effector.

$$\dot{\mathbf{c}}_4 = \left[I_{3X3} \; \tfrac{(\hat{\mathbf{z}} \times 2\mathbf{g}) \cdot \mathbf{b}_4}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) \cdot \mathbf{b}_4} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_3\right) - \hat{\mathbf{z}} \times \mathbf{g} \; \tfrac{2\hat{\mathbf{g}} \cdot \mathbf{b}_4}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_3\right) \cdot \mathbf{b}_4} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_3\right) - \hat{\mathbf{g}} \right] \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{\rho} \end{bmatrix} \tag{4.47}$$

The velocities of the five leg attach points have now been described as function of the end-effector velocity.

4.2.1.3 The PentaG Jacobian Matrix

In section 4.2.1.1 the inverse Jacobian matrix for each leg has been computed as

$$\dot{q}_i = J_{l,i}^{-1} \mathbf{v}_{c,i} = \frac{\hat{\mathbf{l}}_{2,i}^T}{\hat{\mathbf{l}}_{2,i} \cdot (\hat{\mathbf{s}}_{i,1} \times \mathbf{l}_{1,i})} \mathbf{v}_{c,i} \tag{4.48}$$

And in section 4.2.1.2, the inverse Jacobian matrix $J_{c,i}^{-1}$for the leg attach points on the platform has been computed as

$$J_{c,1}^{-1} = [I_{3X3} \; -\hat{\mathbf{z}} \times \mathbf{g} \; -\hat{\mathbf{g}}] \tag{4.49}$$

$$J_{c,2}^{-1} = \left[I_{3X3} \; \tfrac{(\hat{\mathbf{z}} \times 2\mathbf{g}) \cdot \mathbf{b}_2}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) \cdot \mathbf{b}_2} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) - \hat{\mathbf{z}} \times \mathbf{g} \; \tfrac{2\hat{\mathbf{g}} \cdot \mathbf{b}_2}{\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) \cdot \mathbf{b}_2} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1\right) - \hat{\mathbf{g}} \right] \tag{4.50}$$

$$J_{c,3}^{-1} = [I_{3X3}\ \hat{\mathbf{z}} \times \mathbf{g}\ \hat{\mathbf{g}}] \tag{4.51}$$

$$J_{c,4}^{-1} = \left[I_{3X3}\ \frac{(\hat{\mathbf{z}} \times 2\mathbf{g}) \cdot \mathbf{b}_4}{(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_3) \cdot \mathbf{b}_4} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_3\right) - \hat{\mathbf{z}} \times \mathbf{g}\ \frac{2\hat{\mathbf{g}} \cdot \mathbf{b}_4}{(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_3) \cdot \mathbf{b}_4} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_3\right) - \hat{\mathbf{g}}\right] \tag{4.52}$$

$$J_{c,5}^{-1} = [I_{3X3}\ -\hat{\mathbf{z}} \times \mathbf{g}\ -\hat{\mathbf{g}}] \tag{4.53}$$

To obtain the full inverse Jacobian matrix that describes the angular velocity of the actuators as function of the angular velocity of the end-effector, the inverse Jacobian of each attach point can be multiplied with the inverse Jacobian of the corresponding leg. The expression of the inverse Jacobian J^{-1} is then given by

$$\dot{\mathbf{q}} = J^{-1}\dot{\chi} \tag{4.54}$$

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \end{bmatrix} = \begin{bmatrix} J_{l,1}^{-1} J_{c,1}^{-1} \\ J_{l,2}^{-1} J_{c,2}^{-1} \\ J_{l,3}^{-1} J_{c,3}^{-1} \\ J_{l,4}^{-1} J_{c,4}^{-1} \\ J_{l,5}^{-1} J_{c,51}^{-1} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{\rho} \end{bmatrix} \tag{4.55}$$

It can be noted that the calculation of the inverse Jacobian matrix of PRCPs is different than for pure parallel mechanims. It must be performed in two steps where both the inverse Jacobians of the platform attach points $J_{c,i}^{-1}$, of dimension 3X5, and the inverse Jacobians of the legs $J_{l,i}^{-1}$, of dimension 1X3, are calculated. The same principle can be applied to the calculation of the Jacobian of the static forces. Equation 4.32 already presented the mapping relation between the end-effector forces $\mathbf{F}$ and the actuator torques τ as the transpose of the Jacobian matrix, $\tau = J^T\mathbf{F}$. The Jacobian of force transmissions is then expressed as

$$\begin{bmatrix} f_x \\ f_y \\ f_z \\ \tau_\theta \\ f\rho \end{bmatrix} = \begin{bmatrix} J_{c,1}^{-T} J_{l,1}^{-T} & J_{c,2}^{-T} J_{l,2}^{-T} & J_{c,3}^{-T} J_{l,3}^{-T} & J_{c,4}^{-T} J_{l,4}^{-T} & J_{c,5}^{-T} J_{l,5}^{-T} \end{bmatrix} \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \\ \tau_5 \end{bmatrix} \tag{4.56}$$

This analytical relation can be implemented directly in the controller for real-time computation. For a desired end-effector velocity, the corresponding actuators velocity is computed from Equation 4.55. An analytical expression of the direct Jacobian Matrix by inversion of the inverse Jacobian Matrix will be difficult to calculate since the Jacobian Matrix is of dimension 5X5. If the value of the direct Jacobian Matrix J is needed, a numerical computation of the inverse of the Jacobian Matrix J^{-1} for a given configuration will be more efficient. The Jacobian Matrix also allows us to establish several relevant properties and performance of the robot, and plays a central role in the robot analysis. Some of these properties are presented in the following sections.

4.2.2 Accuracy and Stiffness

Since the Jacobian Matrix describes the velocity relation and that velocity corresponds to the derivative of position over time, the Jacobian Matrix can be also used to describe the linear relation between small displacements.

$$\dot{\chi} = J\dot{\mathbf{q}} \tag{4.57}$$

$$\frac{\Delta\chi}{\Delta t} = J\frac{\Delta\mathbf{q}}{\Delta t} \tag{4.58}$$

$$\Delta\chi = J\Delta\mathbf{q} \tag{4.59}$$

The Jacobian can be used to compute the resolution of the workspace as a function of the resolution of the encoders. The errors on the position encoders are bounded by the encoder resolutions. All errors are independent and the possible actual encoder positions form a 5-dimensional hyper-cube in the joint space. By mapping each corner of this hyper-cube with the Jacobian into the end-effector space, a 5 dimensional hyper-polyhedron (A 3D polyhedron is a solid with flat faces and strait edges) is obtained, which represents the sensitivity against variations of the robot for a given configuration.

Assuming that the mechanical links of the robot are perfectly rigid, the stiffness at the end-effector can be described as a function of the stiffness of the actuators. For rotary motor i, the angular displacement Δq_i produces by a torque τ_i with a stiffness k are related by

$$\tau_i = k\Delta q_i \tag{4.60}$$

Assuming that the same stiffness k is applied to each actuator, the stiffness of the end-effector, when all actuators are locked is described by

$$\mathbf{F} = J^{-T}\tau \tag{4.61}$$

$$\mathbf{F} = J^{-T}k\Delta\mathbf{q} \tag{4.62}$$

$$\mathbf{F} = kJ^{-T}J^{-1}\Delta\chi \tag{4.63}$$

The matrix $K = kJ^{-T}J^{-1}$ is called the stiffness matrix. The actual value of k depends on the type of motor used and the gains of the controller.

4.3 Static Balancing

Static balancing is a technique to create equilibrium throughout a considerable range of motion of a mechanism, even in the absence of friction [32]. Static balance is characterized by a constant system potential, regardless of its configuration. In principle, any conservative system can be statically balanced. A common application of static balance is present in gravity equilibration. Gravity balancing has many advantages including reduced operating effort, energy conservation and reduced heat production in actuators, and safety in case of power failure. An overview of gravity balancers using counterweights or springs is available in [32].

It has been shown that any linkage can be statically balanced [73] but practical implementations are largely limited to serial open-chain mechanisms and single-degree-of-freedom closed-chain linkages [32]. However, also parallel kinematic manipulators and mechanisms may benefit from static balancing. In parallel mechanisms, static balancing is much less common. Reported strategies include the incorporation of balancers in the legs of parallel mechanisms (‘integrated balancing systems’), or use separate additional legs to house the balancers (‘separate balancing systems’) [33]. Some publications on statically balanced parallel mechanisms exist for gravity equilibration [39, 27, 28, 17, 68]. Also in cable-actuated parallel mechanisms the use of springs is proposed to decrease the effect of gravity in static mode [3].

One evident problem in the reported cases of static balancing in parallel mechanisms is the increased mechanical complexity of the resulting mechanisms. Usually auxiliary links and many springs are added to achieve static balance. In [33] it is argued that even spatial kinematic mechanisms theoretically need only a single balancing spring or counterweight to achieve perfect static balance, yet for instance in [28] twelve springs are applied. These springs need to have zero free length for perfect static balance, which are hard to obtain or require further mechanical complication.

More recently, [2] proposed a pantograph mechanism and a counterweight as a separate compensation system to gravity balance parallel mechanisms, where the auxiliary pantograph needs to apply at the centre of mass of the moving platform. Also [80] uses a pantograph, this time as an integrated compensation system to dynamically force balance the Delta robot, which implies static balance. Four counterweights and one auxiliary pantograph are needed for perfect gravity equilibration. It therefore remains an open issue how to create mechanically simple gravity balancers.

This section proposes a concept for perfect static balancing for a subclass of translational parallel mechanisms, which includes the well known Delta Robot [10], but also the PentaG which uses the same type of limbs as the Delta robot. The specific kinematic features are exploited to arrive at simple gravity balancers without auxiliary links.

4.3.1 Concept

Much of the mechanical complexity that is observed in present static balancers in parallel mechanisms is due to the need to create a vertical reference to which attach the springs, as shown in Figure 4.6. This vertical may move but needs to remain vertical, i.e. should translate only. In translational mechanisms, this feature of pure translation is inherently present. Therefore it requires no auxiliary mechanism to create the proper spring attachment points: a vertical can simply be attached to the moving platform.

This feature can be employed as illustrated in Figure 4.7 a) for a planar example of a 2RRR robot with an auxiliary passive RRR leg complementing one of the driven legs in a double parallelogram arrangement to create the translational behavior of the platform. Each leg is now furnished with two zero-freelength springs. The first one is connected between the base and the grounded links, and the second one between the platform and the floating links. Note that the passive leg is not used to house the springs but purely to create the translational motion.

Assuming that the centre of mass of the platform is on its centre line, then the mass m is equally distributed between the two active legs. Considering one such leg as in figure 4.7 b), the platform spring can be rearranged such that its potential energy function is not affected by applying modification rules [32]. First the platform spring element is shifted up to the floating joint as in Figure 4.7 c). To maintain the vertical, an auxiliary parallelogram would need to be formed by two added links. As a next step, the spring element is shifted to the base joint as shown on Figure 4.7 d). To maintain the proper

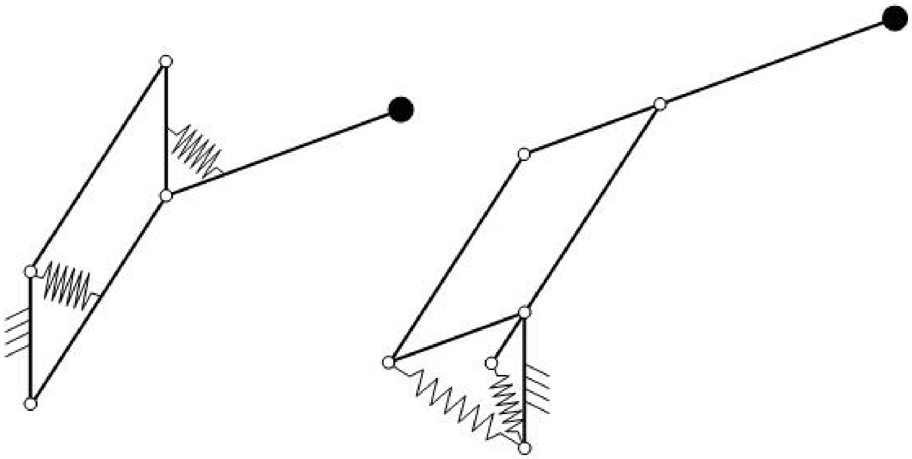


Figure 4.6: 2 DOF open chains with end load and static balancers: (a) vertical type, (b) parallel type.

synchronization with the platform, the floating link is extended and one link is added to form a pantograph. This way, it is shown that the leg corresponds to a well known balancer, i.e. the balanced five-bar also known as the Anglepoise linkage [7], and also known as the parallel type [73]. This shows that the mechanism is gravity balanced from a potential energy perspective.

Similarly, it can be shown that the same concept can be applied to the Delta robot. In spite of its more complex spatial motion, the very same balancing spring arrangement applies. Furthermore, the concept can also be applied to the PentaG, in spite of the fact that its platform is articulated. This is presented in the next section.

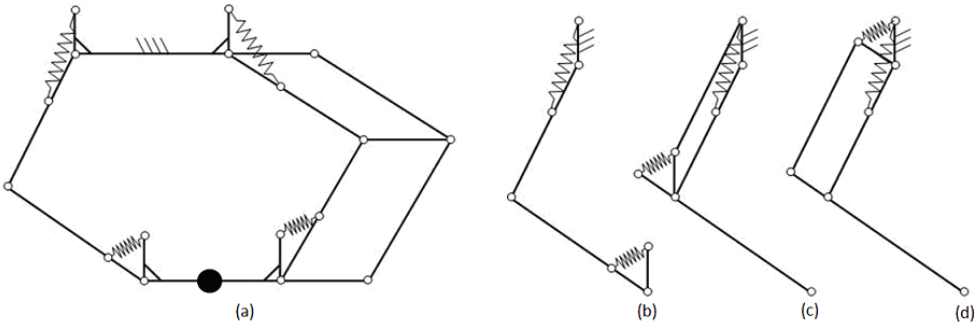


Figure 4.7: Planar translational platform: (a) complete with proposed balancer, (b) one leg, (c) bottom spring shifted to floating joint, (d) spring shifted further to base joint.

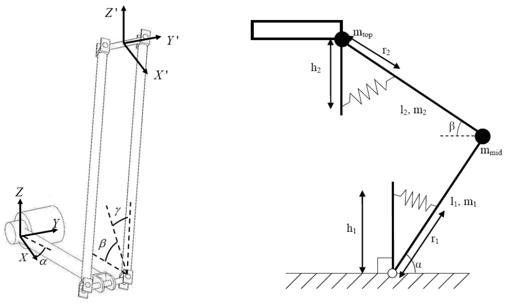


Figure 4.8: Notation for a Delta Leg

4.3.2 Application to the PentaG

For a platform with a mass m_p , the change in potential energy ΔV_p of the platform is directly related to the change of the vertical position of the attached leg Δz by $\Delta V_p = m\Delta z_p$. The vertical distance between the COM of each leg and the attachment point is not constant for every Δz. However, given a certain height for the attachment point Δz_p , this is constant for any horizontal displacement. Therefore, the change in potential energy of a leg i is defined as $\Delta V_p = f_i\left(z_p\right)$.

The function $f_i\left(z_p\right)$ will be explained in detail further. To fully balance the robot to gravity, the sum of the change of potential energy ΔV_i has to cancel ΔV_p , for any configuration.

$$\Delta V_p + \sum_{i=1}^{5} \Delta V_i = 0 \tag{4.64}$$

$$m_p \Delta z_p + \sum_{i=1}^{5} f_i\left(z_p\right) = 0 \tag{4.65}$$

A delta leg can be decomposed into two main units; the base link, rotating around the motor shaft, and the parallelogram unit, attached to the platform with a revolute joint.

In literature, for example [73], a double pendulum or a parallelogram structure is always used to provide a vertical connection point for the parallelogram unit. However, in the case of the 3 DOF delta robot[10] , 4 DOF par4 robot[59], or 5 DOF PentaG robot, one can make use of the fact that the platform always stays horizontal. Instead of balancing the parallelogram unit from below, it could be balanced from above by mounting the spring to a connection point on the platform. Obviously the advantage of the latter is the simplicity: less extra elements are added to the system and the increase in mass is greatly reduced.

In Figure 4.8, a single leg of the PentaG robot is shown. The leg is modeled as a double pendulum with point masses at the top of each of the pendulum, and both bars having their own mass. The length of the revolute unit and parallelogram unit are respectively l_1 and l_2 and the mass m_1 and m_2 . The centre of gravity of the limbs is exactly halfway their length, and for the parallelogram unit exactly in the middle of the two long bars. The point masses at the middle and the top are m_{mid} and m_{top} respectively. The angles between the joint units and the horizontal are respectively α and β . In addition, the sideway rotation of the parallelogram relative to it's position in the XZ plane is noted γ . The global coordinate system (X,Y,Z) is fixed to the outside world where $(0,0,0)$ is defined at the base joint. The local coordinate system (X',Y',Z') has its origin at the bottom joint, where the platform is connected to the leg, and has the same orientation as the global coordinate system. Springs are connected between the joint units and the frames at locations respectively defined by the parameters r and h. At the revolute joint unit, a spring is applied between the environment at the coordinates $(X,Y,Z)=(0,0,h)$ and the limb at distance r_1 from the base joint. Because the parallelogram unit consists of two bars, two springs are connected between the point $(X',Y',Z')=(0,0,-h)$ to each of the bars at distance r_2 from the bottom end of the bar. The stiffness of the lower spring is k_1 and for both the upper springs the stiffness is k_2 . All springs have zero free length.

The potential energy due to gravity for one leg is given by:

$$V_{gravity}=\left(\frac{1}{2}m_1+m_2+m_{mid}+m_{top}\right)gl_1sin\left(\alpha\right)+\left(\frac{1}{2}m_2+m_{top}\right)gl_2sin\left(\beta\right)cos\left(\gamma\right) \tag{4.66}$$

For the lower spring, the elongation is given by:

$$e_1^2=h_1^2+r_1^2-2h_1r_1sin\left(\alpha\right) \tag{4.67}$$

For the upper springs, the (X',Y',Z') coordinates of the connection points of the springs to the limbs are:

$$(X',Y',Z')=\left(-r_2cos\left(\gamma\right)cos\left(\beta\right),-r_2sin\left(\gamma\right)\pm\omega,-r_2cos\left(\gamma\right)sin\left(\beta\right)\right) \tag{4.68}$$

where ω is the half distance between the two long bars of the parallelogram unit. Therefore, the elongations of these springs are:

$$e_{2a}^2=\left(\left(r_2cos\left(\gamma\right)cos\left(\beta\right)\right)^2+\left(r_2sin\left(\gamma\right)-\omega\right)^2+\left(r_2cos\left(\gamma\right)sin\left(\beta\right)-h_2\right)^2\right) \tag{4.69}$$

$$e_{2a}^2=h_2^2+r_2^2+\omega^2-2\omega r_2sin\left(\gamma\right)-2h_2r_2cos\left(\gamma\right)sin\left(\beta\right) \tag{4.70}$$

and

$$e_{2a}^2=\left(\left(r_2cos\left(\gamma\right)cos\left(\beta\right)\right)^2+\left(r_2sin\left(\gamma\right)+\omega\right)^2+\left(r_2cos\left(\gamma\right)sin\left(\beta\right)-h_2\right)^2\right) \tag{4.71}$$

$$e_{2a}^2=h_2^2+r_2^2+\omega^2+2\omega r_2sin\left(\gamma\right)-2h_2r_2cos\left(\gamma\right)sin\left(\beta\right) \tag{4.72}$$

The potential energy in the springs is then:

$$V_{springs}=\frac{1}{2}k_1\left(h_1^2+r_1^2-2h_1r_1sin\left(\alpha\right)\right)+k_2\left(h_2^2+r_2^2+\omega^2-2h_2r_2cos\left(\gamma\right)sin\left(\beta\right)\right) \tag{4.73}$$

leg	m_1	m_2	m_{mid}	m_{top}
1	0.0344	0.0158	0.0018	0.0128
2	0.0314	0.0051	0.0006	0.0123
3	0.0285	0.0144	0.0015	0.0145
4	0.0314	0.0051	0.0006	0.0123
5	0.0344	0.0158	0.0018	0.0128

Table 4.1: Limb masses of the PentaG

leg	k_1(N/m)	k_2(N/m)
1	46.7	23.5
2	29	14.7
3	42.4	22.4
4	29	14.7
5	46.7	23.5

Table 4.2: Stiffness Values used for the proposed balancing method

Comparing this with the potential energy due to gravity, it follows that this leg will be balanced perfectly under the two following conditions:

$$k_1 h_1 r_1 = \left(\frac{1}{2}m_1 + m_2 + m_{mid} + m_{top}\right) g l_1 \tag{4.74}$$

$$2k_2 h_2 r_2 = \left(\frac{1}{2}m_2 + m_{top}\right) g l_2 \tag{4.75}$$

The entire robot will be balanced perfectly if the weight of the platform is divided over the five legs and each leg fulfils the conditions found previously. Since the platform has pure translational motions, the mass of the platform $m_{platform}$ can be distributed arbitrarily between the five legs. Distribute the support of the platform equally between the 5 legs for each leg, $m_{top,i}$ is equal to the mass of the upper short link of the parallelogram m_{upper} plus a fifth of the platform mass, plus the mass of the extra structure at the platform needed to attach the spring m_k. The mass of the spring is neglected:

$$m_{top,i} = \frac{1}{5}m_{platform} + m_{upper,i} + m_{k,i} \tag{4.76}$$

4.3.3 Simulation

To show a numerical example, the masses of the various links of the PentaG prototype, shown in Table 4.1, are used. The total mass of the platform is $m_{platform} = 0.0435$.

The value of h and r are set to 20mm and 60mm respectively. the following values for the spring stiffness are obtained:

These values were checked in a MatLab SimMechanics simulation model, in which the robot is built from basic link, joints and springs elements. In this model, a helical trajectory, covering most of the robot workspace was generated for the platform, and the actuator torques needed to perform this trajectory were recorded. In addition, the 2 DOF of the platform were also actuated from the base motors, in order to show that this balancing technique applies not only to Delta robot, but also to more recent Delta-like robots with

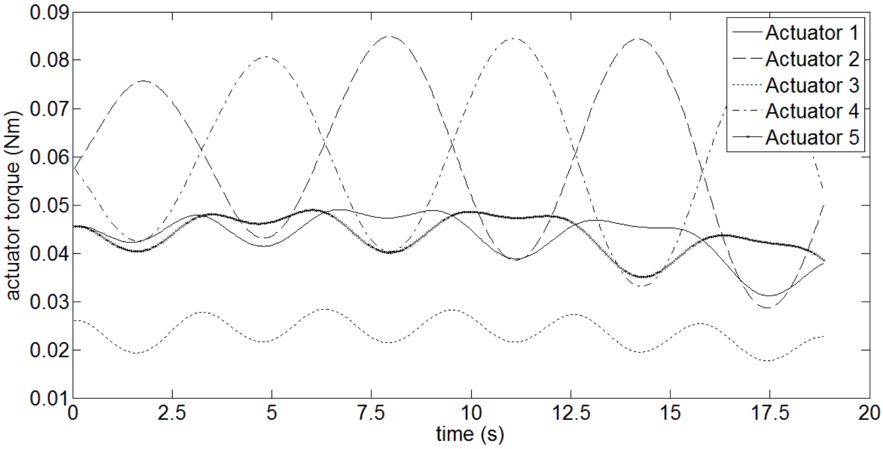


Figure 4.9: Actuator Torques without static balancing

configurable platform. The motion is performed slowly (around 1 rad/s), to minimize the effects of the inertia on the actuator torques. The following helical trajectory (in meters) is used for the platform:

$$X = 0.05 sin\left(t\right) \tag{4.77}$$

$$Y = 0.05 cos\left(t\right) \tag{4.78}$$

$$Z = 0.14 + 0.05t \tag{4.79}$$

where t is the time in second. The rotation θ , and the grasping distance between the finger tips ρ of the platform is:

$$\theta = 0.3 sin\left(t\right) \tag{4.80}$$

$$\rho = 0.01 sin\left(t\right) \tag{4.81}$$

Figures 4.9 and 4.10 show the five actuator torques needed to perform this motion with and without the balancing system respectively. In the unbalanced robot, the actuators are mainly used to equilibrate the gravity force acting on the links and platform. In the balanced version, the actuators only need to generate the inertial forces, which are low since the motion is performed slowly.

A ratio of approximately 1000 can be observed between the torques used in the unbalanced and in the balanced robot, which is also approximately the ratio between the gravity forces and the inertia forces that are expected for the slow motion used. Thus, this model shows that the robot indeed is statically balanced under the above conditions.

4.3.4 Discussion

The proposed balancer is based on zero-free-length springs. In practice these are hard to obtain, although it is possible to produce them by special coiling techniques and tailored

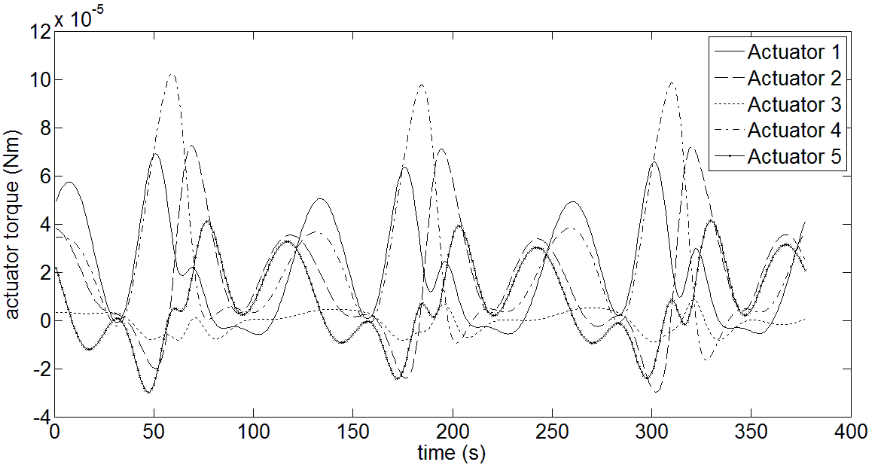


Figure 4.10: Actuator Torques with Static Balancing

heat treatment. If one chooses to emulate the zero-free-length behavior, for instance with a pulley and string arrangement or other mechanical means, then some mechanical complexity is added to the system, as well as some friction. This may degrade the dynamic performance of the robot somewhat.

It was chosen to distribute the load of the platform mass equally among the limbs of the manipulator. This results in the least additional load on the links. However, the number of springs is considerable, i.e. ten in the case of the PentaG. Due to the translational motion, each point of the platform has the same gravity potential. Therefore the mass can be distributed arbitrarily among the limbs. If one is interested in minimizing the number of springs, for instance, the load can be taken by a single limb, requiring only two springs, regardless of the number of legs.

In the balancers, not only the platform mass but also the link masses, and even the spring mass can be taken into account. The spring mass can be equally distributed amongst its end points, and added to the platform or link mass as appropriate.

The addition of mass and, especially in this case, the addition of friction is not a desirable feature for haptic devices. In applying static balancing, one must be careful that the advantages gained from the gravity compensation are not cancelled out by the disadvantages that come with additional masses and friction. For those reasons, it was chosen to not implement for now this solution in the current haptic version of the PentaG prototype. However, friction is much less of a problem for manipulator robots, and the novel proposed balancing solution might be highly beneficial for pick-and-place robots based on Delta, par4, and the PentaG architectures.

Actually the specific mobility, the high stiffness and low inertia of the PentaG architecture make it a good candidate to be used as high-speed pick-and-place robot. In its pick-and-place version, the PentaG robot is turned upside down with the base connected to a ceiling and the finger tips replaced with a gripper linkage to pick and move objects. A detailed explanation of the pick-and-place version of the PentaG robot is presented in Chapter 6. A patent has been granted for the pick-and-place version of the PentaG and the company owning this patent intents to use the novel static balancing method presented in this section on their prototype.

In summary, the static balancing method presented in this section can in theory perfectly compensate for gravity for any type of translational robot using Delta legs. The key idea is the use of the constrained platform (rigid or configurable) to provide a vertical attach point to the springs balancing the upper part of the Delta legs. The main advantage of the method compared to what was already proposed is the low number of springs used and the absence of additional moving links to compensate the upper links, resulting in less additional mass and friction.

4.4 Singular Configurations

In mathematics, a singularity is a point where a mathematical object is not well-defined or loses some properties such as differentiability. In linear algebra, a square matrix is said to be singular if it is not full rank. Singular square matrices are not invertible, their determinant is zero and they have a non trivial null space. The concepts of local singularities and constraint singularities already discussed briefly in Chapter 2. These singularities are properties of the mechanism itself and are not dependent on which joints of the mechanism are actuated in order to control the robot.

This section descibes singularities that are related to the actuated joints, namely on the linear relation between the actuator velocity inputs and the end-effector velocity outputs. In Section 4.2, this velocity relation was described by a five by five Jacobian matrix and an analytical expression for the inverse Jacobian was presented. Among the firsts to study singularities of parallel robots were Gosselin and Angeles[26] with later generalization from Zlatanov [84]. Based on their work, two types of singularity for parallel robots can be defined; serial singularities and parallel singularities. These concepts can be extended to parallel robots with configurable platforms.

Considering the Jacobian square matrix J, the actuator velocity vector $\dot{\mathbf{q}}$, and the end-effector velocity vector $\dot{\chi}$ of a general robot, the value of J is dependent of the configuration of the robot.

$$J\left(\chi\right)\dot{\mathbf{q}} = \dot{\chi} \tag{4.82}$$

When the robot reaches a configuration χ where the determinant the Jacobian vanishes, i.e. $det\left(J\left(\chi\right)\right) = 0$ and the lines in J are no longer independent, a first consequence is that the null space of J is no longer trivial and it is possible to find non-zero actuator velocities $\dot{\mathbf{q}}$ that produce zero end-effector velocity $\dot{\chi} = 0$. A second consequence is that the end-effector space lost a dimension in this configuration. It is now impossible to produce some of the velocity outputs $\dot{\chi}$, no matter which actuator speed $\dot{q}$ is used as input. This implies that the robot is in a configuration with is at the limit of the end-effector workspace.

Due to the power conservation principle, the Jacobian matrix can also be used to describe force transmission; $\tau = J^T\mathbf{F}$. It follows that in the singular configuration described above, some forces applied on the end-effector can be balanced by the robot even if no torque is applied on the actuators, and that some actuator torques cannot be balanced by applying forces on the end-effector.

For configurations near those singularities, the end-effector moves slowly with high accuracy while high but inaccurate forces can be applied by the actuators on the end-effector. These types of configurations are called singularities of type I or serial singularities. Serial singularities also form boundaries between the working modes of the robot. Given an initial end-effector position and actuator position, the robot can reach a serial singularity, and then go back to this initial end-effector position with different values for actuators.

In the neighbourhood of serial singular configurations (near the limits of the end-effector workspace), the determinant of the Jacobian J tends toward zero while the determinant of the inverse Jacobian J^{-1} tends toward infinity. This suggests the existence of a second type of singularities called type II or parallel singularities that occurs when the determinant of the inverse Jacobian is zero and the determinant of the direct Jacobian tends toward infinity. Those singularities are called parallel because unlike type I singularities, they only occur in parallel robot and are not present in serial robots.

The properties of parallel singularity configurations are dual to the properties of serial singularity configurations. In parallel singular configurations, the end-effector can have a non-zero velocity while the actuators are all locked. Some combined velocities of the actuators are impossible which means that those configurations are located at the limits of the actuator space. Some combined forces from the actuators will produce zero forces on the end-effector and it is not possible to balance some end-effector forces with the actuators. Near parallel singularities, the end-effector can move very fast and is inaccurate while only low forces can be applied by the actuators on the end-effector. Parallel singularities form boundaries of the assembly modes. Given an initial actuator position and end-effector position, the robot can reach a parallel singularity, and then go back to the initial actuator position with a different end-effector position.

In practice, these properties make the parallel singularities much more problematic than serial singularities. First, parallel singularities are located inside the end-effector workspace while serial singularities are confined to the workspace boundaries. Near parallel singularities, the position sensing, derived from the direct position kinematic and the actuator sensor values, is inaccurate and only weak forces can be applied on the end-effector due to the low force transmission ratio. Considering closed-loop controller and dynamics effects, it is likely that the robot loses control of the end-effector if it comes too close of a parallel singularity. In the design process of parallel robots, one should try to remove parallel singularities from the centre of the workspace and be aware of the location of those singularities, in any. This is the subject of chapter 5.

An example of serial and parallel singularities with a simple 2 DOF planar robot is presented in Figure 4.11 a). In this robot, the two actuators q_1 and q_2 are used to control the position of the middle joint in the χ_1 and χ_2 direction of the plane. Figure 4.11 b) and Figure 4.11 c) show the robot in a serial and parallel singularity configuration. By inspection of this simple mechanism one can deduce that, in those configurations, the mechanism has indeed the special properties of serial and parallel singularities discussed above.

In this section, the concepts of serial singularities and parallel singularities will be extended to parallel robots with configurable platforms. It will be shown that serial singularities are presents not only at the workspace limits of the legs, but also at the workspace limit of the platform. The PentaG robot has unfortunately some parallel singularities inside its workspace and that those singularities are dependent of both the configuration of the legs of the robot and the configuration of the platform.

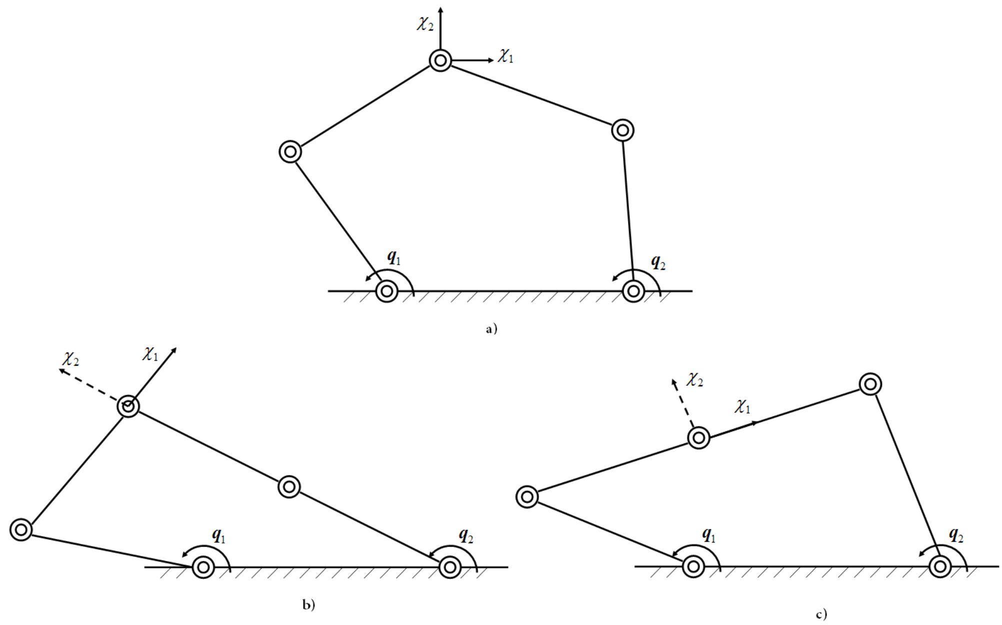


Figure 4.11: A 2 DOF Parallel mechanism in a) non singular configuration b) in serial singularity configuration c) and in parallel singularity configuration

4.4.1 Serial Singularities of the PentaG

As it is the case for parallel robots, serial singularities occur also in parallel robots with configurable platforms like the PentaG. Serial singularities are by definition poses of the end-effector where the determinant of the direct Jacobian matrix is zero. Although an analytical expression of the direct Jacobian was not possible, the determinant of the inverse Jacobian tends toward infinity in serial singular configuration. An analytical expression of the inverse Jacobian matrix was found in section 4.2.1 as

$$\dot{\mathbf{q}} = J^{-1}\dot{\chi} \tag{4.83}$$

$$\begin{bmatrix} \dot{q}_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \end{bmatrix} = \begin{bmatrix} J_{l,1}^{-1} J_{c,1}^{-1} \\ J_{l,2}^{-1} J_{c,2}^{-1} \\ J_{l,3}^{-1} J_{c,3}^{-1} \\ J_{l,4}^{-1} J_{c,4}^{-1} \\ J_{l,5}^{-1} J_{c,51}^{-1} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{\rho} \end{bmatrix} \tag{4.84}$$

$$J_{l,i}^{-1} = \frac{\hat{\mathbf{l}}_{2,i}^{T}}{\hat{\mathbf{l}}_{2,i} \cdot (\hat{\mathbf{s}}_{i,1} \times \mathbf{l}_{1,i})} \tag{4.85}$$

$$J_{c,1}^{-1} = [I_{3X3} - \hat{\mathbf{z}} \times \mathbf{g} - \hat{\mathbf{g}}] \tag{4.86}$$

$$J_{c,2}^{-1} = \left[I_{3X3} \frac{(\hat{\mathbf{z}} \times 2\mathbf{g}) \cdot \mathbf{b}_2}{(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1) \cdot \mathbf{b}_2} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1 \right) - \hat{\mathbf{z}} \times \mathbf{g} \frac{2\hat{\mathbf{g}} \cdot \mathbf{b}_2}{(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1) \cdot \mathbf{b}_2} \left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_1 \right) - \hat{\mathbf{g}} \right] \tag{4.87}$$

$$J_{c,3}^{-1} = [I_{3X3}\,\hat{\mathbf{z}} \times \mathbf{g}\,\hat{\mathbf{g}}] \tag{4.88}$$

$$J_{c,4}^{-1} = \left[I_{3X3}\,\frac{(\hat{\mathbf{z}}\times 2\mathbf{g})\cdot\mathbf{b}_4}{\left(\hat{\mathbf{z}}\times\hat{\mathbf{b}}_3\right)\cdot\mathbf{b}_4}\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_3\right) - \hat{\mathbf{z}} \times \mathbf{g}\,\frac{2\hat{\mathbf{g}}\cdot\mathbf{b}_4}{\left(\hat{\mathbf{z}}\times\hat{\mathbf{b}}_3\right)\cdot\mathbf{b}_4}\left(\hat{\mathbf{z}} \times \hat{\mathbf{b}}_3\right) - \hat{\mathbf{g}}\right] \tag{4.89}$$

$$J_{c,5}^{-1} = [I_{3X3}\,-\hat{\mathbf{z}} \times \mathbf{g}\,-\hat{\mathbf{g}}] \tag{4.90}$$

Computing the determinant of a 5x5 matrix is a rather long process that requires 145 additions and 120 multiplications on the elements of the matrix. Performing this computation analytically would be tedious. Instead, an inspection of each individual element of the matrix is performed to derive the conditions for which those elements tend toward infinity. If some matrix elements are infinite, the determinant of the matrix is likely to be also infinite.

By inspection of the leg inverse Jacobians $J_{l,i}^{-1}$, the two conditions in which serial singularities occur are

$$\hat{\mathbf{l}}_{2,i} \times \mathbf{l}_{1,i} = 0 \tag{4.91}$$

or

$$\hat{\mathbf{l}}_{2,i} \times \hat{\mathbf{s}}_{i,1} = 0 \tag{4.92}$$

which both lead to a zero denominator for the fraction. The first condition corresponds to a configuration where a leg is fully extended of fully folded and the second condition corresponds to a configuration where the parallelogram unit is fully horizontal. Those conditions correspond to the limit of the workspace on a single leg. The workspace limits of the whole parallel robot are formed by combination of the five leg workspace and depend on which leg reach its limit first in each direction.

When one of the legs of the PentaG reaches one of the serial singularity conditions, the properties of serial singularity apply. The impossible end-effector velocity outputs for the first and second conditions are given by $\dot{\chi} = \left[\hat{\mathbf{l}}_{1,i}\,0\,0\right]^T$ and $\dot{\chi} = [\hat{\mathbf{s}}_{i,1}\,0\,0]$ respectively. Those vectors also correspond to the forces that can be applied on the end-effector which can be balanced with no torque applied on the motors. In both cases, the null space of the Jacobian corresponds to a non-zero velocity for actuator i while the four remaining actuators are locked. No forces on the end-effector can balance a torque on motor i for those configurations. Those serial singular configurations correspond to the boundaries between two working modes. For the first condition, the changing mode occurs when the base link of the leg goes inward the base of the robot instead of outward when leaving the singular configuration. For the second condition, the changing mode concern the parallelogram unit for which the long links will cross each other. In practice, the second condition cannot happens since the angular values of the parallelogram joints are mechanically limited. The first condition is possible and happened often during manual handling of the robot.

It may be tempting to apply mechanical locks on the leg actuator to prevent the reach of serial singularities as it is the case for the parallelogram units. It is however impossible since the actuator values for which serial singularities occurs are coupled and the singular values of a single actuator depends on the position of the four other actuators. By imposing mechanical locks on the actuators to avoid serial singularities, one will greatly reduce the reachable workspace of the robot. In other words, in the actuator space, the maximal singularity-free 5 dimensional hyper-cube for which the boundaries would correspond to actuator mechanical locks is much smaller than the more complex-shaped region corresponding to the full workspace of the robot.

Unlike classical parallel robots, for which serial singularities are dependent of the configuration of the robot legs, parallel robots with configurable platforms have an additional type of serial singularity that is related to the platform. By inspection of the terms of the inverse Jacobian of the configurable platform $J_{c,i}^{-1}$, one can notice that infinite values for leg $i = 2$ and $i = 4$ occur for the following condition:

$$\hat{\mathbf{b}}_1 \times \hat{\mathbf{b}}_2 = 0 \tag{4.93}$$

or

$$\hat{\mathbf{b}}_3 \times \hat{\mathbf{b}}_4 = 0 \tag{4.94}$$

These conditions correspond to alignment of the b links of the platform and represent the limits of the platform workspace when the platform is taken alone. When the platform reaches one of these two condition, the PentaG robot is in a serial singular configuration. The impossible end-effector velocity output corresponds to an extension of the platform in the direction of the aligned b links. This direction also corresponds to the forces that can be applied on the end-effector which can be balanced with no torque applied on the motors. The null space of the Jacobian corresponds to non-zero velocity for actuator $i = 2$ for the first condition or $i = 4$ for the second condition while the four remaining actuators are locked. No forces on the end-effector can be balanced by the motor torques for those configurations. Those serial singular configurations correspond to the boundaries between two working modes of the configurable platform.

Unlike the serial singularities of the legs, the serial singularities of the platform can easily be prevented mechanically without too much sacrifice of the platform workspace. Instead of constraining the angular values of the platform joints, one can constraint the distance between leg attach point 1 and 5 and leg attach point 3. This was done for the robot by simply attaching a thin wire between those links. The wire gets fully extended before the alignment of the b links and is preventing the platform to reach those positions. The reason why the workspace is not greatly affected by this mechanical constraint is because the constraint is imposed in the end-effector Cartesian space and not on the joint space.

Since large forces can be applied on the end-effector near the leg serial singularities and that those singularities are located at the limit of the workspace, they are generally easy to avoid with the robot controller. This is not the case of parallel singularities that can be found in the middle of the workspace.

4.4.2 Parallel Singularities of the PentaG

By definition, parallel singularities occur when the determinant of the inverse Jacobian matrix is zero. An investigation of the individual elements of the inverse Jacobian matrix can be performed in search of conditions that make the determinant to vanish. This method doesn't guaranty that all parallel singularities can be found since some non zero elements could cancel each other in the computation of the determinant and lead to a parallel singular configuration but at least a subset of the possible parallel singularities can be obtained.

The leg inverse Jacobian $J_{l,i}^{-1}$ is a 1x3 matrix that multiplies the 3x5 platform Jacobian matrix $J_{c,i}^{-1}$ to form the ith line of the full Jacobian matrix J^{-1}. The numerator of $J_{l,i}^{-1}$ is given by $\hat{\mathbf{l}}_{2,i}^{T}$. This means that if the leg attach point c_i moves in a direction perpendicular to $\hat{\mathbf{l}}_{2,i}^{T}$, no angular velocity is produced on actuator i. If this happens simultaneously for each leg attach point, the robot is in a parallel singular configuration. The geometrical condition is expressed as follows:

If $\hat{\mathbf{l}}_{2,i}^{T}$ is the orientation of the long links of the parallelogram unit i and $\mathbf{c}_i$ is the position of the attach point between the parallelogram unit and the configurable platform, parallel singularity of the PentaG occur when the configurable platform can move in such a way that it produces on all leg attach points $\mathbf{c}_i$ either no movement or a movement in the plane perpendicular to $\hat{\mathbf{l}}_{2,i}^{T}$.

Although this statement gives us a geometric interpretation of the parallel singularities of the PentaG, the actual configurations for which this condition is satisfied are difficult to compute. In order to get a better insight of the position of these parallel singularities, the determinant of the inverse Jacobian can be compute numerically as a function of the end-effector position over the whole workspace. Figure 4.12 shows the computed value of the determinant of the inverse of the Jacobian matrix in the XY plane for a given height, orientation and grasping. The geometric dimension of the robot are the one obtained from the geometric optimization presented in chapter 5. The absolute values of the determinant for a given point is represented by a logarithmic color scale starting from blue form small values to red for high values. On this graphical representation, one can notice that the absolute value of the determinant tends toward infinity at the limits of the workspace, corresponding the serial singularity configuration described in section 4.4.1. Also noticeable is the presence of values that tend toward zero, and represented by the two blue continuous curves. These configurations correspond to parallel singularities and are unfortunately located inside the workspace of the robot. One of the main objective of the geometric optimization presented in chapter 5 is to move those parallel singularities away from the centre of the workspace.

For almost all combinations of height, orientation and grasping, these two lines appear at various distances from the centre, suggesting that they are of the same nature. The null space of a singular inverse Jacobian matrix represents the direction for which the control of the end-effector is lost. Computation of the null space of a large number of singular configurations over the whole workspace reveals that all of them correspond to the condition for parallel singularity defined above.

Figure 4.13 shows one particular singular configuration. In this configuration, the null space of the inverse Jacobian is a rotation of finger tip #2 around finger tip #1 which produces no motion on leg attach points $i = 1$ and $i = 5$ and produce a motion perpendicular to $\hat{\mathbf{l}}_{2,i}^{T}$ for leg $i = 2$, $i = 3$ and $i = 4$. This example was chosen because of its simplicity, since the control is lost in a pure rotation. In general, the degree of freedom that is lost in a parallel singular configuration is a combination of 5 DOF used to describe the position of the end-effector.

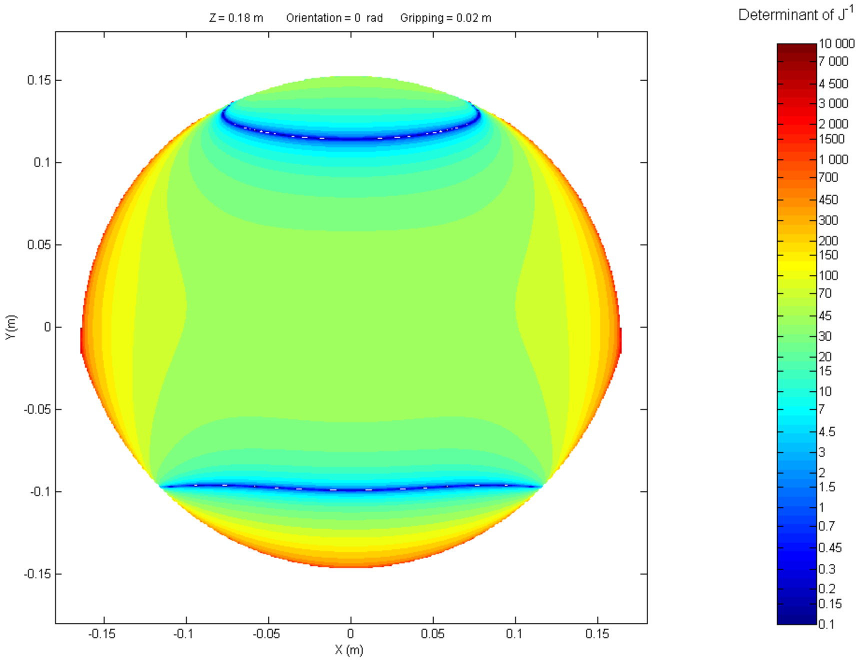


Figure 4.12: The value of the Determinant of the inverse of the Jacobian in the XY plane for a given height, orientation and grasping

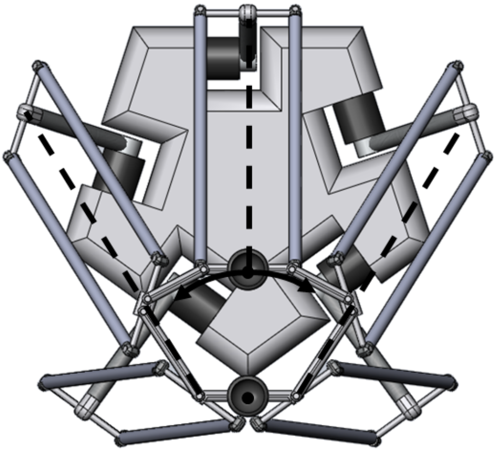


Figure 4.13: Top view of one parallel singular configuration of the PentaG robot. In this particular configuration, the 5 base-located motors can't prevent a rotation of the upper finger tip around the lower finger tip of the configurable platform.

4.5 Summary

In this chapter, some aspects of the kinematic analysis of the PentaG robot were presented. Due to its unique configurable platform, classical and standard methods for kinematic analysis cannot be directly applied to this robot. A method based on static analysis was used to compute both the inverse position kinematics and the inverse Jacobian. In both cases, the problem must be divided into two sub-problems where the kinematics of the platform must be computed before computing the kinematics of the legs. This two step method can easily be generalized to other robots with configurable platforms. Direct position kinematics was solved using a numerical procedure while the direct Jacobian matrix can be obtained by numerical inversion of the inverse Jacobian matrix in non singular configurations, and all working modes were enumerated.

Taking advantage of the fact that the platform of the robot is constrained in the XY plane, a novel static balancing method was proposed that reduces the number of springs and additional links needed while achieving perfect balancing. The method is not only applicable to the PentaG, but also to any translational parallel robot including the Delta robot. Due to the friction added to the system by the balancing system, it was chosen to not apply this balancer to the haptic version of the PentaG prototype but to use it for a pick-and-place version of the robot.

Input-output singularities of the Jacobian matrix were analyzed. Serial singularity conditions were derived and a new type of serial singularity, specific to parallel robots with configurable platforms was described. Conditions for parallel singularity of the PentaG were also described and numerical analysis of the robot workspace was performed to locate those singularities. These conditions can also be generalized to other parallel robots with configurable platforms. Unfortunately, in the case of the PentaG, parallel singularities are present inside the robot workspace and a careful design of the robot dimensions should be performed in order to reduce the influence of these singularities. This is the subject of the next chapter.

Chapter 5

Optimal Kinematic Design of the PentaG

5.1 Introduction

Geometric optimization of the design parameters is a fundamental step in the design process of parallel robots and follows naturally after the choice of mechanical architecture (Chapter 3) and the analysis of its kinematics (Chapter 4). By their nature, parallel robots are faster, stiffer, and stronger than their serial counterparts but it is well known that parallel robots suffer from smaller workspace which may contain parallel singularities[26]. As it was shown in Chapter 4, the performances (e.g. workspace, dexterity, force transmission and stiffness) of the PentaG, like most of parallel robots, are very sensitive to the geometric dimensions of the robot. In general, optimization and customization of design parameters of parallel robot is mandatory to fully exploit the merits of parallel architectures[4]. It has been said that the lack of good optimization process in the design of early parallel robots, i.e. prior to 1990, had lead to many poorly conceived parallel robots with bad performances, which has harm their reputation and delayed their use in the industry as a interesting alternative to conventional design. Fortunately, especially since the last two decades, parallel robot designers have reasserted the fundamental importance of geometric optimization on the robot performance, and this constitutes currently one of the largest topics of research in the field, (see for instance [8], [25],[71], [47], [48]). As opposed to serial robots, geometric optimization of parallel robots is much more complex. Two main reasons make this optimization process rather difficult.

The first problem is the number of design parameters involved. Parallel robots have much more links and joints than serial one and, in general, may have tens of geometric parameters that can be considered as design variables. Each parameter can have a high influence on the performance of the robot and their combined influence is generally highly coupled.

The second problem that is commonly encountered is the high complexity of the objective function. Most performance criteria are based on properties of the Jacobian matrix, such as its condition number, which is generally impossible to compute in an analytic way. The performance is not only dependent on the geometry of the robot, but also on the position at which the robot stands. In order to deal with a performance index that only depends on the robot geometry and not on its position, global performance index must be computed from the integration of local performance index over the complete workspace. It is therefore generally computer intensive to evaluate a robot performance index for a given geometry. In the optimization process, the objective function created by the performance indices will generally have a large number of local optimum and large number of discontinuities over the space of design parameters.

This chapter presents the optimization procedure that was used for the geometric design of the PentaG and the techniques developed to overcome the problems previously mentioned. A first original contribution is the enumeration of all the geometric parameters of the PentaG based on a novel modified version of the Denavit-Hartenburg notation[15] adapted to closed-loop mechanisms. The set of geometric conditions that are necessary to close the mechanical loops of the PentaG and ensure that the mechanism has the approriated mobility are then presented. General design guidelines and principle are then presented and described. These principles are used to reduce the number of design parameters to a small set of critical parameters that can be efficiently used in the optimization algorithm. In the second section, several objective functions are considered for the optimization and their respective advantages and disadvantages are discussed regarding the PentaG robot and its application. Using the sign of the determinant of the Jacobian matrix as an indicator of the presence of a singularity between two given points, a fast algorithm is developed to compute the volume of the centered singularity-free workspace zone for a given set of design parameters. The advantages and restrictions of a selection of search algorithms that are commonly used in parallel robot optimization and the relevance of choosing Genetic Algorithms for the geometric optimization of the PentaG are then discussed. The results obtained from the optimisation process are then presented and analysed.

5.2 Kinematic Parameters

This section presents the geometric parameters of the PentaG, i.e. the parameters that are involved in the kinematic relations between all the links of the PentaG architecture. As previously mentioned, the kinematics of a mechanism describe the motion of the links without consideration of the cause of this motion. For this reason, only the relative positions of the kinematic pairs (i.e. joints) of the mechanism are needed to describe its kinematics, and the parameters describing the particular shape of the rigid bodies (such as the parameters involved in their inertia matrix) are considered in the dynamic model but are not relevant for the kinematic analysis.

A kinematic pair is a connection between two bodies that imposes constraints on their relative movement. The geometric design parameters of the PentaG define in which way all the links and joints of the robot are assembled. The goal of this section is to define all the possible independent geometric parameters of the PentaG and then reduce this number to a minimal set of parameters that are critical to the robot performance. The elegant Denavit-Hartenberg notation [15] can be used to define the maximum set of independent geometric parameters that influence the kinematic of a serial robot. Normally, 6 independent parameters are needed to define the relative position of two links connected by a joint. By introducing conventions on the location of the reference frames on the successive links of a serial robot, the Denavit-Hartenberg notation uses only 4 independent parameters to define uniquely a reference frame on each link. The four parameters

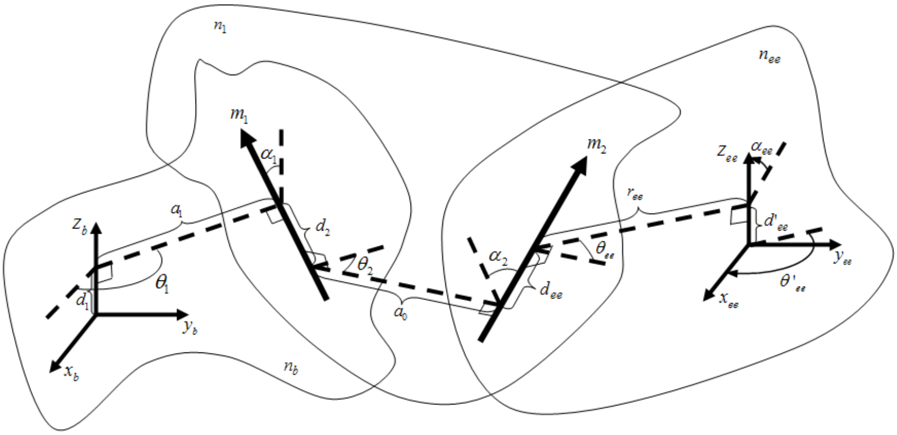


Figure 5.1: Denavit-Hartenburg convention for serial robots. 4 parameters are used to describe the position of a joint axis relatively to the axis of the previous joint.

are derived from the fact that prismatic and revolute joints can be described by a line of translations or rotations, and only four parameters are needed to uniquely describe a line in 3D space. The same principle that is used in the Denavit-Hartenberg notation can also be used to compute the minimum set of kinematic parameters of a serial chain.

5.2.1 On the Number of Kinematic Parameters

In order to determine which subset of kinematic parameters should be considered in the geometric optimization, it is interesting to first have a look at the whole set of geometric parameters that have an influence on the kinematics. The problem of finding the minimum but complete set of kinematic parameters for serial chains has been the subject of several articles [19, 58, 69, 21]. Four independent parameters are needed to describe a line in 3D space, and 6 independent parameters are needed to describe a second reference frame. For a serial chain with m revolute joints, and a reference frame located on the base and the end-effector, the number of independent kinematic parameters P is given by

$$P = 4m + 6 \tag{5.1}$$

since four independent parameters are needed to describe the position of a line relatively to the base reference frame. Those parameters are used in the Denavit-Hartenburg convention for serial robots. Figure 5.1 shows the original DH parameters for a serial robot with two joints and one rigid link between the base and the end-effector. This is only one of the DH conventions that can be used. Some authors find it more intuitive to define θ_i as the angular position of the joint i instead of the joint $i-1$. No matter which version of the DH parameter convention is used, the total number of parameters is the same.

For general closed-loop mechanisms, there is no equivalent and generally accepted convention, although some interesting conventions have been suggested recently[76]. A notation for the geometric parameters of closed-loop mechanisms based on an extension of the DH notation that could be applied to any mechanism is here proposed.

In the original DH notation, m of the $4m$+6 parameters are not strictly speaking kinematic parameters that depend on the geometry of the serial robot but are rather values of the joint position, depending on the configuration of the robot. If the mobility of the robot is defined as M, the number of independent kinematic parameters i.e geometric parameters that are not dependent on the configuration of the serial robot is given by

$$P = 4m + 6 - M \tag{5.2}$$

In a closed-loop mechanism, dependency is created between the values of the joints and overconstraints create also dependency between the kinematic parameters. No matter if the dependency is created on the joint values or on the kinematic parameters values, each independent closed-loop removes 6 independent parameters.

The PentaG mechanism is a closed-loop mechanism with $m = 43$ joints and $E = 2$ end-effectors and $M = 5$ DOF. The number of independent parameters is given by

$$P = 4m + 6E - M = 172 + 12 - 5 = 179 \tag{5.3}$$

The PentaG mechanism has 10 independent closed-loops which create $6 \cdot 10$ dependencies on the parameters. The kinematic parameters can be separated from the mobility parameters in order to establish the maximal number of independent kinematic parameters that can be used for the geometric optimization of the PentaG robot. If all the joints of the mechanism are disassembled, each link possesses either the inner bearing part or the outer bearing part of the connected joints. These will be called attach points. In a disassembled link i, the number of kinematic parameters needed to define the axis of the attach points is given by

$$P_i = 4m_i + 6\,(RF_i - 1) \tag{5.4}$$

where m_i is equal to the number of attact points and RF_i is the number of reference frames on the disassembled link. On each attach point axis, a reference point can be defined where the common perpendicular d intersect with the axis. When the mechanism is reassembled, one additional kinematic parameter is needed per joint to define the distance between the reference points of the two attach points.

These parameters define the kinematics of the robot and can be used in an extended version of the Denavit-Hartenburg convention that include closed-loop mechanisms. In closed-loop mechanisms, some of the links have more than two joint connections and more parameters are needed as shown in Figure 5.2. In table 5.1, the links of the PentaG in regard of their number of attach points and the number of parameters needed to define all attach points on each link are listed. 158 kinematic parameters are needed to define all joint attach point axes on the $n = 34$ links. In addition, $m = 43$ kinematic parameters are needed to define the assembly of the joints on the attach point axes. The total of kinematic parameters is therfore 158+43= 201. It was already established that the PentaG mechanism has 22 overconstraints which introduce dependencies on the kinematic parameters. Among the 201 kinematic parameters, 22 are dependents and 179 are independents, as predicted by Equation 5.3.

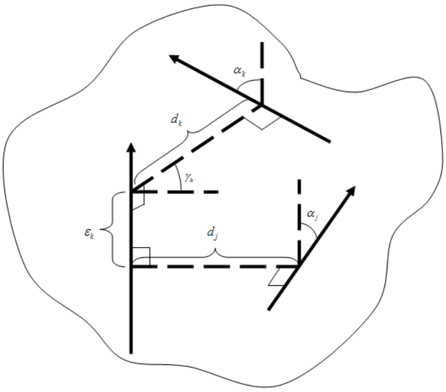


Figure 5.2: A link with three joint attach points. $4m_k + 6\,(RF - 1) = 6$ kinematic parameters are needed to defined the relative position of the three attach points.

Number of attach points	Number of reference frames	Parameters per link	Number of links	Total Number of Kinematic Parameters
2	0	2	19	38
3	0	6	12	72
3	1	12	1	12
4	1	16	1	16
5	1	20	1	20
Attach point parameters (1 per joint)				43
Dependencies due to overconstraints				-22
Total				179

Table 5.1: Geometric parameters of the PentaG

5.2.2 Geometric Conditions for the Feasibility of the PentaG

Among the parameters presented in section 5.2.1, only a certain combination of them will produce a mechanism with the mobility of the PentaG. These geometric conditions insure that the robot can be assembled and that the two end-effectors will have the desired full cycle mobility relative to the base, apart from singularities. The list of geometric conditions for the feasibility of the PentaG are:

- The axes of rotation of the 5 base actuators ${}^{i}m_1$ must lie in a plane.
- The axes of rotation of the second joint ${}^{i}m_2$ of a leg i must be parallel to the axes of rotation of the base joint ${}^{i}m_1$.
- The axes of rotation of the 4 joints in a parallelogram unit must be parallel to each other and perpendicular to the second leg joint ${}^{i}m_2$.
- In the parallelogram unit, the length of ${}^{i}n_3$ must be equal to the length of ${}^{i}n_4$ and the length of ${}^{i}n_2$ must be equal to the length of ${}^{i}n_5$
- The axes of rotation of the leg third joint ${}^{i}m_7$ must be perpendicular to the joints in the parallelogram unit.
- The axes of rotation of the 8 joints ${}^{p}m_j$ on the configurable platform must be parallel to each other and perpendicular to the third joint of the leg.
- The sum of vertical offsets of the attach points in the parallelogram units and the configurable platform must be zero in order to close the loop.

Theoretically, all the remaining geometric parameters are free of choice but in practice, several of them are not considered as variable design parameters because of technical considerations or simplicity. For example, the vertical offsets d of the joints ${}^{p}m_j$ of the configurable platform relative to leg attach point ${}^{i}m_7$ on link ${}^{i}n_5$ are equal for the 8 joints. On the configurable platform, a decision was made for simplicity but also for technical considerations to put the leg attach points and the platform joints in a straight line.

5.2.3 Reduction of the Number of Parameters

As mentioned in the introduction, the geometric optimization of parallel robots is a tedious task because the number of parameters is large and because the computation of the objective function is complex and computational intensive. It will be quite difficult to manage an optimization on the more than 100 parameters that define the geometry of the PentaG robot. A first task in the optimization process is therefore to reduce the number of parameter to a minimum set of critical parameters.

There are no direct guidelines to reduce this number of parameters and one must use some common sense and insights about the geometry of the robot to achieve this goal. Here is a summary of the key techniques and principles that have been used in the case of the PentaG.

- Removal of the parameters that don't influence the position and velocity kinematics

Some of the geometric parameters are not used in the kinematic relations between the end-effector and the base of the robot. Therefore, many performance aspects, such as the dexterity, static and workspace, will not be influenced by those parameters. For example, in the case of the PentaG, the length of the parallelogram units are important but not their width from a kinematic point of view. The vertical thickness of the configurable platform also doesn't influence the velocity kinematics but it influences the position kinematic. These parameters have been removed from the vector input of the optimization process although their values must still be considered in the final design for technical reasons.

- Symmetry of the design

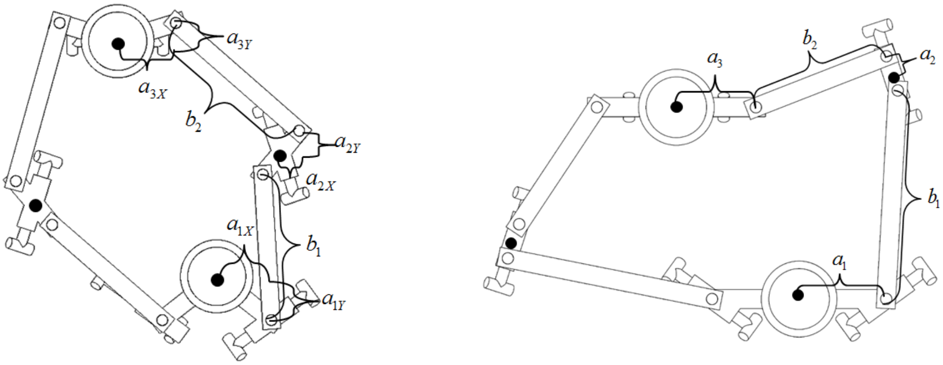
The number of parameters is greatly reduced when symmetry is applied to various geometric elements of the robot. One of the first decisions that were taken was the use of five identical legs, although the legs play different roles on the constraints of the configurable platform. The position of the actuators was also chosen to present a circular symmetry. This simplifies greatly the design since the optimization has to be performed on only one leg. The configurable platform also has a full symmetry in the YZ plane so the number of design parameters in the platform is reduced by half. The use of symmetry offers also a side advantage. In a haptic device, or in any parallel robot in general, designers aim at a rather homogenous performances over the workspace so they can expect the same performance for any position of the robot. Symmetrical design generally leads to more homogeneous performance.

- Technical considerations and simplicity

Several of the possible design parameters consist of offsets between the positions of the joints and are not desirable from a technical point of view or in regard to secondary performance criteria. For example, in the initial geometric optimization of the PentaG, offsets were allowed in the plane of the platform for the position of the platform joints. A larger number of parameters offer more design possibilities and some of the considered platforms with offsets had good kinematic properties but since the offsets may lead to an increase in the bending stress in the platforms and also in an increase of the total number of parameters, the decision was made to align the platform joints with the leg attach points. Figure 5.3 shows the platform parameters with offsets and without offsets. Three platform parameters have been removed due to this decision.

Another example can be used to illustrate the principle of simplicity. In pilot optimizations, the axes of rotation of the actuators were not necessarily tangent to a circle around the centre point on the base. In other words, the actuator could be located at a offset distance from this tangent point meaning that the parallelogram units will not face the centre point of the base as it is the case for some Delta robots used in the industry. Since this parameter is not of great impact on the kinematic performance, it was removed from the parameter list in the latter optimizations.

Finally, by using the principle of symmetry, simplicity, and relative influence on kinematics, the total number of parameters to optimize have been reduced to only 8. The set Γ of the geometric parameters to optimize is



a) b)

Figure 5.3: The parameters of the configurable platform a) with offsets and b) without offsets

$$\Gamma = \begin{bmatrix} m \\ l_1 \\ l_2 \\ a_1 \\ a_2 \\ a_3 \\ b_1 \\ b_2 \end{bmatrix} \tag{5.5}$$

The first three parameters are related to the dimensions of the legs and are shown in Figure 5.4. The next five parameters are related to the dimensions of the configurable platform and are shown in Figure 5.3 b). The optimization algorithm should find the set of geometric parameters that best fits the objective function. Technical aspects are considered to include upper and lower limit constraints to the vector Γ . This is the subject of the next section. The remaining of the 174 parameters described in Section 5.2.1 are set to satisfy the mobility and assembly condition of the PentaG or have been removed from the optimization input during the parameter reduction process.

5.2.4 Upper and Lower Limits of Parameters

Given the set of geometric parameters that should be optimised, a multidimensional parameter space can be created where each dimension represents a parameter and each point within this space represents a unique geometric design of the robot. Clearly, these dimensions are not infinite and practical considerations for the design of the robot are used to define a lower and an upper limit for each parameter. The choise of the input parameters and their limits is the first step of the optimization procedure but in practice, the optimisation process is highly iterative between each step, as it is the case for complete design of the whole robot. Some of the early results of the optimization were interesting from workspace and kinematic point of view but were not practically realisable from a technical point of view. However, these pilot optimizations were very useful to set appropriated limits on the input parameters that were used in the final optimization. The upper and lower limits should exclude the parameter values that are clearly unwanted but should offer a range that is large enough to maximize the design possibilities. Since all

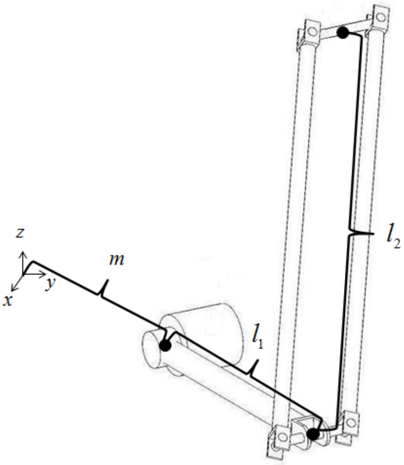


Figure 5.4: The three geometric parameters of the leg

the robot parameters can always be scaled up or down together to create a bigger or a smaller robot with the same relative workspace, the centre of the workspace of the robot was chosen to be at 150 mm above the centre point of the base. This dimension acts as a reference dimension for all other dimensions. The upper and lower limits that have been used in the final optimization are given below in mm.

$$\begin{aligned} m &= [50, 120] \\ l_1 &= [70, 120] \\ l_2 &= [150, 200] \\ a_1 &= [13, 30] \\ a_2 &= [3, 15] \\ a_3 &= [10, 30] \\ b_1 &= [25, 80] \\ b_2 &= [25, 80] \end{aligned} \tag{5.6}$$

The first three parameters m, l_1, and l_2 represent respectively the distance between the actuators and the centre of the base, the length of the base link of the leg and the length of the parallelogram unit. Their upper limits were set to ensure some compactness of the robot. Additional constraints were imposed on the maximum length of the base link since the force that the actuators exert on the robot is inversely proportional to l_1. The lower limit of m was set to give enough place on the base to mount the actuators and the lower limits on l_1 and l_2 were set to ensure that the legs can reach the platform in various positions.

The last five parameters represent the length of the links in the configurable platform. The parameters a_1, a_2, and a_3 represent links that are connected to the legs and are constrained in rotation while the parameters b_1 and b_2 represent the links that connects the legs together and are allowed to rotate around a vertical axis. The upper limits were set for compactness of the configurable platform. Since the links of the platform are the furthest from the base, their combined influence on the total inertia is higher than the influence of the legs and therefore a compact platform is desirable. It was first considered to use negative values for the lower limits of a_1, a_2, and a_3. Indeed, some design solutions in which the links defined by b_1 and b_2 were crossing each other resulted in interesting workspace and kinematic properties. But in order to built such a robot, the platform links should have been placed in different horizontal planes which would have induced bending stress in the whole platform. The lower limits were therefore set to positive values and were normally defined by the minimal space possible between two joints or between a joint and a finger tip. It was also considered to proceed to two separated optimizations for the legs and for the configurable platform. Since the dimension of the legs and the platform seemed to have a highly coupled influence on the robot workspace and kinematics, it was finally decided to carry out the optimization with the 8 parameters together.

5.3 Objective Function

One of the early steps of the optimisation process is to define a list of requirements that is expected from the robot according to the task. This step should normally be done before the reduction of the number of parameters described in Section 5.2 since the performance indices that will be used in the optimization process will influence which geometric parameters are critical and which parameters are less important and can be removed in the parameter reduction process. Many performance aspects can be considered such as the stiffness, dynamics or inertia but the two aspects that are by far the most used in parallel robot optimization are the volume of the workspace and the dexterity of the robot. Various performance indices have been proposed to quantify the performance of a specific design in a single number [54, 47].

Among the most popular is the "global condition index" proposed by Gosselin[?]. In section 4.4, it was shown how the condition number of the Jacobian matrix characterizes the dexterity of the robot. The global condition index is defined as the local condition number integrated over the workspace and divided by the volume of the workspace. Although useful, there are some drawbacks in the use of the global condition number as a criterion for the optimum design. A first drawback is that the computation of the integral of the condition number over the workspace is almost always impossible to do analytically and numerical methods must be used in which the workspace is sampled and the average value and the condition number is computed from the sampled configurations. The result will depend on the density of the sampling and since the workspace is multidimensional, a dense sampling will be very computer intensive. The second drawback is that the global condition index doesn't tell us how uniform and smooth is the dexterity of the robot over the workspace. A second global conditioning index based on the ratio of the maximum and minimum values of the condition number should be also used. As described in Section 4.4, singular configurations occur when the condition number becomes infinite and the actuators lose control of the end-effector in at least one direction. The transition between a dexterous position and a singular position can be very abrupt and a less dense sampling may fail to detect the singularities within the workspace.

A second type of performance indices that are often used is the workspace performance index. The main criteria used here is the total volume of the workspace. It should be noted that for a robot with at least one rotational DOF, the boundaries of the workspace are usually coupled between the translational and rotational DOF and arbitrarily equivalent units must chosen between translational and rotational units when the total volume of the workspace is computed. Some authors use the total orientation workspace [52], which includes all the positions of the end-effector that may be reached with all the orientations within a certain ranges of the orientation angles, as performance index.

In a general parallel robot, like a machine-tool robot, the presence of singularities within the workspace can be tolerated because the controller is fully in charge of the position of the robot and can avoid them. Regions of the workspace which are close to singularities must be avoided since they result in poor force transmission performance, in the sense that the device cannot apply forces at the end-effector in certain directions. In the case of haptic devices, the operator is in charge of the position of the robot and since it is not possible to predict which position he will want to reach, the workspace in which he operates must be singularity-free. For haptic applications, the useful workspace is defined as a centered singularity-free region, where the user can safely operate the robot in every DOF allowed by the architecture. This region should have a shape that is simple enough so the operator can recognize easily which zones are inside and outside the useful workspace. For safety, the haptic device should prevent the operator to move outside the useful workspace either with mechanical locks on the device or with a force from the actuators that push the operator away from the useful workspace limits.

Various approaches can be used to optimise the geometry of the robot according to the various performance indices. Some authors use a weight function of different performance criteria to create a single criterion[46]. The problem with these methods is the arbitrarily choice of the weight of each performance criteria and the additional computations that come with the evaluation of multiple performance criteria, each of them being already rather computer intensive. A more efficient approach is to use a single performance index that is considered fundamental in the global performance of the robots and uses secondary performance indices to choose between solutions that are obtained from the optimization that has been performed with the primary performance index.

The main innovation in the PentaG is the use of a configurable platform that can provide grasping capabilities while all the actuators are located on the base. The robot can offer multi-point contacts without moving around the weight of the gripper motor. This results in a device with a higher structural stiffness, resulting in increased transparency for the whole haptic system. The choice of the geometric parameters has a direct impact on the total moving mass of the robot. For a same workspace size, a more compact robot will have lower moving mass. The 8 geometric parameters described in Section 5.2 can be scaled up or down all together to produce smaller or bigger robots with corresponding workspace size. Given a certain workspace that is suitable for the human ergonomic, the goal of the optimisation is to increase the compactness of the robot. The use of compactness as the primary performance criterium for the optimization of the PentaG is in line with the primary requirement to keep the robot as light as possible. Compactness was already identified as one of the most important criteria for haptic devices based on parallel architectures[82]. Several secondary requirements are also considered ranging from global dexterity, stiffness and technical feasibility.

5.3.1 Maximum Useful Workspace

In the previous section, the useful workspace (UW) for a haptic device was defined as a singularity-free region with a simple geometry in the middle of the total workspace. The basic shape chosen for the useful workspace of the PentaG robot is the half-ellipsoid. An ellipsoid is the three dimensional analogue of an ellipse. If the ellipsoid is centered on the origin of the coordinate system, its surface is defined by the equation

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \tag{5.7}$$

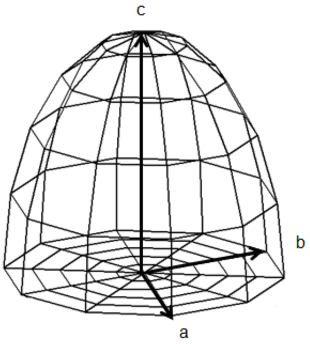


Figure 5.5: The half ellipsoid and its principal axes

where a, b, and c are the length of the semi-principal axes. When $a = b = c$, the ellipsoid degenerates into a sphere. For our purpose, the axes lengths a and b are chosen to be equal and the vertical dimension c to be one and a half the values of a and b. The upper half ellipsoid is simply the part of the ellipsoid where $z \geq 0$. This particular shape was used because it is simple and seems to approximate the best the more complex shape of the actual singularity-free region. Figure 5.5 shows the half ellipsoid and its principal axes. The volume of the useful workspace is given by

$$\begin{aligned} V &= \tfrac{4}{3}\pi abc \\ V &= \tfrac{16}{9}\pi c^3 \end{aligned} \tag{5.8}$$

The centre of the half ellipsoid useful workspace is located at a certain height right over the centre on the base plane. To increase the compactness of the robot, the volume of the useful workspace relative to a certain height of the workspace must be maximised. The height of the workspace is used as a dimensional reference for all the upper and lower limits of the geometrical parameters.

The half ellipsoid only represents the 3 translational DOF of the PentaG. For each combination of rotation and grasping, a different 3D useful workspace is obtained. For the PentaG robot, the complete useful workspace is defined as the half ellipsoid in which the operator can rotate ±45 degrees and grasp from 4 to 8 cm everywhere without encountering a singularity. These values are obtained from ergonomic considerations for the human operators. They will also be useful to define the upper and lower limits of the geometric parameters of the configurable platform.

5.3.2 Computation of the Singularity-Free Workspace

In order to compute the maximal singularity-free workspace volume, the singular configuration which is the closest to the centre point of the workspace must be detected. In a singular configuration, the determinant of the Jacobian matrix is zero. Searching for null determinants by sampling the whole workspace will require a dense sampling since the transition between dexterous region and singular region can be sharp. Furthermore, the robot has 5 DOF, which implies a 5 dimensional sampling that would be computer

Algorithm 5.1 The Pseudo Code of the Useful Workspace Volume

```
Inputs:
        Geometric Parameters (P)
Data:
        height of the UW centre (H)
        sampling points on the boundaries of a unit UW
        rotation and grasping sampling points of a unit UW
        Number of boundaries sampling points (NB)
        Number of rotation and grasping sampling points (NRG)
        Resolution (R)
Outputs:
        Volume of the useful workspace (V)

Initialization:
        Flag = 0
        Z = H
        S = SJD for the UW centre at zero grasping and zero rotation

WHILE (Flag=0) DO
        Compute the boundary sampling points for a UW of height (Z-H)
        FOR i:=1 to NB
                FOR j:=1 to NRG
                        Compute the inverse kinematic (IK) for the point defined by i and j
                        Use IK to compute the local jacobian matrix (J)
                        Compute the sign of the determinant of the jacobian matrix (SDJ)
                        If (SDJ*S<=0)
                                Flag = 1
                        end
                end
        end
        Z = Z + R
end
V = 16/9*pi*(Z-R)^3
```

intensive. A second useful property of singular configurations is that the determinants of the Jacobian on each side of the singularity have opposite signs. For the computation of the useful workspace, we only want to know if a singularity is present between the centre of the workspace and the border of the workspace. In order to obtain the size of the maximal singularity-free half-ellipsoid of a particular design, we compute the sign of the determinant of the Jacobian (SDJ) at various points on the boundaries of the half-ellipsoid and increase gradually the dimension of the half-ellipsoid until one of the boundary points has a SDJ that is different than the centre of the workspace. For each of the boundary points, the computation must be done for a sampling of the rotational values and the grasping values.

The pseudo code of the objective function is given in Algorithm 5.1.

The sampling resolution used for increasing the size of the useful workspace was 5 mm. 25 sampling points on the boundaries of the UW, 6 sampling points for the rotation and 6 sampling points for the grasping were used. The total number of jacobian computation for one WHILE loop is $25 \times 6 \times 6 = 900$. The code was implemented in Matlab and typical time computation for one performance index was around 120 milliseconds on a desktop computer.

5.4 Optimisation Method

As opposed to serial robots, performance indices of parallel robots have generally a much higher variability along the robot dimensions. The performance index that was chosen in the previous section is dependent of the location of the singularities within the workspace of the robot. Beside the fact that the number of parameters to optimise is large, the objective function has a large number of local optima and discontinuities and can vary sharply for small variations of the geometric parameters.

Several optimisation algorithms have been proposed for parallel robots. If the number of design parameters can be reduced to 2 or 3, one may choose to use a brute force approach and compute the performance index for all the sets of design parameters possible and choose the most appropriated. The number of possible designs increases in an exponential way with the number of design parameters. Since the performance of the PentaG is highly dependent on the dimension of both the legs and the configurable platform, it was not appropriate to reduce the number of design parameters to less than 8.

5.4.1 Genetic Algorithms

Genetic Algorithms have been increasingly popular in many fields of engineering because of their highly adaptivity to broad range of optimization problems. The development of Genetic algorithms is attributed to John Holland [35] who developed the method in the 70s. The method is described here briefly. The algorithm mimics the process of natural selection to find a set of solutions that fit the best the objective function. In the context of genetic algorithm, each design parameter represents a “gene” and the vector of 8 design parameters presented in Equation 5.5 represents a “chromosome”. An initial population of chromosomes is formed by randomly assigning values to each gene within the range of their upper and lower limits. The objective function is then evaluated on each member of the population. The algorithm then uses concepts inspired from natural selection, such as elites, crossovers and mutations to produce a new generation of the population. The elites are the part of the population that scored the best on the objective function and are guaranteed to survive to the next generation. The rest of the population is called parents and the new generation is formed by combination of the chromosomes of the parents. The parents that satisfied the best the objective function are used more often in the combination. Finally, mutations are introduced by random variations of some of the genes in the new population. The algorithm is stopped after a certain number of generations or after no more improvement is noticed from generation to generation.

The main advantages of the genetic algorithms are their capabilities to deal with a large number of parameters, optimize the variables on a very complex objective function, and produce a list of optimum parameters instead of a single result. Their main disadvantage is that there is no absolute certainty that a genetic algorithm will find a global optimum. These advantages are very well suitable to the current problem but the lack of confidence that the solution consists of a global optimum should restraint their use to optimization problems in which traditional optimization approaches fail. Genetic Algorithms have been used in optimisation of many parallel robot design parameters as in [72] and [63] for example.

In the case of the PentaG, the specific requirements about the rotation and the grasping translation of the platform geometry makes the objective function highly discontinuous over the geometric parameter space. This is due to the fact that for certain platform geometric parameters, the platform cannot even reach ±45 degrees of rotation and the determinant of the Jacobian matrix cannot be computed. In addition, the locations of the singularities are quite sensitive to the platform dimensions and the number of parameters is rather large. Since the optimisation is done on a single criterion, a set of various solutions is useful since the solution that fits best the secondary requirements can be chosen among the various solutions. For all these reasons, it was decided to use Genetic Algorithms for the optimization of the design parameters.

The stopping criteria used in the genetic algorithm was generations stalling. The average of the values of the objective function over the population of the last generations is evaluated and if there is no improvement, the algorithm stops. Many settings of the algorithm need to be well tuned to produce significant results. The settings that had to be adjusted in this optimisation were

- Upper and Lower Limits of Parameters
- Population size of each generation
- Elites, Crossover, and Mutation ratios
- The number of stall generations and the improvement tolerance

Among those settings, the size of the population is the most influential in terms of quality of the results and time of computation. A large population is more likely to find the global optimum among the various local optima but it is also computational intensive. A good approach is to perform several pilot computations with small populations in order to choose efficient values for the other settings such as the parameters limits and the elites, crossover and mutation ratios.

Typical computation time on desktop computer was around 1 minute for small populations of 10 designs per generation, from 1 to 4 hours for medium population of 50 to 100 designs per generation, and from 1 to 2 days for large population of 500 to 2000 designs per generation.

5.5 Results

Once the algorithm settings were found to be efficient for the problem at hands, larger population sizes optimisations were performed. The goal again was to find an optimum solution for the volume of the useful workspace that also satisfies the secondary performance criteria. Since the algorithm needs a stalling situation to stop, the best solutions are found in the elite part of the last generation. If the algorithm was efficient enough to find a local optimum, the elites should have little difference among them.

Optimisations performed with small populations showed generally noticeable differences among the elites, meaning that the algorithm failed to find a local optimum. This step was however useful to set the design limits and populations ratios. Medium population generally showed homogenous among the elites, which suggests that a local optimum was found. However different randomly starting populations gave sometime different final results, suggesting that the optimum found with medium size population is not a global optimum. Large populations are computational-intensive and may take up to 1 or 2 days of computations. Four runs of large populations have been computed with populations of 500, 1000, 1000, and 1500 individuals respectively, starting from randomly selected populations. The four runs converged to the same result, suggesting that this is maybe a global optimum, but it is not possible to be sure about that. The advantages of Genetic Algorithm are here obvious. A functional solution to an optimization problem, which given its complexity couldn't be solved by classical optimisation method, is obtained. Although the repeatability of the results suggests that an optimum has been found, the nature of Genetic Algorithms prevents us to have certainty about this. The parameters, in millimeters, of the chosen optimal design were

$$\Gamma = \begin{bmatrix} 100 \\ 90 \\ 200 \\ 23 \\ 5 \\ 21 \\ 48 \\ 30 \end{bmatrix} \tag{5.9}$$

It is already known that the optimal design satisfies the secondary objective concerning the technical feasibility. The next step is to ensure that the dexterity of the robot is good enough. The condition number of the Jacobian is a good indicator of the dexterity of the of the robot since it represents how much the output values of the transfer function can change for a small change in the input arguments. Since the dexterity is computed with the condition number of the jacobian, it can also be verified that the useful workspace is indeed singularity-free.

The local dexterity index of the robot can be computed for all positions of the workspace. The workspace is of dimension 5, which makes it difficult to represent it in a graphical form. The technique used here was to separate the dimension of the workspace into continuous dimensions and discrete dimensions. The X and Y dimensions were chosen to be continuous while the values for the height Z, the rotation angle θ, and the grasping p are computed at discrete intervals. Discrete dimensions are represented with five samples each. Figure 5.6 shows the results for the height $Z = 150$. A 5X5 grid is formed by the samples of rotation and grasping. In each element of the grid, the values of the condition number are computed over the XY plane for the corresponding set of height, rotation, and grasping values. The values in the XY plane also have to be discredited for the computation, but a sampling that is dense enough can be used so that the graphical representation of the results approximates a continuous representation. In the following figures, each dimension X and Y have discredited into 2500 samples. The condition number is represented by a color scale from blue, for perfect conditioning to red, for singular configuration. Since the condition number becomes infinite in singular configurations, a logarithmic scale has been used. Figures 5.6, 5.7, 5.8 and 5.9 shows the results for a height of 150, 190, 220 and 250 mm over the centre of the base of the workspace.

It can be noticed first that the total workspace becomes smaller at higher height. This justifies the use of the top half ellipsoid as a natural shape for the useful workspace of the robot. It is also seen that at any height, the singularities are getting closer to the centre point for extreme rotation and grasping. If the task that is intended to be performed doesn't require the full range of rotation and grasping of the configurable platform, the volume of the total workspace in the X, Y and Z coordinates can be increased. Inside the useful workspace, the Jacobian matrix is usually well conditioned with a condition number lower than 10. This ensures a good kinematic transfer between the actuators and the end-effectors of the robot for the chosen workspace. Figures 5.10 and 5.11 show the relative dimensions of the total workspace, the useful workspace and the half ellipsoid.

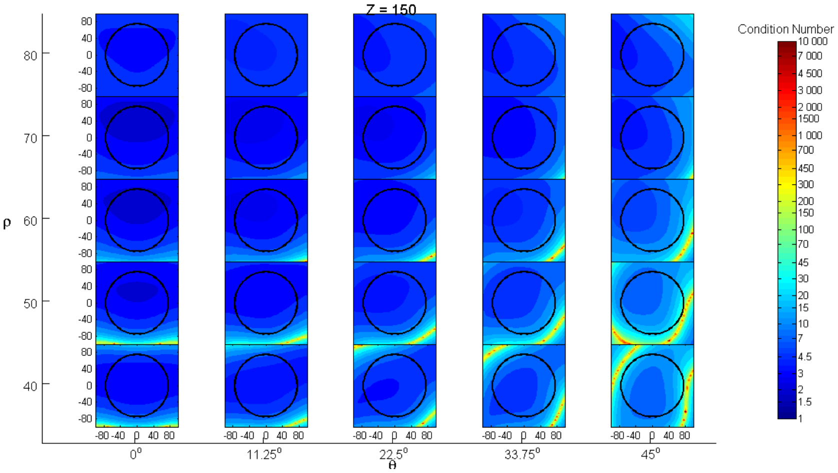


Figure 5.6: Condition Number of the Jacobian matrix at height Z = 150 mm

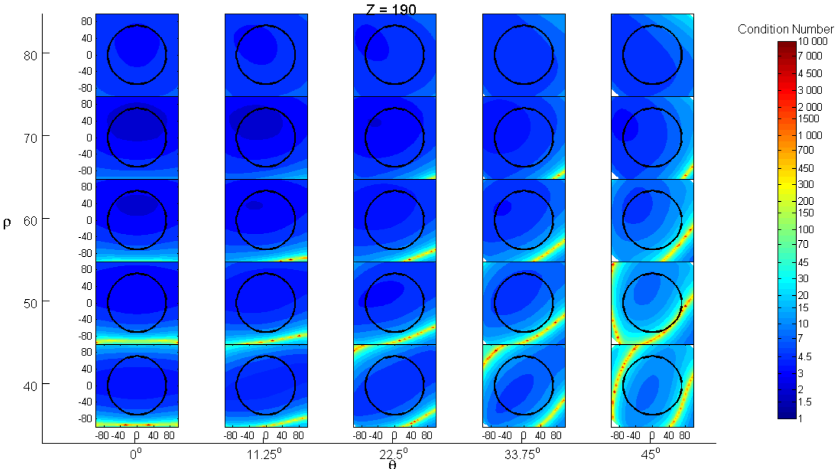


Figure 5.7: Condition Number of the Jacobian matrix at height Z = 190 mm

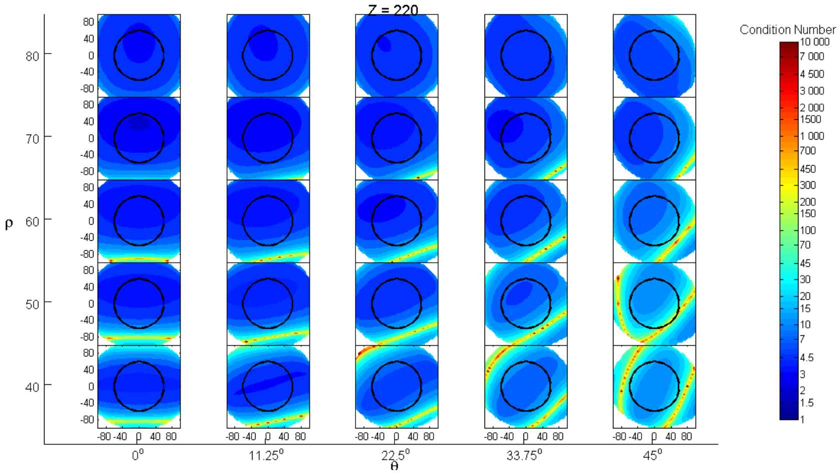


Figure 5.8: Condition Number of the Jacobian matrix at height Z = 220 mm

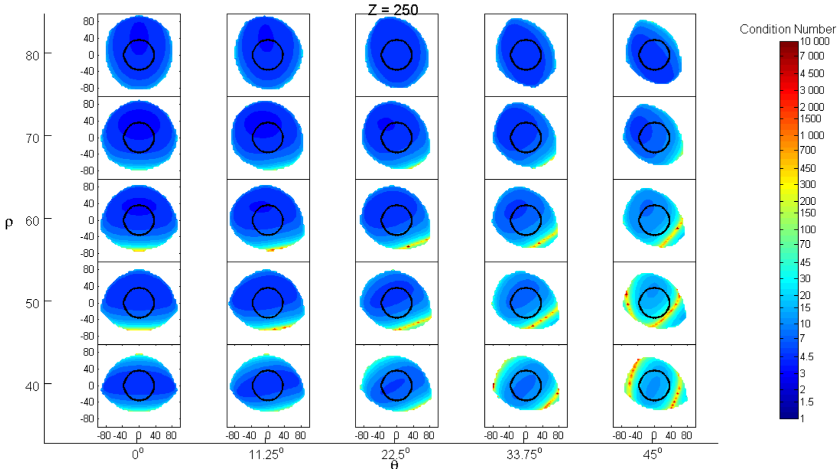


Figure 5.9: Condition Number of the Jacobian matrix at height Z = 250 mm

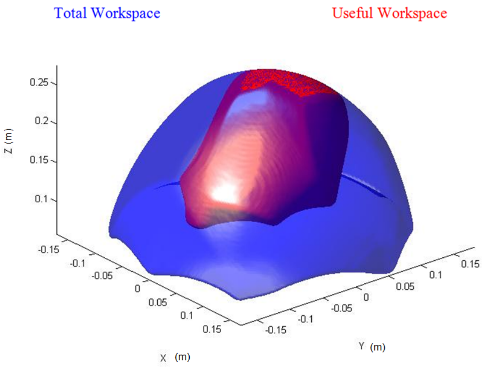


Figure 5.10: Optmised useful workspace in comparison of the total workspace of the robot. At the boundaries of the useful workspace exists one particular configuration of the platform (θ, ρ) that produces a parallel singularity. This occurs generally for maximum rotation $\theta = \pm 45^o$.

5.6 Summary

In this chapter the optimisation procedure that has been used for the geometric design of the PentaG is presented. The problem at hand was quite complex due to the number of parameters and the complexity of the objective function. A general notation for mechanisms, which was based on the Denavit-Hartenburg convention and that can be used to represent any closed-loop mechanism, was presented . Based on this notation, a set of general geometric parameters was defined and these parameters can be used to uniquely define a particular geometry of the PentaG. The conditions on the general parameters, for which the PentaG can be assembled and has the desired mobility, have been enumerated. General guidelines and principles used to reduce the number of parameters to a smaller set that is manageable in an optimisation algorithm were then presented. Due to the importance of compactness and avoidance of singularities in haptic devices, it was chosen to optimise the geometry of the robot for a maximum singularity-free workspace. Inside this workspace, a simple shape was defined; a half ellipsoid, that can be easily recognizable by the human operator as the useful workspace. The secondary performance aspects such as dexterity, stiffness and technical feasibility have been influential in the choice of the set of parameters and their upper and lower limits. An efficient algorithm, based on the sign of the determinant of the Jacobian matrix, has been designed to compute the 5 DOF of the useful workspace for a given geometry. Due to the complexity of the objective function, genetic algorithms have been used to search for a optimum for the geometric parameters. The final results for the useful workspace were verified by a graphical representation of the dexterity and singularities of the whole workspace based on the discretisation of some of the DOF of the robot.

5.7 Discussion

As mentioned in the introduction, the optimisation of the geometric parameters of parallel robots is fundamental in order to take full advantage of the parallel structure. Due to the

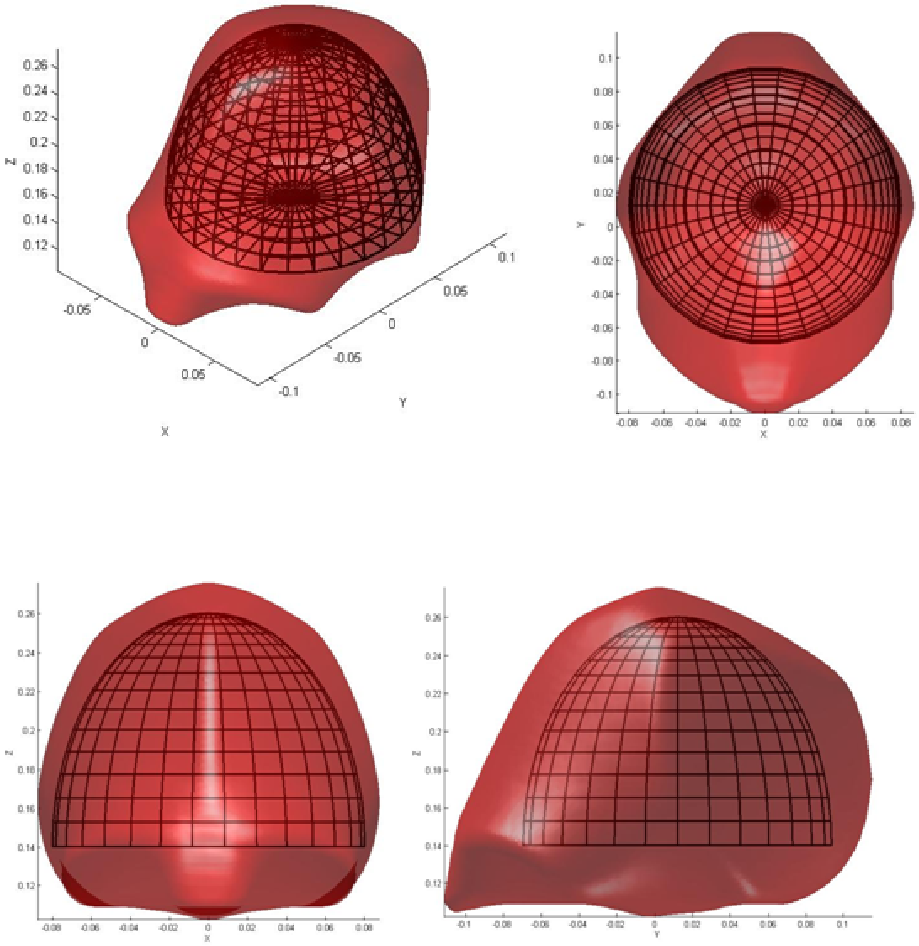


Figure 5.11: The half ellipsoid in comparison of the useful workspace

complexity of the structure, it is not intuitive to locate the singularities and this optimization process improved greatly the size of the useful workspace compared to initial designs. Most of initial designs based on trial-and-error had actually at least one singularity near the workspace centre, which was not acceptable. The optimisation process is presented in a linear way through this chapter but it is in practice a highly iterative process. It is therefore important to go back and forth from the optimization results to the range of parameters and optimization settings with several pilot optimizations. It should be noted that although singularities must be avoided, most of them are not at the boundaries of the whole workspace and are not considered dangerous for the human or the robot. In singular positions, the robot loses control of one of the DOF but can still exert forces in the other DOF. What is likely to happen in singular positions is that the controller will ask for an infinite force from the actuators in the lost DOF direction. Excessive forces on the actuator can be prevented by introducing saturation in the controller. However, the robot was designed to be controllable in 5 DOF and it is highly desirable to remove singularities as far as possible from the centre.

The singularities at the boundaries of the workspace are more critical. Those occur for example when the legs are fully extended or when the platform links are parallel or extended. If the user passes through these singularities, the robot may change assembly mode and the position and velocity kinematic model inside the controller will not be valid anymore. Clearly such situations must be avoided and force repulsion or an audio signal could be used to inform the user that he is near a boundary singularity. That is why a simple shape was chosen to represent the useful workspace so the human operator can recognize easily in which region he should stay. From our experience with actual users, this was well understood and no incidents occurred since the central useful workspace was large enough to let them perform their task in a comfortable way.

Chapter 6

Prototype Design and Implementation

6.1 Introduction

The PentaG robot is a high structural stiffness device based on a parallel architecture with configurable platform that offers 3T1R motions and grasping capabilities. It is therefore very suitable for any applications that required this type of motion in addition to grasping and a high structural stiffness. Two applications that come in mind are high speed pick-and-place and haptic teleoperation for micro-assembly.

Pick-and-place robots are designed for manufacturing, assembly, material handling and packaging applications. Speed and acceleration are the most important aspects to increase performance and commercial profitability of such devices. In order to reach a high number of cycles per minutes, pick-and-place robots need to have a high structural stiffness. The well-known Delta robot [10], shown in Figure 2.1 b), developed in the 80's and commercialized in the 90's, offered new solutions for the pick-and-place applications. This 4 DOF robot uses a vacuum grippers to handle the products. However, vacuum gripping is often not possible when the products present non-flat, porous or irregular surfaces as common in the food industry and mechanical grippers must then be mounted on the rigid platform, increasing the total inertia and degrading the robot performance. In the PentaG architecture, all the motors are located on the base and the device can be in principle lighter and faster than comparable robots with actuated mechanical gripper.

Haptic teleoperation devices are used to remotely control robotic tools and reproduce contact forces to the operator. The specific types of motion of the PentaG architecture make it highly suitable for teleoperation in micro-assembly. Via this master robot, the operator controls a slave robot to perform a $2\frac{1}{2}$D micro-assembly task and receives force-feedback information from the micro-force interactions between the slave robot and the assembled micro-parts. This can be useful, when automated micro-assembly is not possible such as in prototyping and low batch production scenarios. It is important to notice the impact of structural stiffness in the performance of the haptic device, especially when impedance control is used. One can conclude that the moving mass should be as low as possible in order to not disturb the commanded forces that have to be felt by the operator. However, the master robot needs a minimum of stiffness to be able to render high-frequency force-feedback. In addition, multi-point contact provides more information to the human operator and the operator can use his index and thumb to grasp and interact with the objects in a more natural way than with a single point contact. The configurable platform allows a design where all actuators are mounted on the base, which results in higher structural stiffness than if a grasping device was attached to a rigid platform.

For both applications, an optimisation that minimises the inertia-stiffness ratio of the robot structure is useful to improve the performance of the device since it increases the mechanical bandwidth of the system. The particular choices of actuators, sensors, amplifiers and transmission depend more on the chosen application since pick-and-place devices and haptic devices have very different requirements in term of velocity, forces, accuracy, friction, backlash, and backdrivability. In the following sections, the technical design of a haptic device based on the PentaG architecture is presented.

Section 6.2 of this chapter presents a novel and general method to minimise the total inertia of parallel robots while satisfying a minimum stiffness requirement. The results of this optimisation are then implemented in the design of the mechanical parts of a PentaG robot for haptic application. Section 6.3 of this chapter presents the design of the other components of the robot including the bearings, actuators, sensors, hardware and software controller based on analysis of the propagation of the specifications from the finger tips to the various components of the robot.

6.2 Inertia-Stiffness Optimisation

In an impedance control scenario, a haptic device records the movements of an operator and gives force feedback by mimicking the forces that would be present if the operator was interacting directly with a virtual or remote environment. For realistic force feedback, the device should be able to simulate high frequency forces while maintaining low inertia for free-air motion. Ideally, the inertia of the device should be zero.

In the PentaG architecture, the lengths of the parts have been already defined in Chapter 5 as they highly influence the kinematics of the robot. The kinematic parameters were chosen in a way that increase the compactness of the structure, therefore reducing the inertia. It was then verified that the force transfer functions was sufficient in the whole useful workspace, providing enough stiffness. The shape and size of the cross-section of the parts, as well as the material used haven't been defined so far since they have no influence on the kinematics of the robot. However they have a direct influence on the perceived inertia and stiffness of structure. Given a certain shape of the cross-section of a general robot link, increasing the cross-sectional area results in higher stiffness but also in a higher inertia.

Both the inertia and the stiffness that are perceived by the operator at the finger tips of a haptic device depend on the coupled influence of all mechanical parts of the robot. In addition, the perceived inertia and stiffness depend also on the position of the robot in the workspace. Since all those parameters are coupled, a global inertia-stiffness optimisation method must include all parts and positions simultaneously.

This section proposes a novel and general inertia-stiffness optimisation method that doesn't influence the kinematic parameters of the robot. Given the overall stiffness of the machine, the stiffness of each of the individual parts can be chosen such that the inertia of the total structure is as low as possible. Each part moves differently as a function of the motion of the finger tips. Parts that move relatively slower can be made stiffer and therefore heavier, for they have a relatively low contribution to the total perceived inertia.

For each mechanical link of the structure, an “Inertia Participation” and a “Compliance Contribution” are defined. The computation of the inertia participation required a rigid body model and the computation of the compliance contribution required a finite element model. Both are presented in Section 6.2.1. The computation of the inertia participation is presented in Section 6.2.2 and represents the average disturbance forces felt at the finger tips due to the inertia of the parts. Those forces depend on the position, the velocity and the accelerations of the finger tips. The computation of the compliance contribution is presented in Section 6.2.3 and represents the difference in stiffness perceived at the finger tips between an infinitely stiff part and the actual part, when the actuators are locked. The perceived stiffness depends on the load direction and the position of the robot. The optimization method and the results are then presented in Section 6.2.4 and 6.2.5 respectively.

6.2.1 Rigid Bodies Model and Finite Element Model

The analysis of the PentaG robot presented in Chapter 4 and the geometric optimization presented in Chapter 5 only deal with the kinematic parameters i.e. they were concerned with the motions of the robot without consideration of the forces that cause of these motions. In order to compute the disturbance forces at the finger tips created by the inertia of the various moving parts of the robot, a rigid body model that links forces and motions in all rigid parts must be formulated. Since this model is not intended to be used in real-time computation but only for the optimization process, a numerical model is sufficient for this purpose.

SimMechanics is a Matlab Simulink toolbox that allows easy implementation of multibody simulation. The inputs of the rigid bodies model are the kinematics connections, the definitions of the joints, and the masses and moments of inertia of the parts. A number of possible boundary conditions can be applied on the model, like gravity, tip forces and tip velocities/accelerations.

Due to the overconstraints in the mechanism, it is not possible to determine all the internal forces in the robot with a simple rigid bodies analysis. A second model is needed to compute the constrained forces. This model was made using the finite elements package Ansys. In this model each link is a beam with a certain cross section and stiffness. For each type of part a distinct element type, material model and real constant set is created. The material properties are modulus of elasticity, Poisson ratio and density. The final part of the FEM model consists of the constraints and the loads. Since the FEM model is controlled by a Matlab program, batch analysis over the workspace can be performed and results for each position are recorded in a Matlab file. The FEM model has been compared to the SimMechanics model in order to verify its validity. The two software programs work in very different ways so similar results for a large quantity of load cases increase the confidence in the model.

6.2.2 Inertia participation

6.2.2.1 Tip forces as function of velocity and acceleration

The inertia participation is defined as the average force the user experiences during free manipulation. These tip forces depend on both the velocity and acceleration the user imposes on the tips. For example, the acceleration of the actuator $\ddot{\mathbf{q}}$ depends on both both velocities $\dot{\boldsymbol{\chi}}$ and accelerations $\ddot{\boldsymbol{\chi}}$ at the tips.

$$\ddot{\boldsymbol{q}}\left(\boldsymbol{\chi},\dot{\boldsymbol{\chi}},\ddot{\boldsymbol{\chi}}\right)=\frac{d\left(\dot{\mathbf{q}}\left(\boldsymbol{\chi},\dot{\boldsymbol{\chi}}\right)\right)}{dt}=\frac{d\left(\mathbf{J}^{-1}\left(\boldsymbol{\chi}\right)\dot{\boldsymbol{\chi}}\right)}{dt}=\dot{\mathbf{J}}^{-1}\left(\boldsymbol{\chi},\dot{\boldsymbol{\chi}}\right)\dot{\boldsymbol{\chi}}+\mathbf{J}^{-1}\left(\boldsymbol{\chi}\right)\ddot{\boldsymbol{\chi}} \tag{6.1}$$

In general, the acceleration of each link of the robot depends on both the velocity and on the acceleration of the finger tips and these contributions can be calculated separately. Following simple tests with subjects moving a 3 DOF Novint Falcon haptic device in free-air, typical hand velocity and acceleration during haptic manipulations were evaluated at $v_m = 0.23{}^{\mathrm{m}}/_{\mathrm{s}}$ and $a_m = 1.5{}^{\mathrm{m}}/_{\mathrm{s}^2}$ respectively and those values are used for the velocity and acceleration cases.

6.2.2.2 Velocity and Acceleration Cases

The finger tips connected to the configurable platform of the PentaG robot can move in five independent directions. To determine the inertia participation, only velocities and accelerations that point into purely x, y or z direction were considered, the grasping movements in θ and ρ direction are neglected because during the high precision gripping operations (5 DOF) the velocities and accelerations stay relatively low, in contrast to the transport manoeuvres (3 DOF), where the movements are faster. This choice was made to reduce computation time, limiting the number of cases to six, i.e.:

$$\begin{array}{rcl}
\dot{\boldsymbol{\chi}}_1=[v_m 0000]^T & , & \ddot{\boldsymbol{\chi}}_1=\mathbf{0}\\
\dot{\boldsymbol{\chi}}_2=[0 v_m 000]^T & , & \ddot{\boldsymbol{\chi}}_2=\mathbf{0}\\
\dot{\boldsymbol{\chi}}_3=[00 v_m 00]^T & , & \ddot{\boldsymbol{\chi}}_3=\mathbf{0}\\
\dot{\boldsymbol{\chi}}_4=\mathbf{0} & , & \ddot{\boldsymbol{\chi}}_4=[a_m 0000]^T\\
\dot{\boldsymbol{\chi}}_5=\mathbf{0} & , & \ddot{\boldsymbol{\chi}}_5=[0 a_m 000]^T\\
\dot{\boldsymbol{\chi}}_6=\mathbf{0} & , & \ddot{\boldsymbol{\chi}}_6=[00 a_m 00]^T
\end{array} \tag{6.2}$$

By assigning a mass to the part of interest in the rigid body model and leaving the other masses to zero it is possible to calculate the tip disturbance forces as function of that mass, for each of the six cases. The tip forces resulting from the total structure is made up from the contributions of all parts. The inertia participation of a part type is the average of the tip forces of all load cases. Average forces are computed for both finger tips $\boldsymbol{F}_{\mathrm{tip1}}$ and $\boldsymbol{F}_{\mathrm{tip2}}$, and the inertia participation $\boldsymbol{F}_I$ is calculated as $\boldsymbol{F}_I = |\boldsymbol{F}_{\mathrm{tip1}}| + |\boldsymbol{F}_{\mathrm{tip2}}|$.

Since the inertia participation, as most properties of the robot, depend on the posture, the workspace has beeen sampled and for each posture the six load cases have to be applied to find the mean inertia participation. As each posture (x, y, z, θ, ρ) is symmetrical to $(-x, y, z, -\theta, \rho)$, it suffices to only investigate positive x coordinates when both positive and negative θcoordinates are examined.

For each part of the mechanism, a single number is then computed, the inertia participation, which represents the average forces of the different load cases over the sampling

Part type	Average force (per part of each part type) [N/kg]				
	Platform velocity		Platform acceleration		Total [N/kg]
	Due to mass of part	Due to rotational inertia of part	Due to mass of part	Due to rotational inertia of part	
n_{11},n_{51}	0.08	0.03	0.23	0.08	0.42
n_{21},n_{41}	0.10	0.04	0.27	0.09	0.5
n_{31}	0.09	0.03	0.25	0.09	0.46
n_{12},n_{52}	0.33	0.00	0.93	0.00	1.26
n_{22},n_{42}	0.41	0.00	1.08	0.00	1.49
n_{32}	0.35	0.00	1.02	0.00	1.37
n_{13},n_{14},n_{53},n_{54}	0.15	0.03	0.87	0.11	1.16
n_{23},n_{24},n_{43},n_{44}	0.19	0.04	0.95	0.13	1.31
n_{33}, n_{34}	0.17	0.04	0.90	0.11	1.22
n_{15},n_{55}	0	0.00	1.46	0.00	1.46
n_{25},n_{45}	0	0.00	1.63	0.00	1.63
n_{35}	0	0.00	1.46	0.00	1.46
n_{p1}	0	0	1.46	0	1.46
n_{p2},n_{p4}	0	0	1.63	0	1.63
n_{p3}	0	0	1.46	0	1.46
n_{p5},n_{p6}	0	0	1.51	0	1.51
n_{p7},n_{p8}	0	0	1.51	0	1.51

Table 6.1: Inertia Participation of each type of rigid links. In this notation,n_{ij} is the link j of leg i. Links with identical shape and symmetrical inertia participation are shown on the same line. Platform links n_{p1} to n_{p4} are directly connected to the robot legs and are constrained in rotation. Platform links n_{p5} to n_{p8} provide the mobility to the configurable platform. The final inertia participation is the average inertia perceived at the finger tips per unit of mass for all velocity and acceleration cases sampled over the useful workspace. Results shown that, in general, rotational inertia has a much lower contribution than mass inertia.

position of the workspace, per unit of mass of the part. Results are shown in Table 6.1. The notation used is the same as in the graph presented in Figure 3.16. Due to the symmetry of the workspace, leg 1 is symmetrical to leg 5 and leg 2 is symmetrical to leg 4. Parts with identical shape and symmetrical contribution are presented on a single line. 17 part types were identified and each of them represents a mass parameter to optimize.

6.2.3 Compliance Contribution

Each part of the structure has a certain stiffness in the different load directions. The stiffness of the total robot is build up from the stiffness of the different parts. When applying a force on a structure, the structure deflects some amount in the direction of the force and possibly also in another direction orthogonal to the force. When one of the parts of the structure is made infinitely stiff, the structure as a whole will be stiffer and the deflection in direction of the applied force will be smaller. The difference between the deflections of both cases can be considered as an estimation of the compliance contribution.

The FEM model is used to compute the stiffness of the structure for a given posture of the mechanism. The stiffness at the finger tips is computed when the motors are locked and the mechanism has zero mobility. The directions of the loads were not chosen in the end effector coordinates x, y, z, θ, ϱ but in finger tips coordinates x_{tip1}, y_{tip1}, z_{tip1}, x_{tip2}, y_{tip2}, z_{tip2}. For rigid body dynamics, five coordinates are sufficient to fully capture the position of both tips, but when taking the finite stiffness into account, six independent coordinates are needed, because position z_{tip1} can be different from z_{tip2}. For each of the six directions, the compliance is calculated by applying a unit force in the respective direction. When applying a force on a tip, the tip will displace over a distance δ in the direction in which the force is applied, but one might expect also a displacement of the tip in orthogonal direction to the applied force and a displacement of the other tip in some direction. Therefore, for each applied force, six displacements are found.

The stiffness of the structure greatly varies as function of the position in the workspace. For each of the sampled points, a number of static finite element analysis has to be done, which is time-consuming and this limits the number of samples that can be taken. The workspace was sampled using a set of 45 points in the x,y,z coordinates, each with 5 values for rotation θ and 5 values for grasping ρ, so in total 1125 postures.

The optimisation method minimises the inertia, while putting constraints on both the mean compliance and the 0.99-quantile of the compliance. It was chosen to eliminate the extreme 1% compliance computations so that the final results are not driven by a single point but is still valid for the vast majority of the workspace. Interesting is that the sum of the compliance contributions of all 17 part types, calculated individually for each type, comes close to the complete compliance that can be calculated with the whole FEM model. This is an indication that the above method is at least a good estimation of the compliance contribution.

6.2.4 Optimisation

The goal of the optimisation is to minimise the mean inertia participation, while keeping the compliance at a certain level. The parameters to optimise are included in a vector $\boldsymbol{m}$ which represents the mass of each of the part types. The inertia participation of these masses can be calculated using the inertia participation $\boldsymbol{\mathcal{I}}$ of Table 6.1:

$$f\left(\boldsymbol{m}\right) = \boldsymbol{\mathcal{I}}^T \boldsymbol{m}.$$

Changing the mass of a part also changes its stiffness. The length of the parts has been defined in the kinematic optimization and are fixed. The stiffness of parts can however be modified by changing the cross-section. For the optimisation it is presupposed that the shape of the cross-section stays the same as the initial geometry, for which the compliance participation was calculated. It was chosen to use a tubular cross-section for all parts except for parts n_{p1}, n_{p2}, n_{p3} and n_{p4} which make the connection between the legs and the configurable platform. These parts hold all the bearings of the platform and leg attach points and have a rectangular cross-section. All cross-sections can be scaled with a scale factor s. The mass of the part scales with a factor s^2, so $s = \sqrt{\frac{m}{m_0}}$. The parts have different stiffness for the different load types. Bending, torsion and compression of the links can occur and normally bending stiffness and torsion stiffness are lower than the stiffness under compression. Therefore, it can be assumed that deflections are most likely to be a result of the limited bending and torsion stiffness.The bending and torsion stiffness both scale as s^4 and therefore it is assumed that when scaling the cross-section of a part, the stiffness scales with s^4 and the compliance with s^{-4}. So when the mass of a part is changed, the compliance contribution changes approximately as follows:

$$\frac{C}{C_0} = \left(\frac{m_0}{m}\right)^2 = s^{-4}$$

By adding up for each sampled posture and load case the compliance contribution, the total compliance after changing the mass of the parts can be approximated.

The part mass is optimised, such that the inertia participation is as low as possible, while meeting the two conditions on the compliance i.e. the mean compliance $\overline{C}$ and the 0.99-quantile of the compliance $C_{0.99}$. With n the number of sampled workspace points, p the sample number and t the part type number, the optimisation is implemented in Matlab using the function fgoalattain, using a goal of $\boldsymbol{m} = 0$.

$$\min_{\boldsymbol{m}} f(\boldsymbol{m}) \text{ such that } \begin{cases} \frac{1}{n}\sum_{p=1}^{n}\sum_{t=1}^{17}\frac{C_{p,t}}{C_{p,t,0}}\left(\frac{m_{t,0}}{m_t}\right)^2 < \overline{C} \\ \left[\sum_{t=1}^{17}\frac{C_{1,t}}{C_{1,t,0}}\left(\frac{m_{t,0}}{m_t}\right)^2 \cdots \sum_{t=1}^{17}\frac{C_{1,t}}{C_{1,t,0}}\left(\frac{m_{t,0}}{m_t}\right)^2\right]_{0.99} < C_{0.99} \\ \boldsymbol{m} > 0 \end{cases} \tag{6.3}$$

The shape of the cross-section has to be defined manually before they are scaled by the optimisation procedure. The final goal of the method is however not to find some geometry with theoretical identical stiffness, but to design parts that can be manufactured and can be integrated in the rest of the structure. The design adds a lot of practical constraints to the geometries that can be chosen. Therefore, the initial choice of cross-section is based on the typical loads a part is expected to deal with. During the detailed design, the cross-sections can be changed, based on considerations like manufacturability.

6.2.5 Results

For the compliance constraints a mean compliance of $\bar{C} = 1 \cdot 10^{-4}$ m and 0.99-quantile of the compliance $C_{0.99} = 3 \cdot 10^{-4}$ m were chosen. The mean compliance value corresponds to accuracy of kinaesthetic position feeling in the human, which is about 100 µm. More details on specifications derived from human kinaesthetic properties are presented in Section 6.3.1 of this chapter. Table 6.2 lists the required radii, bending stiffness and resulting mass and inertia participation of the different part types. Parts n_{p1}, n_{p2}, n_{p3} and n_{p4} have a square cross section instead of a tube cross-section. Their rotational inertia is not computed since those parts are not rotating. The parts that contribute the most to the inertia participation are the lower limbs and the upper limbs, for almost 60%. The total inertia participation is 0.22 N, which means that the user will feel average tip forces at his hand of 0.22 N. Combining this with the earlier found mean acceleration $\bar{a} = 1.5$ $^{\mathrm{m}}/_{\mathrm{s}^2}$, this is comparable with a mass of 150 g. These values are theoretical/desired since the mass of bearings, axles and extra features of parts is not yet included. A new resulting inertia participation, calculated for the detailed design is presented in Section 6.4.

The resulting compliances of the structure in different postures are plot in Figure 6.1. Clearly, the 0.99-quantile constraint is limiting. The resulting average stiffness of the structure is $2 \cdot 10^4$ $^{\mathrm{N}}/_{\mathrm{m}}$.

Closely related to the stiffness of the parts are the eigenfrequencies of the structure. The eigenfrequencies were not part of the inertia-stiffness optimisation of the structure

Parts	For single part of each part type						Number of parts	Total per part type	
	Length l [mm]	Outer radius r_o [mm]	Inner radius r_i [mm]	Bending stiffness I_{yy}/I_{zz} [10^5 m^4]	Inertia I_y/I_z [10^5 kgm^2]	Mass m [kg]		Mass m [kg]	Inertia participation [N]
n_{11},n_{51}	90	6.5	4.4	3.09	2.20	0.032	2	0.064	0.027
n_{21},n_{41}	90	6.2	4.2	2.61	2.02	0.029	2	0.059	0.029
n_{31}	90	5.4	3.7	1.48	1.51	0.022	1	0.022	0.010
n_{12},n_{52}	44	3.3	0.0	2.28	0.07	0.004	2	0.008	0.010
n_{22},n_{42}	44	2.7	0.0	1.00	0.04	0.003	2	0.005	0.008
n_{32}	46	3.5	0.0	2.59	0.09	0.005	1	0.005	0.007
n_{13},n_{14}, n_{53},n_{54}	200	3.6	2.6	0.03	2.20	0.007	4	0.026	0.031
n_{23},n_{24}, n_{43},n_{44}	200	2.7	1.9	0.01	1.19	0.004	4	0.014	0.019
n_{33},n_{34}	200	4.0	2.8	0.04	2.67	0.008	2	0.016	0.020
n_{15},n_{55}	44	3.2	0.0	2.06	0.06	0.004	2	0.008	0.011
n_{25},n_{45}	44	2.7	0.0	1.01	0.04	0.003	2	0.005	0.009
n_{35}	46	3.4	0.0	2.21	0.08	0.004	1	0.004	0.007
n_{p1}	46	height= 7.9	width= 6.9	0.77	-	0.007	1	0.007	0.010
n_{p2},n_{p4}	11.1	height= 4.9	width= 5.4	1.21	-	0.001	2	0.002	0.003
n_{p3}	42	height= 6	width= 5	0.69	-	0.003	1	0.003	0.005
n_{p5},n_{p6}	48	4.1	3.2	2.68	0.05	0.003	2	0.005	0.008
n_{p7},n_{p8}	30	3.2	2.3	4.50	0.01	0.001	2	0.002	0.004
total	-	-	-	-	-	-	33	0.26	0.22

Table 6.2: Results of the inertia-stiffness optimisation. Same notation is used as in Table 6.1.

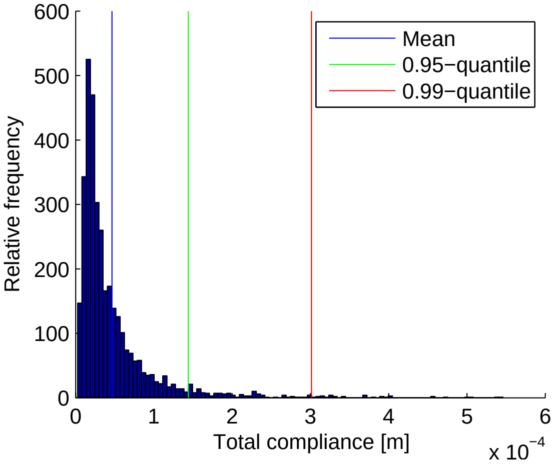


Figure 6.1: Histograms of the compliances for a sample of different postures in the workspace. For each posture, compliances in six directions are taken into account.

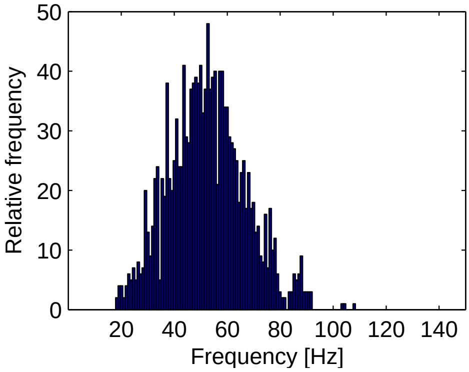


Figure 6.2: Histograms showing the first eigenfrequencies for workspace sample. Clearly, the eigenfrequencies are posture dependent.

but they are important for the bandwidth of the system. The frequency content of the movements the user can make is low, typically up to 10 Hz [6]. The desired actuation bandwidth is higher, 30 Hz, which has to do with the fact that this is about the maximum frequency that can be felt kinaesthetically.

For the resulting geometries and stiffness, the eigenfrequencies were analysed, using the finite element model in Ansys. This was done with a mesh of ten elements per link. As with almost all properties of the device, also the eigenfrequency depends on the robot position, For a number of postures in the useful workspace, the eigenfrequencies were calculated. Figure 6.2 shows histograms of the frequencies of the first eigenmode. The first eigenfrequencies ranges ranges between 20 and 100 Hz. Due to the non-linear and not fully predictable stiffness of the bearings, the actual eigenfrequencies may be different, but it is for certain that the first eigenfrequency can be below 30 Hz. This means that the feedback to the hand can be distorted.

6.3 Technical Design

6.3.1 Specifications

Component specifications of haptic devices, such as motors, sensors and bearings, may sometime be tedious to define since ultimately, the device should allow the user to perform his task efficiently in a comfortable way. There are different ways to define task performance and ergonomic. Task performance can be judged based on criteria sush as completion time, accuracy, average force exerted ect.. or any arbitrary combination of those criteria. Ergonomics are dependent on the individual and are difficult to measure objectively. In addition, the relation between the specification at the finger tips and the specification of the rest of the system is dependent on the position of the robot. The derivation of the specifications involves the computation of a high number of sampling points within the workspace and is based on average results and worst-case scenarios results.

Nevertheless, specifications are important in the detailed design of any robotic system as they serve as guideline for the many decisions and trade-off involved in a design process. A

Specification	Quantity
Maximum force to the user finger	7 N
Largest disturbance force	0.3 N
Force feedback bandwidth	30 Hz
Position resolution	0.1 mm
Position precision	1 mm

Table 6.3: List of Specifications at the finger tips interface for the PentaG haptic device

complete teleoperation system has two types of interface with the outside world; the task environment and the human operator. The haptic device is physically disconnected from the task environment and interacts virtually with it via a bilateral control. For stability reasons, this bilateral control cannot perfectly reproduce the environment interactions to the haptic device and some transparency of the environment is already lost between the user and environment. The bilateral controller can also be used to enhance the real forces interaction by amplification, filtering, or augmented force feedback guidance to help the operator to complete his task. Since the haptic device is not physically connected to the task, few design specifications will come directly from the task requirements.

On the other hand, the haptic device is physically connected to the human operator interface. In an impedance control sheme, the role of the haptic device is to record human operator motions and provide on the human hand force feedback information. If a haptic device is perfectly adapted to the human motion and perception limits, it makes no sense to require more from the haptic device, since the human operator will not be able to actuate or sense this. The specifications are therefore derived from the human actuation and sensing capabilities.

Table 6.3 shows the list of specifications that have been used for the implementation of the PentaG haptic device. Comfortable forces have to be at most 15% of the maximum force the human can exert [70]. Assuming maximum grasping forces of the fingers in the order of 30-50 N, operators can comfortably exert forces of 5-7 N. The resolution of finger forces is circa 0.3 N [74] and this value is used as the maximum allowed disturbance force between the finger tips forces and the command input force. The human actuation bandwidth is around 10 Hz while the bandwidth of perception of kinaesthetic movement is around 30 Hz [31]. Since a haptic device is used both for actuation and perception, the highest of those two values is used as specification for the system. The accuracy of kinaesthetic position feeling in the human is about 100 µm for frequencies up to 30 Hz. Therefore the resolution of position measurement should be 100 µm. Absolute positioning should be accurate up to the human free air positioning accuracy of 1 mm. The specifications at the finger tips are summarized in Table 6.3.

6.3.1.1 Specification Transfer

The mechanical transfer between the actuators and the finger tips is highly dependent on the position of the haptic device. Because of this, it is not possible to derive single specification numbers for the actuators and sensors based on a unique position. In Chapter 5, a useful workspace was defined within which the human operator is intended to use the robot. In order to obtain single numerical values for the actuator specifications, a distribution of the propagation of the finger tips specifications to the actuator specifications was done for a 5 dimensional grid that cover the entire useful workspace of the robot. For each robot position, a set of load cases or displacement cases is defined that cover the 5 dimensions of the finger tips in this specific position. From this distribution, actuator specifications are chosen to respect the specifications at the finger tips for 95% of the distribution. A statistical approach is needed here due to the extreme large range of different transfer functions between the finger tips and the actuator over the workspace and all cases. Design requires compromises and trade-off between the different requirements and a design that would satisfy one of the specifications for 100% of the workspace will be well over-designed for the vast majority of the workspace and makes the satisfaction of the other requirements impossible to achieve. Figure 6.3 presents the distribution of the propagation of the specifications from the finger tips to the actuator-sensor system for each specification listed in Table 6.3. Also presented are the values corresponding to the limit of the worst 10%, 5% and 1% of the cases.

The motion range of the actuator and sensor are entirely defined by the inverse kinematics of Section 4.1.1. A probability range should not be used here because the entire workspace should be reachable. The motion range of each actuator relatively to the horizontal plane is -42° until 85° for the useful workspace.

In order to determine the sensor resolution, the smallest motion-step that a human is able to detect must also be seen by the sensor system. To determine the entire space in which the fingertips are able to move without a change in the sensor readout, the entire space spanned by all combinations of positive and negative actuator angle deviations are transferred from the actuator angle space to the fingertip space using the inverse kinematic relations. Note that this assumes small angular deviations. This method of mapping the corners of a space between coordinate systems can be found in [51]. The largest deviation in fingertip space determines the scale-factor needed to define the smallest detectable sensor step. Using a maximum fingertip displacement of 0.1 mm the sensor resolution should be $6.4 \cdot 10^{-5}$ rad for the 95% interval.

Based on the maximum force of 7 N a user should be able to exert to the device, 34 load-cases for each grid position on the fingertips of the device are defined and the resulting actuator torques are computed. This results for the 95% interval in an actuator torque of 2.3 Nm. Since static disturbance forces may be present at all actuators and sensor at the same time, the worst case is determined by evaluating all combinations of positive and negative unit disturbance torques at the actuator side. The factor needed to scale down the worst case resulting fingertip force to 0.2 N determines the allowed disturbance. For the 95% interval this is $4.8 \cdot 10^{-3}$ Nm. The force on the user due to viscous damping in the actuator equals $\boldsymbol{F}_{friction} = c_{actuator}\mathbf{J}^{-T}\mathbf{J}^{-1}\dot{\boldsymbol{\chi}}$. To determine the end-effector velocities that are applied, it is assumed that disturbing actuator forces due to grasping and end-effector rotation velocities ($\dot{\rho}$ and $\dot{\theta}$) are negligible compared to the actuator motion due to end-effector motion ($\dot{x}$, $\dot{y}$ and $\dot{z}$). Therefore the typical user speed of $v_m = 0.19\,{}^{\mathrm{m}}\!/\!{}_{\mathrm{s}}$ is applied in equally distributed directions in Cartesian space using a unity friction coefficient. The scaling factor between the magnitude of the fingertip forces and the maximum allowed budget of 0.1 N then determines the allowed friction coefficient. This is $1.44 \cdot 10^{-3}\,\frac{\mathrm{Nm}}{rad/\mathrm{s}}$ for the 95% interval.

In a way comparable to the dynamic friction, an equation for the force due to inertia of the actuator can be derived. There is however an extra term due to the dependency of the Jacobian on χ: $\boldsymbol{F} = \mathbf{I}_{actuator}\left(\mathbf{J}^{-T}\mathbf{J}^{-1}\ddot{\boldsymbol{\chi}} + \mathbf{J}^{-T}\dot{\mathbf{J}}^{-1}\dot{\boldsymbol{\chi}}\right)$. can be derived: $\boldsymbol{F} = I_{actuator}\mathbf{J}^{-T}\mathbf{J}^{-1}\ddot{\boldsymbol{\chi}}$. This can be evaluated in exactly the same was as the dynamic friction, only using a maximum acceleration of $a_m = 1.5\,{}^{\mathrm{m}}\!/\!{}_{\mathrm{s}^2}$. The results for the 95% interval is $1.87 \cdot 10^{-4}\,\mathrm{kgm}^2$.

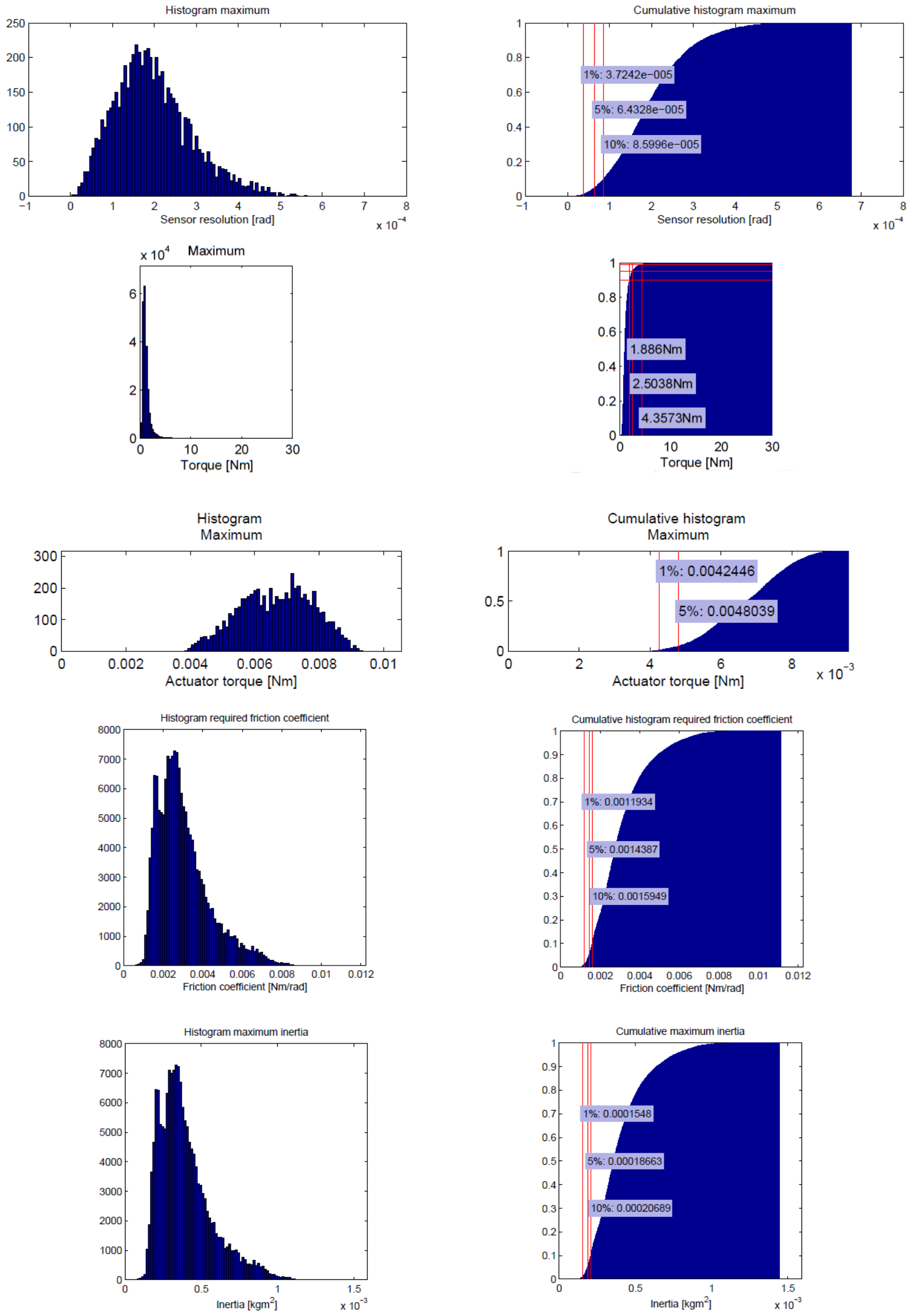


Figure 6.3: Distribution of the finger tip specifications over the useful workspace and the load cases to the actuator-sensor system

6.3.2 Actuators

The required function of the actuation system is to supply a well defined torque to the lower limb of the structure, such that the desired force is exerted to the operator. When observing commercially available haptic interfaces based on parallel mechanisms, such as for example the Omega from Force Dimension, the actuation system contains mostly a conventional DC-motor and a transmission system. However, any transmission system will magnify the disturbance forces produced by the motor inertia and friction. It can be concluded that to achieve minimal disturbance forces, a small transmission ratio must be used. In the extreme case this would mean application of a very large motor and a very small transmission ratio. This is however not possible due to size and cost limitations on the motor. Also every type of transmission will introduce unwanted forces, especially when large forces have to be passed through. Therefore using a motor with a large torque output that does not require a transmission and fits within the specified dimensions would be recommended. By using a brush-less type of motor the allowed output torques can be higher and the parasitic forces lower than with a brushed type motor. By using a direct drive approach for the motors, the disturbance forces created by the motor inertia and friction are small enough to be orders of magnitude lower than the disturbance forces created by the optimized inertia of the robot structure defined in Section 6.2. From a number of available motor manufacturers a DC brush-less motor type has been selected. For the specified maximum torque the stall torque can be used because this torque will only occur on a limited number of locations when the maximum force is requested. The Faulhaber 4490 048BS met the specifications in terms of torque.

6.3.3 Sensors

The sensors measure the angle between the lower limb of each leg and the base-plate. The final concept is the optical encoder, combined with a small transmission. The transmission is introduced to limit the needed encoder resolution so that the budget spent on encoders remains in balance with the cost of the rest of the prototype. The sensor transmission will be of the cable type. This will provide a play-free transmission that will prevent slip. Because the inertia and dynamic friction of encoders is much more limited, the disturbance forces created by the encoder transmission is expected to be less critical.

Using the direct drive motor concept and the rotary encoder with wire transmission, the detailed design of the actuation and sensing system can be made. Orientation on available rotary encoders that remain within the specifications, led to the conclusion that the upper limit for the encoder resolution is around 5000 pulses per revolution. To minimise the required transmission, the quad-counts of the encoder will be used, resulting in an effective resolution of $\frac{2\pi}{5000 \cdot 4} = 3.1 \cdot 10^{-4}$ rad. Dividing by the required resolution gives an encoder transmission of $\frac{3.1 \cdot 10^{-4}}{6.4 \cdot 10^{-5}} = 4.9$. Using a small margin the transmission ratio is defined at 5. The barrel at the encoder around which the wire is wound has a spiralled groove with a radius larger than the wire to provide smooth winding of the wire.

The selected Scancon 2MCA encoder has a 5000 pulses or 10000 steps per revolution resolution. Because of the transmission that has been used, the encoder should have effectively 20000 positions on a single revolution. The effective inertia of this encoder at the lower limb, including the effect of the transmission, is small compared to the rest of the system. Figure 6.4 shows a picture of a complete actuation-sensor system for a single leg.

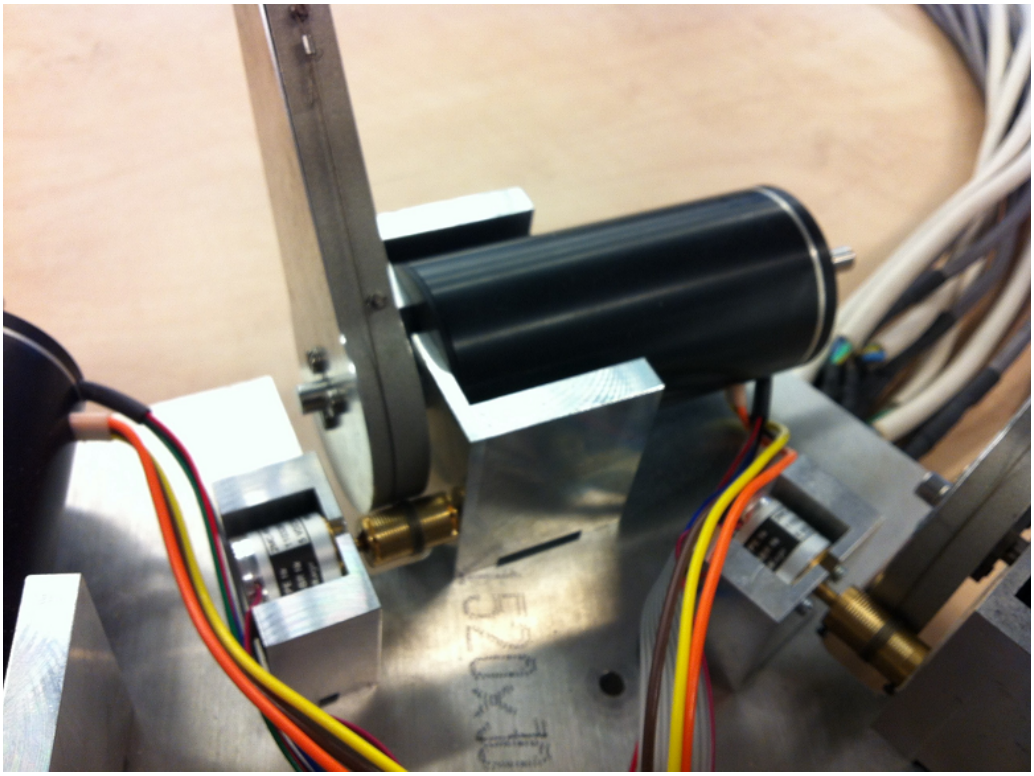


Figure 6.4: Implementation of the actuator-sensor system. The lower part of the robot leg is fixed to the motor shaft with screws and a motor flange, so that the leg is still detachable from the motor is case of maintenance or reparation of the robot. Also visible are the motor holder, the encoder holder, the wire bus, and the base plate.

6.3.4 Bearings

Two possibilities for the revolute joints are slider bearings and rolling bearings. Rolling bearings consists of two rings, with rolling elements in between. Slider bearings consist of a bush with a thin inner layer of a material with a low friction coefficient like Teflon or polyacetal, such that a metal axle can run with relatively low friction. Rolling bearings will normally show less friction then slider bearings. There are all kinds of roller bearings, but an easy to use and often applied one is the deep groove ball bearing. A slider bearing does not constrain in axial direction, so additional bearings are needed for handling axial force. Combined with the fact that slider bearings have higher friction, deep groove ball bearings were chosen.

All bearings have some play. Play is unwanted in the mechanism, as it causes losses in the mechanism during oscillations and consequently results in less transparent force feedback. Play was eliminated by pre-loading the bearings, such that all bearings are always pushed to one side and always keep contact. The pre-load force has to be higher than the expected maximum load of the bearing. Still a deep groove ball bearing has a problem with play and its inability to take radial moments. A solution to both problems is to use a pair of bearings at each hinge. Radial moments at the hinge will be resolved into two force components that work in radial force, exactly the type of load a deep groove bearing is designed for.

There are different possibilities for the mounting of the bearings and to constraint them axially; by using circlips inside the hole or other stops to be put in afterwards or by gluing. The first alternative makes the parts more laborious to fabricate, as the part needs extra features like internal grooves. The pre-load should be applied either at the inner ring or the outer ring, but in both cases a spring and some suspension of the spring is needed. Also using an axle with thread was considered, to remove the play by screwing a nut instead of a spring, but in that case the fit between axle and bearing is bad and the axial loading inexact and possibly high. Gluing however does not have these disadvantages. During the curing of the glue either inner or outer rings are lightly pressed towards each other, such that the play is removed. The axial force applied during gluing can be precisely controlled. The hole in neither the part nor the axle need extra features. Because of these advantages, gluing was chosen. The main disadvantage of gluing the bearing is that disassembly is not so easy. In case the robot needs some maintenance or reparation, the glue has to be heated such that it degenerates in order to take a joint apart.

6.3.5 Mechanical Parts Design

6.3.5.1 Lower limb

Figure 6.5 shows the final design of the lower limb assembly. The lower limb is a complex part as it is has to connect the mechanism to the motors and sensors. The PentaG has no transmission between lower limb and motor, but the sensor needs a transmission of approximately four times by means of a cable transmission. In the figure, the front plate is removed to show the inside of the part, but normally the profile is closed by a plate of the same shape. The part is directly connected to the motor axle through the big hole at the left. At the right, a tube is inserted, in which the bearings are placed, such that the bearings can be placed at the desired mutual distance.

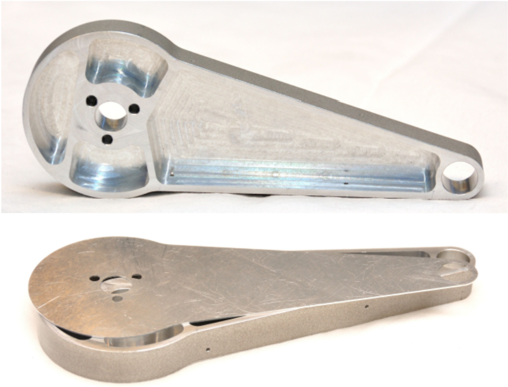


Figure 6.5: Lower limb pictures. In the upper picture the inner structure is clearly visible. In the lower picture also the cover plate is shown. The cover plate and the lower limb are glued together.

The inertia-stiffness optimisation was done for a rigid beam with tubular cross-section for the lower limbs. However, it was chosen instead to use a variable cross-section with a cover plate as shown in Figure 6.5. The required bending stiffness of the lower limbs is circa $k_{\mathrm{F}} = 3.6 \cdot 10^5\ {}^{\mathrm{N}}\!/_{\mathrm{m}}$. The final lower limb was tested using a FEM and has a stiffness of $k_{\mathrm{F}y} = 4.0 \cdot 10^5\ {}^{\mathrm{N}}\!/_{\mathrm{m}}$. The actual stiffness is higher than the calculated required stiffness, with a lower mass than a tubular cross-section. The thickness of the left out flat surface at the side is 1 mm, the thickness of the walls 2 mm. Making them smaller increases the chance of damage during milling.

6.3.5.2 Parallelogram units

The PentaG robot has five parallelogram units. Figure 6.6 a) shows a drawing of such a parallelogram. It consists of two equal horizontal rods n_{i2} and n_{i5} and two equal upper limbs n_{i3} and n_{i4}.

The horizontal rods are made out of one part. It consists of a single axle with a diameter of 5 mm, with two small ridges for the bearings to position the lower limb and platform bearings against. The far ends are milled on both sides, to make a flat contact with the small rings horizontal rod and the bearings of the upper limb fork. The upper limbs are for the greater part made out of carbon composite material. This material consists from stiff carbon fibres hold together by epoxy resin. The material is light and has a high stiffness to weight ratio. At the hinges, bearings or pins should be connected to the part, which is hard to realise in the carbon composite. Therefore, inserts are needed that can be connected to the carbon profile. These inserts can be made from aluminium and a drawing is shown of one of them is shown in Figure 6.6 b).

6.3.5.3 Configurable Platform

In the configurable platform, platform link n_{P1} is connected to two legs and platform links n_{P2}, n_{P3} and n_{P4} are each connected to a single leg. Links n_{P5}, n_{P6}, n_{P7} and n_{P8} connected the previous links together to close the configurable platform. In the useful

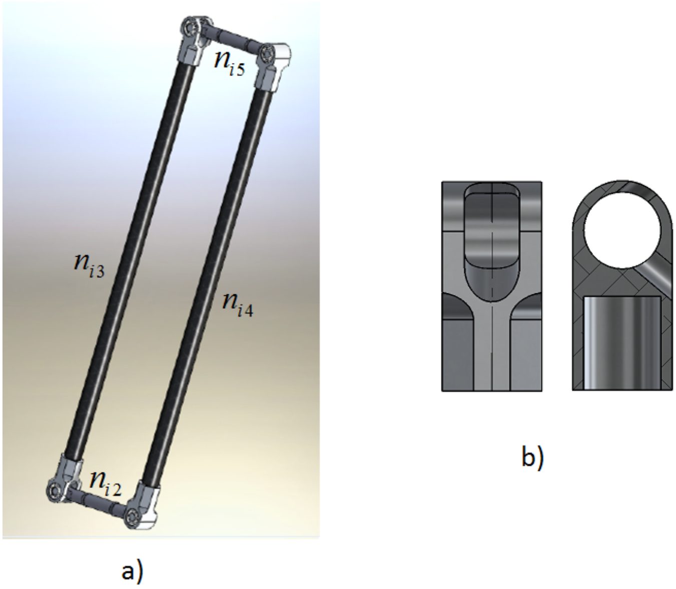


Figure 6.6: Drawing of the Parallelogram Units

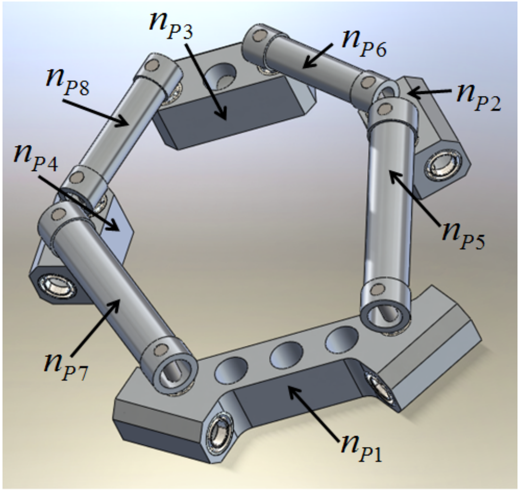


Figure 6.7: Drawing of the platform.

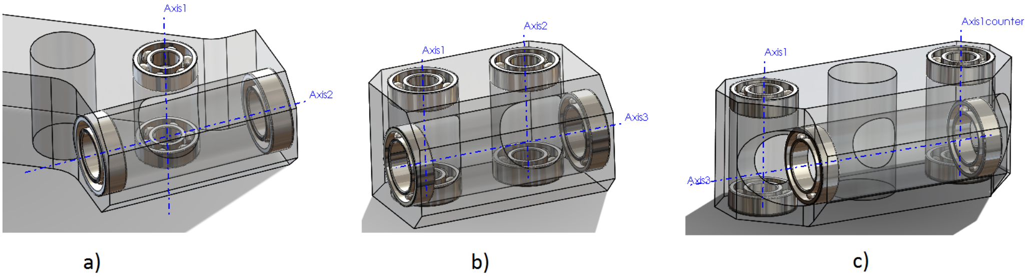


Figure 6.8: Drawings of a) the n_{P1} link, b) the n_{P2} and n_{P3} links, and c) the the n_{P3} link with bearings.

workspace defined in Section 5.5, the joints of the configurable platform must covert a range of about 50 degrees. In some positions, the links of the configurable platform cross the upper horizontal rods of the limbs. The consequence is that the n_{P5}, n_{P6}, n_{P7} and n_{P8} parts have to lie higher than the upper horizontal rods in order to avoid collision.

The n_{P1} link is drawn as part of the platform in Figure 6.7. This part is a complex, holding four bearing pairs. To make the part lighter three extra holes are bored. These holes could be used to attach the finger tips and possibly force sensors. To make the part as small as possible, the vertical holes of the platform joint and the horizontal hole of the leg joint intersect each other, which is clearly visible in the semi-transparent drawing of Figure 6.8 a). Of course the axles do not intersect.

The n_{P2} and n_{P4} parts (see Figure 6.8 b)) are comparable to the A1-part, but are connected to a single leg. Therefore, the part can be relatively small. The wall thickness is however nowhere smaller than 1 mm, as parts that have to be milled could otherwise break due to the high machining forces.

The n_{P3} part is in fact comparable to the n_{P2} and n_{P4} parts, but the connections are further apart and the part is consequently bigger. As the part is indeed longer, a problem of possible collision between the bottom of the n_{P3} part and the connected upper limbs appears. The available space is tight and therefore on both sides a vertical chamfer is applied, see Figure 6.8 c). This chamfer also prevents the part from collision with the upper limbs.

The n_{P5}, n_{P6}, n_{P7} and n_{P8} parts hold the pins that connect them to the n_{P1}, n_{P2}, n_{P3} and n_{P4} and therefore don't have bearings. The base material is standard aluminium 6060 tube with a wall thickness 1.5 mm. It is milled to get a small wall thickness of 1 mm, while the far ends are left thicker, for more strength at the contact points with the axles. Two types are shown in Figure 6.9.

6.3.5.4 Comparison between the desired and the final geometry

The optimisation of Section 6.2 resulted in an ideal geometry with a certain inertia participation. During the detailed design, sometimes it was necessary to deviate from the ideal geometry. This concern especially the n_{p1}, n_{p2}, n_{p3} and n_{p4} links which are small and are holding 6 to 8 bearings each. Now the geometry of all parts is determined, it is interesting

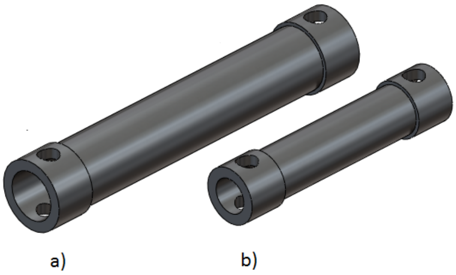


Figure 6.9: a) n_{P5} and n_{P7} parts b) n_{P6} and n_{P8} parts

to evaluate the final properties of the structure. The inertia participation of the final geometry was computed using the same method as in Section 6.2. Meant by additional parts are the parts that were not taken into consideration in the inertia-stiffness optimisation, namely the bearings and the axles. The total inertia participation of the final geometry is 0.31 N, whereas that of the geometry used in the inertia-stiffness optimisation is 0.22 N. The inertia participation is thus increased by 33% during the detailed design.

6.3.6 Hardware Control

A flexible and easily programmable controller system is required to experiment with different controller structures. Therefore the system runs on standard PC hardware using a dedicated controller card. The real-time operating system xPC Target runs the controller and maintains the connection to the Matlab environment. Different considered solutions for the interface card are: separate analogue, digital and encoder cards and all-in-one cards. Using multiple single purpose cards appeared to be a relatively expensive solution and would require a large PC system as well. The performance of the selected Quanser Q8 all-in-one card appeared to be sufficient and the costs lower than when using separate cards. The PC that will be running the system should have sufficient space for the interface card, an xPC Target supported network card and a recent processor. The TU Delft standard system: Dell Optiplex 780 is chosen.

6.3.7 Software Control

Figure 6.10 shows a schematic representation of the robot control structure. The control implemented is a haptic impedance control where the position of the robot is recorded and sent to a slave robot or virtual environment and the iteration forces are fed back to the robot. The input part reads the encoder and light barrier signals and computes the absolute angles $\mathbf{q}$ of the lower limbs.

As mentioned in the Section 4.1.2, it was not possible to obtain an analytical solution to the direct position kinematic (DPK) problem and therefore a iterative procedure that was fast enough to be used in real-time was developed. Using this procedure, the end-effector position χ is computed and sent to a virtual environment or to a remote slave device. The end-effector position is also used to compute the Jacobian matrix J for the current position. This matrix is used to convert the force feedback signals received from the slave side into desired motor torques. The forces due to gravity do disturbs the feedback forces

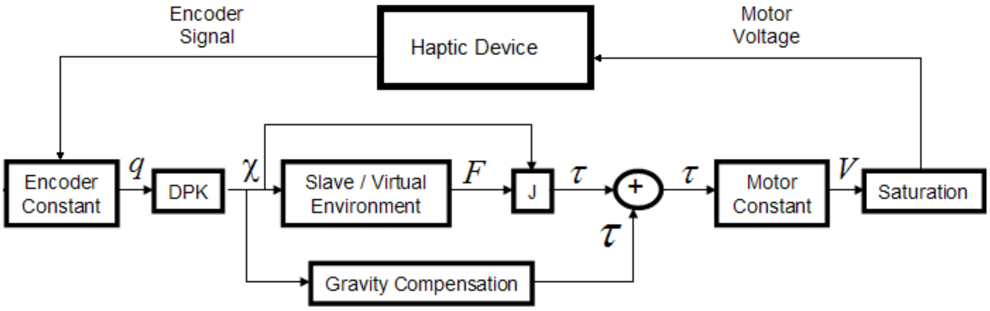


Figure 6.10: Schematic Overview of the Software Structure

Table 6.4: Mechanical system properties of the actuator-sensor system. Of all properties except the maximum force the extreme values excluding the 5% worst cases are given. For the maximum force the 10% worst cases are omitted.

Property	Value at		
	Lower limb	Fingertip (workspace extreme)	Fingertip (workspace average)
Inertia	$1.7 \cdot 10^{-5}\,\mathrm{kgm}^2$	$9.1 \cdot 10^{-3}\,\mathrm{N}$	$5.1 \cdot 10^{-3}\,\mathrm{N}$
Dynamic friction	$9.55 \cdot 10^{-6}\,\mathrm{Nm}/\mathrm{rad/s}$	$6.6 \cdot 10^{-4}\,\mathrm{N}$	$3.7 \cdot 10^{-4}\,\mathrm{N}$
Static friction	$4.65 \cdot 10^{-2}$ Nm	0.2 N	0.14 N
Measurement resolution	$6.28 \cdot 10^{-5}\,\mathrm{rad}$	0.1 mm	$2.7 \cdot 10^{-2}\,\mathrm{mm}$
Maximum force	1.7 Nm	6.3 N	10 N

and should be compensated as much as possible. For this purpose, a Jacobian matrix was developed to compute the transmission of the gravity force from the centre of mass of each link into the actuator torque. All requested torques are added and the corresponding output voltage in computed. Maximum absolute values for the output voltage are defined to prevent that high torques that could damage the device or harm the operator.

6.3.8 Properties of the actuator-sensor systems

After evaluating the design choices an overview of the performance of the planned system is made and given in Table 6.4. Note that the mentioned static friction is relatively high, but is must be noted that this is a worst case value based on the maximum expected encoder friction, which is the largest contribution to this value. Additionally the structure introduces on average 0.31 N at the fingertips of disturbing forces and an average deformation of 0.35 mm for the velocity, accelerations and load cases over the useful workspace.

6.4 Implementation and Testing

The aluminium parts and the carbon tubes of the configurable platform and the legs were assembled together by fixing the various deep-groove bearings in place. The final structure

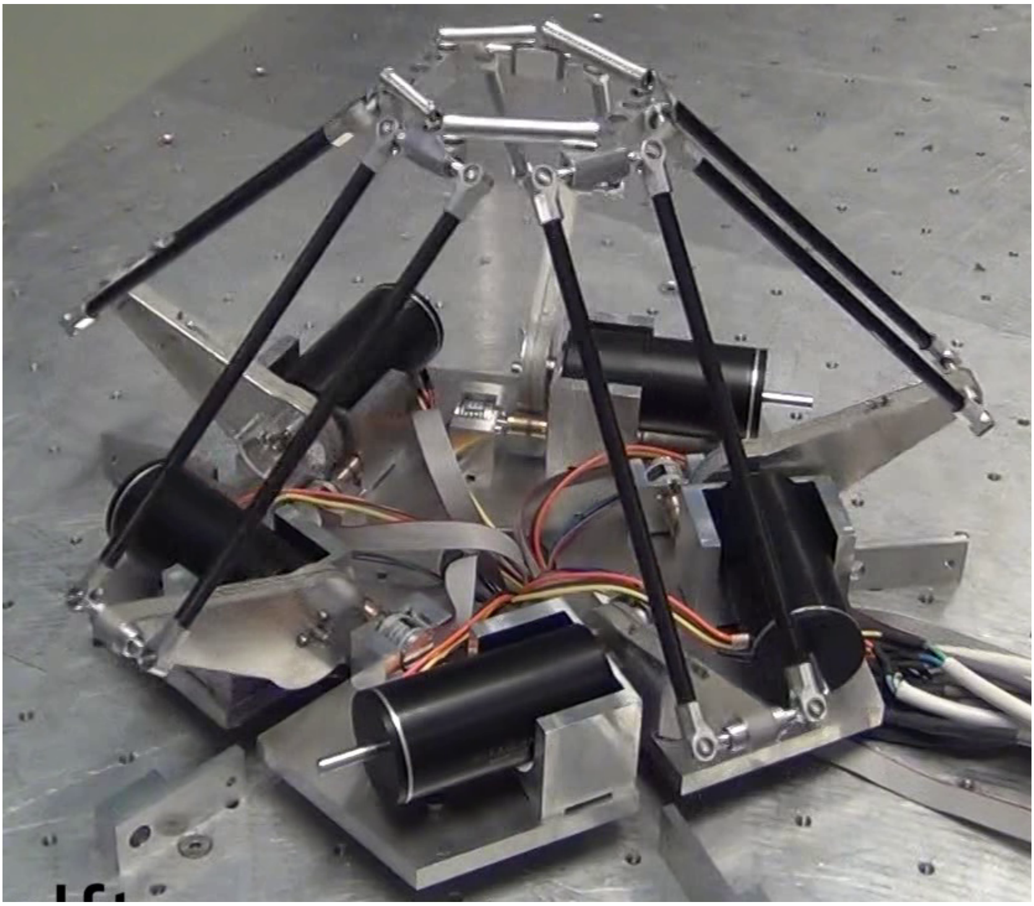


Figure 6.11: Final Assembly of the PentaG robot

was mounted on the 5 motors located on the base. Figure 6.11 shows a picture of the final assembly, finger tips not included. In order to make the connection between the robot and the hardware control, several connection cables must be constructed and soldered. Wire connections were needed between the sensors and the control card, between the amplifiers and the control card and between the motors and the amplifiers.

In order to validate the position kinematic relations described in Chapter 4 and to calibrate the actual robot, external measurements of the position of the end-effectors have been performed and than compared to the encoder values predicted by the kinematic relations. In this setup, an external measurement was done using a vertical linear stage, with a precision of 50 μm. The linear stage is rigidly connected with bolts to a common base with the PentaG prototype. A plate with holes is fixed to the stage and the two end-effectors are bolted to the plate for various position, rotation and grasping values. The measurment setup is shown on Figure 6.12.

In the test, 30 different measurements are carried out on three height layers: 180mm, 205mm and 250mm. In each layer, 10 postures are measured, with different positions, rotation angles and grasping distances, chosen to cover a representative sampling of the entire useful workspace described in Chapter 5. The results were first used to calibrate some of the lengths of the robot links. Since each leg must be calibrated individually, the number of calibration parameters is 3X5 parameters for the legs and 5 parameters for the platform. The calibration method used is standard to robot calibration and uses a identification Jacobian matrix that describes the variations of the measurements as function of the varation of the kinematic parameters. This identification matrix is used in a quasi-Newton method algorythm to find the set of kinematic parameters that minimises the errors on the measurments. Most of the links have been manufactured to a tolerance of 0.05 mm which is precise enough to have very little influence on the kinematic. However, the parallelogram units were slightly longer than predicted due to the gluing imprecision between the carbon tube and the limb forks. Actual lengths of the parallelogram units were typically between 201 and 203 mm instead of the prescribed 200 mm. This was first revealed by the calibration procedure and then measured on the robot. Computation of the robot dexterity and workspace for these new parallelogram lengths show no significant influence on the dexterity of the configurable platform and a small increase in the size of the workspace. It was chosen to keep the same dimension for the useful workspace defined in Chapter 5. The correction on the other kinematic parameters was in general smaller than 0.05 mm. After these corrections in the kinematic model, the maximal difference between the kinematic model and the external measurement on the motor position, for the 30 measured position, was around 0.001 rad as shown on Figure 6.13. It is interesting to notice that leg 1, 3 and 5 present lower errors in general than leg 2 and 4. This is probably due to the fact that leg 1, 3 and 5 are directly connected to one of the end-effector while more joints and links separate leg 2 and 4 from the end-effectors, resulting in more kinematic uncertainties from play and tolorances. It should also be noted that those errors are in average about 20 times higher than the resolution of the encodor system.

Those values of course combine the uncertainties of both the robot and the external measurement setup. Since the linear stage is solidly bolted to the rigid base, that it has a high precision in translation, and since the position of the holes in the platform is known with a high precision, it is expected that the relative position certainty of the external measurement setup is much higher than the PentaG robot, as it should be. The initial distance between the reference frame of the robot and the reference frame of the external measurement setup was also obtain with a calibration procedure.

The total mass of the robot moving parts, including all bearings, hinges and axles but excluding the shafts of the motors and encoders located on the base is given in Table 6.5. The total mass is less than half kilo and thanks to the inertia-stiffness optimization, most of the mass is located near the motors, resulting in low perceived inertia at the end-effectors.

The stiffness if highly dependent on the position of the robot in the 5 DOF workspace. An experimental evaluation of stiffness performance can consist in measuring the compliant displacements due to a known external wrench. Nevertheless, the compliant displacements are composed of 3 linear and 3 angular components for each of the two end-effectors. In particular, the measurement of small angular compliant displacements can be very complex to achieve through commercially available sensors. The structural stiffness of a robotic system is closely related with the dynamic behaviour of the robotic system. Instead to determining set of stiffness that are only valid for a particular position and particular loads, it is interesting to study some limits of the performances that involve displacement over the complete useful workspace.

A simple PD position control was implemented using control gains $K_p = 100$ and $K_d = 5$

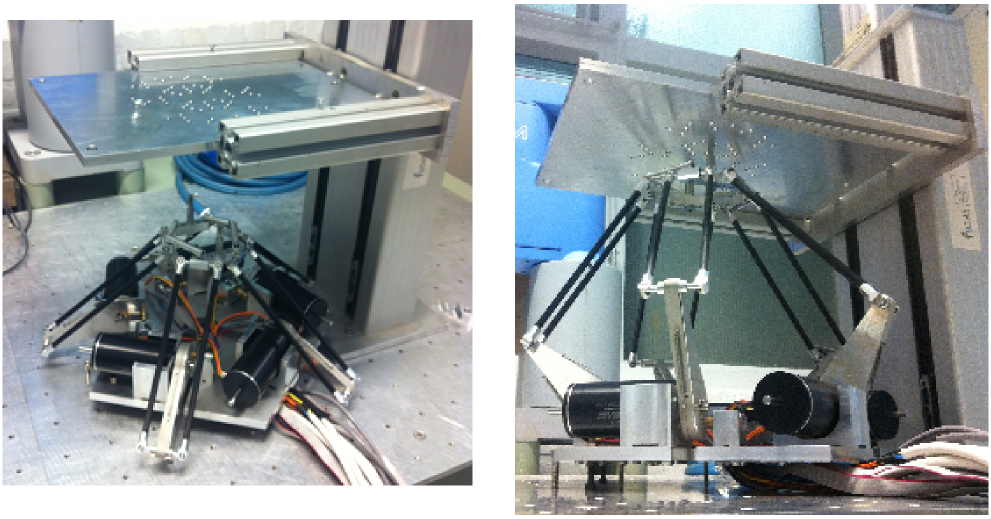


Figure 6.12: External end-effector measurement setup.

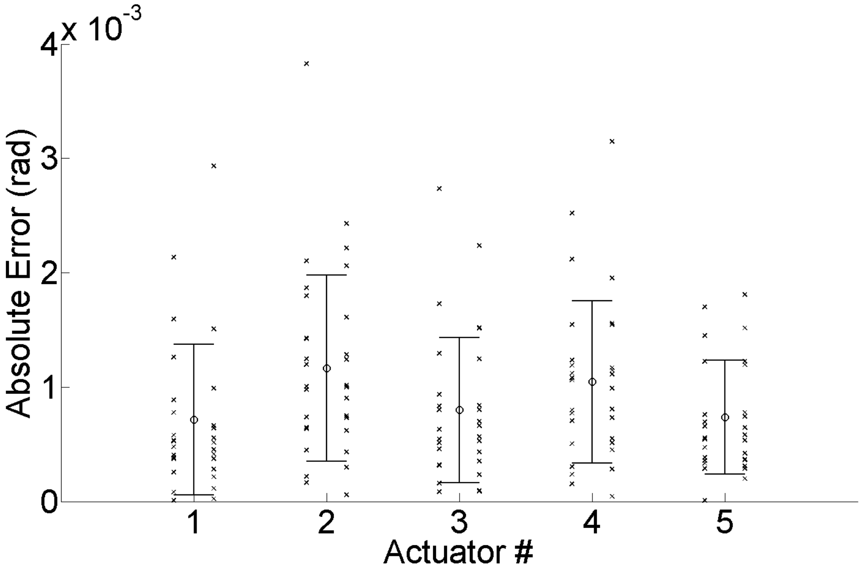


Figure 6.13: Distribution of the errors between the measured values and the values predicted by the kinematic model for 30 positions inside the useful workspace, after calibration.

Total mass and inertia contribution of the moving parts of the final design		
	Mass	Total Inertia Contribution
Configurable Platform	69.3 grams	0.100 N
5 Parallelogram Units	94.6 grams	0.117 N
5 Lowers Limbs	270.4 grams	0.094 N
5 actuator-sensor systems	-	0.015 N
Total	434.3 grams	0.326 N

Table 6.5: Total masses and inertia contribution at the finger tips of all the moving parts of the final design of the PentaG Prototype, including bearings, pins, inserts but excluding the motors and encoder shafts. Thanks to direct drive actuation, the total inertia contribution of the 5 motor-encodor systems is only 0.015 N, which brings the average inertia felt at the finger tips at 0.326 N.

Command motion covering the useful workspace	maximal frequency f	maximal smooth acceleration of the finger tips
$z = 0.2 + 0.06sin\,(2\pi ft)$ m	3.7 Hz	$32.4\,\mathrm{m/s^2}$
$y = 0.05sin(2\pi ft)$ m $z = 0.2$ m $\rho = 0.04$ m	3.5 Hz	$24.1\,\mathrm{m/s^2}$
$x = 0.05sin(2\pi ft)$ m $z = 0.2$ m $\rho = 0.04$ m	3.3 Hz	$21.5\,\mathrm{m/s^2}$
$\theta = 0.7sin(2\pi ft)$ rad $z = 0.2$ m $\rho = 0.04$ m	2.4 Hz	$12.7\,\mathrm{m/s^2}$
$\rho = 0.04 + 0.04sin(2\pi ft)$ m $z = 0.2$ m	3 Hz	$28.4\,\mathrm{m/s^2}$

Table 6.6: Maximal accelerations reached for a sinusoidal motion covering the entire useful workspace in each decoupled DOF of the PentaG prototype

in each DOF. The derivative of the error was computed numerically at a sampling time of 1 KHz. A command motion for each translational DOF was implemented as a sinusoidal motion, with an amplitude that cover the complete range of the useful workspace and the frequency of the motion was increased at each tests. The idea was to evaluate the maximum accelerations possible while keeping the tracking error below a certain threshold level, in this case 5×10^{-2} rad for all actuators. These results of course depend on the controller gains and the saturation voltage for the motors but also give a good idea of the structural stiffness of the robot over the complete useful workspace.

The maximum frequency reached for a sinusoidal motion that was continuously stable during that test was in the z direction at $f_z = 3.7$ Hz, which corresponds to an acceleration of $0.06 \cdot (2\pi \cdot 3.7)^2 = 32.4\,\mathrm{m/s^2}$ or around 3 G-forces which occur at the boundaries of the useful workspace. Other sinusoidal command motions with amplitude corresponding to the limits of the useful workspace were done in the horizontal plane at the middle height in the x and y direction. In every test, it is always the stability of the rotational DOF of the configurable platform that cause problems during high accelerations. An inspection of the dexterity of the robot shows that the lowest dexterity values are indeed for the rotational DOF and are located near the boundaries of the useful workspace.

6.5 Conclusion

In this chapter, the various design steps from an optimized kinematic model of the PentaG mechanism to the actual fabrication, assembly and realization of a complete haptic robot based on this architecture was presented. Some of the specific techniques developed during the design process are novel and can be adapted to the design of general parallel robots, regardless of the application or the presence of a configurable platform. Two of these general contributions are the inertia-stiffness optimization and the propagation of the specifications from the end-effector to the rest of the system by sampling of the workspace. The application of these two techniques is illustrated in the design of the haptic PentaG robot. The result is a haptic device with grasping capabilities that has a remarkably low inertia while having sufficient stiffness to render proper force feedback.

Chapter 7

Conclusions

7.1 Contribution of this thesis

The following contributions can be drawn from this thesis:

- A method based on screw theory to analytically calculate the local mobility and the overconstraints of general pure parallel mechanisms was introduced. In both cases, the solutions are the set of all finite screws in a set of screw systems that satisfy 6 homogeneous linear equations. It was already known that this solution, in the variable space, corresponds to the kernel of a matrix where each basis screw is represented by a column. It was also known that the dimension of this solution is equal to the difference between the sum of the dimensions of all the screw systems and the dimension of the screw systems sum. For each screw system included in the system of six homogeneous linear equations, the procedure creates two vector subspaces, namely a restricted screw system and a complement unrestricted screw systems. The restricted screw systems obtained represent either local mobility or overconstraints and their unique projection into the independent unrestricted screw systems represent their distribution in the whole pure parallel mechanism. The method only uses the vector space operations of intersections, complementary subspaces, and parallel projections. These operations are often well known for usual screw systems, which facilitate the use of configuration-free analytical vector calculation which are valid for the full-cycle workspace instead of configuration-dependent numerical computation which are only valid locally. In addition, analytical vector calculation can be used to detect geometric conditions for serial singularities and constraint singularities.
- The concept of series-parallel mechanism was defined based on the properties of their associated graph. A variety of already known classes of mechanisms, such as Delta robots and hybrid mechanisms, fall into this more general category of mechanism and their different layers of serial and parallel connections have been presented in this new general framework using the concept of graph reduction. For each class, the standard method of mobility and overconstraints analysis can be extended using this framework as long as the mechanism is series-parallel. General recursive formulas and methods have then been presented that are applicable to any series-parallel mechanism.

- It was seen that the series-parallel reduction of a parallel mechanism with configurable platform (PMCP), as defined in this thesis, always results in a wheel graph. Since wheel graphs fall into the category of non series-parallel graphs, the mobility and over-constraint analysis of PMCPs cannot be computed with the extended standard method for series-parallel mechanisms. Wheel graphs have the particular property of being self-dual and unlike most of other classes of mechanisms, PMCPs have a self dual topology. This is due to the fact that pure parallel mechanisms are dual to a single closed loop and vice-versa. Combining a single closed-loop and a parallel mechanism together results in a self dual mechanism. The particular self dual topology of PMCPs can be exploited in their analysis. Any general method developed for calculating the mobility of PMCPs can be reused directly to calculate their overconstraints and vice-versa.

- A general method was proposed to perform the analysis of the mobility of PMCPs. One key aspect of the new proposed method is based on the concept of restricted and unrestricted screw systems of the legs. In the mobility analysis of PMCPs, the restricted screw system between two legs corresponds to the mobility of the legs relatively to the base when they are connected together with a rigid platform. The unrestricted screw system corresponds to their relative mobility when they are not connected. Both the restricted and unrestricted screw systems represent independent type of mobility. By using an appropriated change of variables in a matrix representation, the constraints imposed by the configurable platform on the mobility of the legs is calculated. The process is repeated for each platform connection until an analytical matrix representation of the mobility of each leg as function of the global DOF of the mechanism is finally obtained.

- The general overconstraint analysis method of PMCPs was introduced based on the self-dual topology of those mechanisms. Since the homogenous conditions imposed by the closed-loops on the joint velocities is analogue to the homogeneous condition imposed on the rigid links by the internal stresses due to the over-constraints, and since wheel graph are self-dual, the method can also be applied in a dual way to the analysis of the over-constraints of the mechanism. The input for the over-constraints analysis method is the dual graph of the reciprocal screw systems of the input of the mobility analysis method. Overconstraint analysis is useful to determine the manufacturing tolerances that allow the assembly of the mechanism. Fundamentals of mobility and over-constraints analysis of PMCPs is an essential aspect in the future development of a broader type synthesis method of mechanism that includes PMCPs.

- The architecture of the PentaG robot, a 5 DOF parallel robot with configurable platform (PRCP) that generates 3T1R in addition to grasping capabilities was introduced. It is the first operational robot that offers grasping capabilities from a configurable platform while all the motors are located on the base. This concept offers in principle a higher structural stiffness than conventional parallel robot with grasping actuator attached to a rigid platform.

- The inverse position kinematics and the inverse instantaneous velocity of the PentaG robot were presented. In both cases, the problem must be divided into two subproblems where the kinematics of the platform is calculated before the kinematics of the legs. This two step method can easily be generalized to other PRCPs. It follows that the standard method for the static analysis of pure parallel robots must also be modified in the case of PRCPs. The static analysis also required two step in which the static of the legs is calculated before the statics of the configurable platform.

- Serial singularity conditions of the PentaG robot were derived and a new type of serial singularity, specific to PRCPs was described. In addition to the serial singularities that can occur in the legs of PRCPs, the configurable platform can also present serial platform singularities that are not related to the configuration of the legs. Serial singularities represent boundaries of the various working modes of a robot and the 128 working modes of the PentaG robot were described, including the working mode generated by the presence of serial platform singularities.
- Parallel singularities occur in PRCPs when the configurable platform can instantly move in such a way that it produces at the same time on every leg attach point either no movement or a movement that generates no motion of the leg actuators. In the case of the PentaG robot, those legs attach point movements correspond to a motion in a direction perpendicular to the direction of the parallelogram units of the legs. Two distinct parallel singularities were found inside the workspace of the PentaG robot. Each of them forms a 4-dimensional hypersurface in the 5-dimensional end-effectors space.
- A novel static balancing method was proposed that reduces the number of springs and additional links needed to achieve perfect balancing of translational parallel robot. The method is based on the fact that for those robots, the platform is already constrained in orientation by the kinematic of the robot and can be used as a fixed attach point for springs that balance the part of the legs near the platform. The method can be applied to any translational parallel robot with or without configurable platform including the Delta robot and the PentaG robot.
- A general method to calculate the maximal number of independent kinematic parameters, which was based on the Denavit-Hartenburg convention, was proposed and can be used for any closed-loop mechanism. For each independent loop that is closed in a general mechanism, 6 parameters which represent either joint positions or geometric condition between joint axes become dependent and the proposed notation is based on this principle.
- A novel inertia-stiffness optimisation method was proposed which is independent of the kinematic of the robot. By linking the stiffness and inertia contribution of each link of the robot with the size of the cross-section of the link, it is possible to improve the structural stiffness of the robot by distributing most of the inertia near the base of the structure while keeping an acceptable stiffness. The method is novel and applicable to any parallel robot and has the interesting advantage of not affecting the kinematic parameters of the robot, which means that this optimization method can be performed on a later stage of the design process where the kinematic parameters are already fixed.

7.2 Some Remarks

- As seen during the kinematic optimisation of the PentaG robot, symmetry in the design is a highly desirable feature that greatly reduces the complexity of the kinematic optimisation problem. It is also desirable because it reduces the complexity of the performance analysis and generally offers more homogeneous performances throughout the workspace.

- In a haptic device based on parallel robot architecture, the useful workspace of the device should be limited to a centered singularity-free subregion of the complete workspace where the operator can move freely. The useful workspace should be defined as a simple shape that can easily be recognised by the operator.
- Despite their inherent disadvantages, such as the lack of certainty to find a global optimum and the lack of constant optimisation response time, Genetic Algorithms are powerful to quickly find good solutions to complex optimisation problems with a large number of parameters and discontinuous and sensitive objective function. They should be seriously considered when traditional optimisation approaches fail.
- Due to the extreme large range of different transfer functions over the workspace of parallel robots, a statistical approach of the distribution of the propagation of the specifications from the end-effector to the actuators-sensors system is useful to prevent over-design of the hardware control system for one particular specification that would make the satisfaction of the other specifications difficult to achieve.
- The main advantage of PRCPs is that they combine motions and grasping capabilities into a structure hat provides an inherent high structural stiffness. Two applications that can benefit from this new class of architecture are high-speed pick-and-place robots and haptic devices. In both applications, addition of grasping capabilities can be beneficial for some tasks and a high mechanical structural stiffness improves the performance of the device given a certain hardware and software control system.

Bibliography

[1] RS Ball. *A Treatise on the Theory of Screws.* Cambridge University Press, Cambridge, 1900.

[2] C. Baradat, V. Arakelian, S. Briot, and S. Guegan. Design and prototyping of a new balancing mechanism for spatial parallel manipulators. *Journal of Mechanical Design*, 130:072305, 2008.

[3] G. Barrette and C. Gosselin. Kinematic analysis and design of planar parallel mechanism actuated with cables. In *ASME Design Engineering Technical Conference*, 2000.

[4] F. Bleicher. *Optimizing a three-axes machine-tool with parallel kinematic structure.* 2002.

[5] I.A. Bonev, D. Zlatanov, and C.M. Gosselin. Singularity analysis of 3-dof planar parallel mechanisms via screw theory. *Journal of Mechanical Design*, 125:573, 2003.

[6] T.L. Brooks. Telerobotic response requirements. In *Systems, Man and Cybernetics, 1990. Conference Proceedings., IEEE International Conference on*, pages 113–120. IEEE, 1990.

[7] G. Carwardine. Improvements in equipoising mechanism. *UK Patent*, 433, 1935.

[8] D. Chablat and P. Wenger. Architecture optimization of a 3-dof translational parallel mechanism for machining applications, the orthoglide. *Robotics and Automation, IEEE Transactions on*, 19(3):403–410, 2003.

[9] PA Chebychevl. Théorie des mécanismes connus sous le nom de parallélogrammes. *2 ème partie, Mémoires présentés à l'Académie impériale des sciences de Saint-Pétersbourg par divers savants*, 1854.

[10] R. Clavel. Delta, a fast robot with parallel geometry. In *Proc 18th International Symposium on Industrial Robots*, pages 91–100, 1988.

[11] J.S. Dai, Z. Huang, and H. Lipkin. Mobility of overconstrained parallel mechanisms. *Journal of Mechanical Design*, 128:220, 2006.

[12] J.S. Dai and J. Rees Jones. Interrelationship between screw systems and corresponding reciprocal systems and applications. *Mechanism and machine theory*, 36(5):633–651, 2001.

[13] TH Davies. Simple examples of dual coupling networks.

[14] TH Davies. Kirchhoff's circulation law applied to multi-loop kinematic chains. *Mechanism and machine theory*, 16(3):171–183, 1981.

[15] J. Denavit. A kinematic notation for lower-pair mechanisms based on matrices. *Trans. of the ASME. Journal of Applied Mechanics*, 22:215–221, 1955.

[16] Y. Dong, Y. Byung-Ju, and K. Wheekuk. Design of a new grasper having xyz translational motions. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 1, pages 690–695. IEEE, 2003.

[17] I. Ebert-Uphoff, C.M. Gosselin, and T. Laliberté. Static balancing of spatial parallel platform mechanismsâĂŤrevisited. *Journal of Mechanical Design*, 122:43, 2000.

[18] AY Elatta, F. LiangZhi, and L. XiaoPing. Performance evaluation and kinematic calibration for hybrid 5dof manipulator based on 3-rps parallel mechanism. *Journal of Applied Sciences*, 4(2):214–224, 2004.

[19] L. Everett, M. Driels, and B. Mooring. Kinematic modelling for robot calibration. In *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, volume 4, pages 183–189. IEEE, 1987.

[20] M. Fayet. Distribution of wrench-twist duality in over-constrained mechanisms. *Journal of mechanical design*, 125(1):81–91, 2003.

[21] G. Gatti and G. Danieli. A practical approach to compensate for geometric errors in measuring arms: application to a six-degree-of-freedom kinematic structure. *Measurement Science and Technology*, 19:015107, 2008.

[22] G. Gogu. Mobility of mechanisms: a critical review. *Mechanism and machine Theory*, 40(9):1068–1097, 2005.

[23] G. Gogu. *Structural synthesis of parallel robots: Part 1: Methodology*, volume 149. Springer, 2007.

[24] C. Gosselin. *Kinematic analysis, optimization and programming of parallel robotic manipulators.* McGill University, 1989.

[25] C. Gosselin and J. Angeles. The optimum kinematic design of a spherical three-degree-of-freedom parallel manipulator. *Journal of mechanisms, transmissions, and automation in design*, 111(2):202–207, 1989.

[26] C. Gosselin and J. Angeles. Singularity analysis of closed-loop kinematic chains. *Robotics and Automation, IEEE Transactions on*, 6(3):281–290, 1990.

[27] C.M. Gosselin. Static balancing of spherical 3-dof parallel mechanisms and manipulators. *The International Journal of Robotics Research*, 18(8):819–829, 1999.

[28] C.M. Gosselin and J. Wang. Static balancing of spatial six-degree-of-freedom parallel mechanisms with revolute actuators. *Journal of Robotic Systems*, 17(3):159–170, 2000.

[29] VE Gough and SG Whitehall. Universal tyre test machine. In *Proc. FISITA 9th Int. Technical Congress*, pages 117–137, 1962.

[30] M. Grübler. *Allgemeine Eigenschaften der zwangläufigen ebenen kinematischen Ketten.* L. Simion, 1884.

[31] K.S. Hale and K.M. Stanney. Deriving haptic design guidelines from human physiological, psychophysical, and neurological foundations. *Computer Graphics and Applications, IEEE*, 24(2):33–39, 2004.

[32] J.L. Herder. Energy-free systems: theory, conception, and design of statically balanced spring mechanisms. 2001.

[33] J.L. Herder. Some considerations regarding statically balanced parallel mechanisms (sbpm's). In *Proceedings of the Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators*, pages 3–4, 2002.

[34] JM Hervé. The lie group of rigid body displacements, a fundamental tool for mechanism design. *Mechanism and Machine Theory*, 34(5):719–730, 1999.

[35] J.H. Holland and J.S. Reitman. Cognitive systems based on adaptive algorithms. *ACM SIGART Bulletin*, (63):49–49, 1977.

[36] K.H. Hunt. *Kinematic geometry of mechanisms.* Cambridge Univ Press, 1978.

[37] KH Hunt. Structural kinematics of in-parallel-actuated robot-arms. In *American Society of Mechanical Engineers, Design and Production Engineering Technical Conference, Washington, DC*, page 1982, 1982.

[38] M.L. Husty. An algorithm for solving the direct kinematics of general stewart-gough platforms. *Mechanism and Machine Theory*, 31(4):365–379, 1996.

[39] M. Jean and C.M. Gosselin. Static balancing of planar parallel manipulators. In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, volume 4, pages 3732–3737. Ieee, 1996.

[40] S.A. Joshi and L.W. Tsai. Jacobian analysis of limited-dof parallel manipulators. *Journal of Mechanical Design*, 124:254, 2002.

[41] S.M. Kim, W. Kim, and B.J. Yi. Kinematic analysis and optimal design of a 3t1r type parallel mechanism. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 2199–2204. IEEE, 2009.

[42] X. Kong and C.M. Gosselin. *Type synthesis of parallel mechanisms.* Springer Publishing Company, Incorporated, 2007.

[43] K. Kutzbach. Mechanische leitungsverzweigung, ihre gesetze und anwendungen. *Maschinenbau, Der Betrieb*, 8:710–716, 1929.

[44] H.Y. Lee and C.F. Reinholtz. Inverse kinematics of serial-chain manipulators. *Journal of Mechanical Design*, 118:396, 1996.

[45] HY Lee, C. Woernle, and M. Hiller. A complete solution for the inverse kinematic problem of the general 6r robot manipulator. *Journal of Mechanical Design*, 113:481, 1991.

[46] J.H. Lee, K.S. Eom, and I.I. Suh. Design of a new 6-dof parallel haptic device. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 1, pages 886–891. IEEE, 2001.

[47] X.J. Liu, Z.L. Jin, and F. Gao. Optimum design of 3-dof spherical parallel manipulators with respect to the conditioning and stiffness indices. *Mechanism and Machine Theory*, 35(9):1257–1267, 2000.

[48] M.J.H. Lum, J. Rosen, M.N. Sinanan, and B. Hannaford. Optimization of a spherical mechanism for a minimally invasive surgical robot: theoretical and experimental approaches. *Biomedical Engineering, IEEE Transactions on*, 53(7):1440–1445, 2006.

[49] J.G. Luo and M.Y. He. Research on the configurations and singularity of serial-parallel mechanism. *Advanced Materials Research*, 393:265–268, 2012.

[50] R. Manseur and K.L. Doty. A robot manipulator with 16 real inverse kinematic solution sets. *The International Journal of Robotics Research*, 8(5):75–79, 1989.

[51] J.P. Merlet. *Parallel robots.* Springer, 2006.

[52] J.P. Merlet, C.M. Gosselin, and N. Mouly. Workspaces of planar parallel manipulators. *Mechanism and Machine Theory*, 33(1):7–20, 1998.

[53] G. Michelitsch, A. Ruf, H. van Veen, and J. van Erp. Multi-finger haptic interaction within the miamm project. In *Proceedings of Eurohaptics*, pages 144–149, 2002.

[54] K. Miller. Optimal design and modeling of spatial parallel manipulators. *The International Journal of Robotics Research*, 23(2):127–140, 2004.

[55] MG Mohamed and J. Duffy. A direct determination of the instantaneous kinematics of fully parallel robot manipulators. *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, 107(2):226–229, 1985.

[56] M.G. Mohamed and C.M. Gosselin. Design and analysis of kinematically redundant parallel manipulators with configurable platforms. *Robotics, IEEE Transactions on*, 21(3):277–287, 2005.

[57] M. Monroy, M. Oyarzabal, M. Ferre, A. Campos, and J. Barrio. Masterfinger: Multi-finger haptic interface for collaborative environments. *Haptics: Perception, Devices and Scenarios*, pages 411–419, 2008.

[58] B.W. Mooring, Z.S. Roth, and M.R. Driels. *Fundamentals of manipulator calibration.* Wiley New York, 1991.

[59] V. Nabat, M. de la O RODRIGUEZ, S. Krut, F. Pierrot, et al. Par4: very high speed parallel robot for pick-and-place. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 553–558. IEEE, 2005.

[60] Z. Najdovski and S. Nahavandi. Extending haptic device capability for 3d virtual grasping. *Haptics: Perception, Devices and Scenarios*, pages 494–503, 2008.

[61] MS Ohwovoriole and B. Roth. An extension of screw theory. *ASME Journal of Mechanical Design*, 103(4):725–735, 1981.

[62] B.J. Park, B.J. Yi, and W.K. Kim. Design and analysis of a new parallel grasper having spherical motion. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 1, pages 106–111. IEEE, 2004.

[63] S. Perreault and C.M. Gosselin. Cable-driven parallel mechanisms: Application to a locomotion interface. *Journal of Mechanical Design*, 130:102301, 2008.

[64] J. Phillips. Freedom in machinery, vol. 2: screw theory exemplified, 1990.

[65] J. Phillips. Freedom in machinery, vol. 1: Introducing screw theory, 1984.

[66] EJF Primrose. On the input-output equation of the general 7r-mechanism. *Mechanism and Machine Theory*, 21(6):509–510, 1986.

[67] M. Raghavan and B. Roth. Inverse kinematics of the general 6r manipulator and related linkages. *Journal of Mechanical Design*, 115:502, 1993.

[68] A. Russo, R. Sinatra, and F. Xi. Static balancing of parallel robots. *Mechanism and machine theory*, 40(2):191–202, 2005.

[69] K. Schröer, S.L. Albright, and M. Grethlein. Complete, minimal and model-continuous kinematic models for robot calibration. *Robotics and Computer-Integrated Manufacturing*, 13(1):73–85, 1997.

[70] K.B. Shimoga. A survey of perceptual feedback issues in dexterous telemanipulation. ii. finger touch feedback. In *Virtual Reality Annual International Symposium, 1993., 1993 IEEE*, pages 271–279. IEEE, 1993.

[71] R.E. Stamper, L.W. Tsai, and G.C. Walsh. Optimization of a three dof translational platform for well-conditioned workspace. In *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, volume 4, pages 3250–3255. IEEE, 1997.

[72] S.D. Stan, M. Manic, V. Maties, and R. Balan. Evolutionary approach to optimal design of 3 dof translation exoskeleton and medical parallel robots. In *Human System Interactions, 2008 Conference on*, pages 720–725. Ieee, 2008.

[73] DA Streit and E. Shin. Equilibrators for planar linkages. *Journal of Mechanical Design*, 115:604, 1993.

[74] H.Z. Tan, M.A. Srinivasan, B. Eberman, and B. Cheng. Human factors for the design of force-reflecting haptic interfaces. *Dynamic Systems and Control*, 55(1):353–359, 1994.

[75] T.K. Tanev. Kinematics of a hybrid (parallel–serial) robot manipulator. *Mechanism and machine theory*, 35(9):1183–1196, 2000.

[76] U. Thomas and F. Wahl. A unified notation for serial, parallel and hybrid kinematic structures. *Robotic Systems for Handling and Assembly*, pages 3–15, 2011.

[77] K.J. Waldron. The constraint analysis of mechanisms. *Journal of Mechanisms*, 1(2):101–114, 1966.

[78] H. Wang and K.C. Fan. Identification of strut and assembly errors of a 3-prs serial–parallel machine tool. *International Journal of Machine Tools and Manufacture*, 44(11):1171–1178, 2004.

[79] F. Wen and C. Liang. Displacement analysis of the 6-6 stewart platform mechanisms. *Mechanism and Machine Theory*, 29(4):547–557, 1994.

[80] V. Wijk and JL Herder. Dynamic balancing of clavelâĂŹs delta robot. *Computational Kinematics*, pages 315–322, 2009.

[81] B.J. Yi, H.Y. Na, J.H. Lee, Y.S. Hong, S.R. Oh, I.H. Suh, and W.K. Kim. Design of a parallel-type gripper mechanism. *The International Journal of Robotics Research*, 21(7):661–676, 2002.

[82] J. Yoon and J. Ryu. Design, fabrication, and evaluation of a new haptic device using a parallel mechanism. *Mechatronics, IEEE/ASME Transactions on*, 6(3):221–233, 2001.

[83] D. Zlatanov, I.A. Bonev, and C.M. Gosselin. Constraint singularities of parallel mechanisms. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 1, pages 496–502. IEEE, 2002.

[84] D.S. Zlatanov, R.G. Fenton, and B. Benhabib. Classification and interpretation of singularities of redundant mechanisms. In *Proceedings of the 24th ASME Annual Design Automation Conference (DETCâĂŹ98)*, pages 1–11, 1998.

[85] M. Zoppi, D. Zlatanov, and R. Molfino. On the velocity analysis of interconnected chains mechanisms. *Mechanism and machine theory*, 41(11):1346–1358, 2006.

Summary

This thesis explores the fundamentals of a new class of parallel mechanisms called parallel mechanisms with configurable platforms as well as the design and analysis of parallel robots that are based on those mechanisms. Pure parallel robots are formed by two rigid links, the base and the end-effector, connected in parallel by independent serial chains, called legs. The novel concept behind parallel robots with configurable platforms is that the rigid (non-configurable) end-effector is replaced by a closed-loop chain (the configurable platform). Some of the links of this closed-loop chain are attached to the legs so its configuration can be fully controlled from the motors located on the base. The use of a closed-loop chain instead of a rigid end-effector allows the robot to interact with the environment from multiple contact points on the platform. This results in a robot that can combine motion and grasping capabilities into a structure that provides an inherent high structural stiffness. All the actuators are grounded on the base requiring no additional motor at the end-effector location to provide the grasping. High-speed pick-and-place robots and haptic interfaces are examples of devices that can benefit from this new type of architecture. In both applications, the addition of grasping capabilities is needed for some tasks and a high mechanical structural stiffness is desirable to improve the performance of the device given a certain hardware and software control system.

This thesis is structured into two distinct parts. The first part explores the fundamentals of the topology of parallel mechanisms with configurable platform (PMCPs) and proposes general methods to obtain an analytical vector representation of the distribution of their mobility and overconstraints via Screw Theory and Graph Theory. A motivation for this analysis is that a better understanding of the fundamentals of PMCPs will help robot designers to consider them as a valid option in their choices of robot architectures. Also, this analysis constitutes also an essential aspect in a future development of a broader type synthesis method that includes those mechanisms. The second part focuses on the general input-output relations of parallel robots with configurable platforms (PRCPs) and presents the design, optimisation and implementation for haptic application of the PentaG robot, a novel 5 DOF parallel robot that includes a 2 DOF configurable platform. The motivations for the design and implementation of the PentaG prototype are to show the feasibility of PRCPs that use the configurable platform as a grasping device and to see to what extent the design principles and analysis tools that are generally used in the study and implementation of pure parallel robots can be applied or must be modified in the design and analysis of PRCPs.

First, a state of the art method for the mobility analysis of overconstrained pure parallel mechanisms is presented. Mechanisms that do not satisfy the Chebichev criterion of mobility are nowadays generally called overconstrained mechanisms because their constraint equations are not full rank. Mobility and overconstraints of mechanisms are intimately

linked since each overconstraint creates an extra mobility in regard of the Chebichev criterion. The use of numerical values to compute the rank of these equations is sensitive to round-off errors and the solutions obtained from numerical values for the distribution of the mobility and overconstraints are only valid for a particular position of the mechanism. The concept of restricted screw systems is then introduced to analytically calculate both the local mobility and overconstraints distribution of pure parallel mechanisms using vector space operations of intersections, complementary subspaces, and parallel projections on screw systems.

A topological analysis of mechanisms that are not purely parallel, such as Hybrid and Delta mechanisms, is then presented via Graph Theory. Current existing methods for the mobility and overconstraint analysis of mechanisms that are not purely parallel but have a series-parallel graph are unified with a graph reduction analysis of their topologies. Their several layers of local mobilities and overconstraints are described and recursive formulas are derived for the general case. It is seen that the upper topological layer of PMCPs is always represented by a wheel graph, which is a non series-parallel graph. A method based on a matrix representation of the mobility of their legs and the concept of restricted screw systems is used to obtain an analytical distribution of their global mobility. An important self-dual property of their topology, that is particular to PMCPs, is then revealed and is exploited to extend the method used for their mobility analysis to their overconstraint analysis. Two mechanisms, a 4-RRR with a 8R configurable platform mechanism and the PentaG mechanism, are used to illustrate the method and an analytical representation of the distribution of their global mobilities and overconstraints is presented.

The second part of this thesis starts with the introduction of the PentaG robot and its kinematic analysis. Unlike classical parallel robots, the inverse position and velocity kinematic analysis of PRCPs must be performed in a two stages method where the kinematics of the platform is calculated before the kinematics of the legs. Serial singular configurations are defined analytically and the working modes of the PentaG robot are enumerated. Parallel singular configurations are obtained numerically. New types of serial and parallel singular configurations related to the platform and particular to PRCPs are then described. In addition, a novel static balancing method that reduces the number of springs and additional links needed to achieve perfect static balancing of translational parallel robots with or without configurable platforms is presented.

A geometrical optimisation of the PentaG robot for use as a haptic device is then performed. The concept of the haptic useful workspace is presented and the robot is optimised in order to maximise its compactness relatively to the size of the useful workspace. Several techniques are shown to reduce the number of kinematic parameters to a number that is manageable and an efficient algorithm is developed to reduce the computation time of the objective function. Due to the discontinuities and local optima of the objective function, Genetic Algorithms are used for the search for a optimal solution.

Finally, the detailed design and implementation of a prototype of the PentaG robot for haptic application is presented. The kinematic parameters obtained from the optimisation process are used for this prototype and a new method to optimise the structural stiffness of the robot without modifying its kinematic parameters is presented. The method links the stiffness and inertia contribution of each link of the robot to its cross-sectional area and improves the structural stiffness of the robot by distributing most of the inertia near the base of the structure while keeping an acceptable stiffness. Specifications are defined at the fingertips of the robot based on human ergonomics and the propagation of those specifications to the design of the motor-sensor system is done using a statistical distribution over the useful workspace. The implementation of the final design is presented and both simulations and experiments are used to show the validity of the final prototype and new design methods.

Samenvatting

Deze thesis verkent de beginselen van een nieuwe klasse parallelle mechanismen genaamd ‘parallelle mechanismen met configureerbare platformen’ alsmede het ontwerp en de analyse van robots die gebaseerd zijn op deze mechanismen. Zuiver parallelle robots bestaan uit twee rigide verbindingselementen, de basis en de eindeffector, die parallel verbonden zijn door middel van onafhankelijke seriële ketens, genaamd ‘benen’. Het nieuwe concept dat ten grondslag ligt aan parallelle mechanismen met configureerbare platformen is dat de rigide (niet-configureerbare) eindeffector vervangen is door een gesloten keten (het configureerbare platform). Sommige verbindingselementen van deze gesloten keten zijn verbonden aan de benen, zodat de configuratie van het platform volledig gecontroleerd kan worden met de motoren die zich op de basis bevinden. Het gebruik van een gesloten keten in plaats van een rigide eindeffector stelt de robot in staat om te interacteren met de omgeving vanuit meerdere contactpunten op het platform. Dit resulteert in een robot die het vermogen om te bewegen en grijpen combineert in een mechanisme met een inherent hoge, structurele stijfheid, omdat er geen motoren nodig zijn op de eindeffector om te kunnen grijpen, gezien deze allemaal direct verbonden zijn aan de basis. Hoge snelheid pak-en-plaats robots en haptische apparaten kunnen beiden profiteren van dit type architectuur. Het vermogen om te grijpen is in beide toepassingen noodzakelijk om sommige taken uit te voeren en daarbij is een hoge mechanische, structurele stijfheid gewenst om de prestatie van het apparaat te verbeteren, gegeven bepaalde hardware en een bepaald softwarematige controle systeem.

Deze thesis is gestructureerd in twee afzonderlijke delen. Het eerste deel verkent de beginselen van de topologie van parallelle mechanismen met configureerbare platformen (PMCPs) en draagt algemene methoden voor om een analytische vectorrepresentatie van de distributie van hun beweeglijkheid en overbepaaldheid te verkrijgen via Screw Theory en grafentheorie. Een motivatie voor deze analyse is dat een beter begrip van de beginselen van PMCPs robot ontwerpers zal helpen om dit type mechanisme te overwegen als een realistische optie voor de robot architectuur. Tevens vormt deze analyse een essentieel aspect in de toekomstige ontwikkeling van een meer algemene ‘type synthesemethode’ welke ook deze mechanismen bevat. Het tweede gedeelte van deze thesis richt zich op de algemene input-output relaties van parallelle robots met configureerbare platformen (PRCPs) en presenteert het ontwerp, de optimalisatie en de implementatie voor haptische toepassingen van de PentaG robot. De PentaG robot is een nieuwe, vijf graden van vrijheid parallel robot waarvan 2 graden van vrijheid zijn te danken aan het configureerbare platform. De motivatie voor het ontwerp en de implementatie van het prototype van de PentaG is ten eerste om de haalbaarheid aan te tonen van PRCPs die een configureerbaar platform gebruiken om te grijpen en ten tweede om te bepalen tot op welke hoogte de ontwerpprincipes en analysetools die in het algemeen gebruikt worden in

de studie en implementatie van zuiver parallelle robots toegepast kunnen worden, of dat deze moeten worden gemodificeerd voor het ontwerp en de analyse van PRCPs.

In het eerste deel wordt een state-of-the-art methode gepresenteerd voor de analyse van de beweeglijkheid en overbepaaldheid van zuiver parallelle mechanismen. Mechanismen die niet voldoen aan het Chebichev criterium van beweeglijkheid worden heden dag over het algemeen overbepaalde mechanismen genoemd omdat hun overbepaaldheidsvergelijkingen matrices bevatten die geen volle rang hebben. Beweeglijkheid en overbepaaldheid van mechanismen zijn nauw verbonden omdat elke overbepaaldheid een extra mobiliteit creëert volgens het Chebichev criterium. Het gebruik van numerieke waardes om de rang van deze matrices te bepalen is gevoelig voor afrondfouten en de oplossingen die met numerieke waardes worden verkregen voor de distributie van de beweeglijkheid en overbepaaldheid zijn enkel geldig voor een specifieke positie van het mechanisme. Het concept van beperkte Screw systemen wordt vervolgens geïntroduceerd om zowel de lokale beweeglijkheid en verdeling van overbepaaldheden van zuiver parallelle mechanismen te berekenen met behulp van intersectieoperaties op de vectorruimte, complementaire deelruimtes, en parallelle projecties op Screw systemen.

Een topologische analyse van mechanismen dat niet alleen puur parallel zijn, zoals hybride en delta mechanismen, is vervolgens gepresenteerd door middel van grafentheorie. Huidig bestaande methoden voor de analyse van beweeglijkheid en overbepaaldheid van mechanisme dat niet geheel parallel maar een serie-parallel graaf hebben zijn samengevoegd met een graaf reductie analyse van hun topology. Hun meerdere lagen van lokale beweeglijkheid en overbepaaldheid zijn beschreven en recursieve formules zijn herleid in het algemeen. Het is gezien dat de bovenste topologische laag van PMCPs altijd weergegeven worden met een wielgraaf, wat een niet serie-parallele graaf is. Een methode gebaseerd op een matrix weergave van de beweeglijkheid van hun benen en het concept van systemen van Screw zijn gebruikt om een analytische verdeling te verkrijgen van hun globale beweeglijkheid. Een belangrijke zelfduale eigenschap van hun topologie, wat een specifieke eigenschap is van PMCPs, is dan onthuld en gebruikt om de methode voor hun beweeglijkheids analyse uit te breiden naar hun analyse van overbepaaldheid. Twee mechanismen, een 4-RRR met een 8R configureerbaar platform mechanismen en de PentaG mechanismen, zijn gebruikt om de methode te illustreren en een analytische weergave van de verdeling van hun globale beweeglijkheid en overbepaaldheid is gepresenteerd.

Het tweede gedeelte van de thesis begint met een introductie van de PentaG robot en zijn kinematische analyse. In tegenstelling tot klassieke parallele robots, de inverse positie en snelheid kinematische analyse van PRCPs moet uitgevoerd worden in een twee stappen methode waarbij de kinematica van het platform is berekend voordat de kinematica van de benen. Seriële singuliere configuraties zijn analytisch bepaald en de werkings-modes van de PentaGrobot zijn opgesomd. Parallelle singuliere configuraties gerelateerd aan het platform en in het bijzonder aan PRCPs zijn dan beschreven. Ter aanvulling, een nieuw statisch gebalanceerde methode dat het aantal veren reduceert en aanvullende verbindingselementen nodig heeft om perfect statisch balans te bereiken van transleerbaar parallele robots met of zonder configureerbare platformen is gepresenteerd.

Een meetkundige optimisatie van de PentaG robot voor het gebruik van haptische apparaten is vervolgens uitgevoerd. Het concept van het haptisch bruikbare werkgebied is gepresenteerd en de robot is geoptimaliseerd teneinde de compactheid te maximaliseren in zijn compactheid relatief aan de grootte van het bruikbare werkgebied. Meerdere technieken zijn weergegeven om het aantal kinematische parameters te reduceren tot een hanteerbaar aantal en een efficiënt algoritme is ontwikkeld om de berekeningstijd van het optimaliseringsprobleem te reduceren, genetische algoritmen zijn gebruikt bij het zoeken naar een optimale oplossing.

Ten slotte is het gedetailleerde ontwerp en de implementatie van een prototype van de PentaG robot voor een haptische applicatie gepresenteerd. De kinematische parameters die verkregen zijn van het optimalisatie proces zijn gebruikt voor dit prototype en een nieuwe methode om de stijfheid van het mechanisme van de robot zonder aanpassing van de kinematische parameters te optimaliseren is gepresenteerd. De methode koppelt de stijfheids- en traagheidsbijdrage van elk verbindingselement van de robot tot zijn doorsnedeoppervlakte en verbeterd de stijfheid van het mechanisme van de robot door het grootste gedeelte van de traagheid bij de basis te verdelen van het mechanisme terwijl een acceptabele stijfheid gehanteerd blijft. Specificaties zijn bij de vingertoppen van de robot gedefinieerd gebaseerd op menselijke ergonomie en de voortzetting van deze specificaties tot het ontwerp van het motor meet systeem is gedaan door gebruik te maken van statistische verdeling over het gehele werkgebied. De implementatie van het uiteindelijke ontwerp is gepresenteerd en zowel simulaties als experimenten zijn gebruik om de geldigheid aan te tonen van het uiteindelijke prototype en de nieuwe ontwerp methoden.

Acknowledgements

Getting through my PhD was not only the result of an individual effort, but was also made possible with the help, advice and support from many people I met during this journey. First, I would like to thank professors and supervisors Rob, Hans, Jo, Frans, John, David and Just for all the scientific as well as professional advices and feedbacks I got from you during that period. I had the chance to work with excellent researchers coming from different fields and that mixture of different expertise and interests was a great opportunity for me to learn and to consider various aspects in my research.

I would like to address a special thank to Just Herder who, despite having no official contract with me at the time, accepted to “adopt” me as a PhD when I lost my former supervisor. Your guidance was very useful and appreciated and our shared interests in mechanisms often led to stimulating and fun discussions.

Thanks also to my former project colleagues, Pablo, Ilhan, Shahzad and Marcello for all the inputs you gave on this project. A special thank to the master students I had the chance to supervise during this research. Marijke, Adhy, Johan, Oscar and Hongfei, it was a pleasure to work with you. I learned from you at least as much as I could teach you. Also thanks to my current colleagues Teun, Roel, Jeroen, Henri, Bram and Jack, as well as Nima and Juan, not only for your valuable discussions and advices but also for the friendly support.

Finally, I would like to thank my parents and my family back in Canada for your love, support and understanding during that period, even from far away.

Curriculum Vitae

15 January 1981 Born in Shawinigan-Sud, Canada.

Employement and Education

1998-2000 Natural Science, Collège Shawinigan, Shawinigan, Canada

2000-2004 B. Sc. in Physic Engineering, Department of Physics, Physics Engineering, and Optics. Faculty of Science and Engineering, Laval University, Quebec, Canada

2004-2006 M. Sc. in Aerospace Engineering, Department of Mechanical Engineering, Faculty of Science and Engineering, Laval University, Quebec, Canada

2005 6 months Internship at Bombardier Aerospace, Montreal, Canada

2006-2007 Research associate at Laval Robotic Laboratory, Laval University, Canada.

2007-2011 PhD Candidate, Department of Precision and Mycrosystems Engineering, Delft University of Technologies, Delft, The Netherlands

2011- Postdoctoral research, Department of Precision and Mycrosystems Engineering, Delft University of Technologies, Delft, The Netherlands

Publications

P.Lambert, J. Herder, “Self Dual topology of parallel mechanisms with configurable platform”, Computational Kinematics (2013)

T. Hoevenaars, P. Lambert, J. Herder, “Fundamental 3 DOF parallel mechanism with configurable platform”, Computational Kinematics (2013)

P. Lambert, J. Herder, “Mobility Analysis of Non Series-Parallel Mechanisms” Proceedings of the European Conference of Mechanism Science (2012)

T. Van Dam, P. Lambert, J. Herder, “Static Balancing of Translational Parallel Mechanisms” Proceedings of the ASME 2011 International Design Engineering Technical Conferences (2011)

P. Lambert, H. Langen, R. Munnig Schmidt, “A Novel 5 DOF Fully Parallel Robot combining 3T1R Motion and Grasping ” Proceedings of the ASME 2010 International Design Engineering Technical Conferences (2010)

P. Lambert, R. Munnig Schmidt, “A Novel 5 DOF Fully Parallel Robot for Haptic Microassembly ” (2010)

Estevez, P., S. Khan, P. Lambert, M. Porta, I. Polat, C. Scherer, M. Tichem, U. Staufer, H. Langen, and R. Munnig Schmidt. “A Haptic Tele-operated System for Microassembly.” Precision Assembly Technologies and Systems (2010): 13–20.

Martin J.-D. Otis, Simon Perreault, Thien-Ly Nguyen-Dang, Patrice Lambert, Marc Gouttefarde, Denis Laurendeau, and Clément Gosselin. “Determination and Management of Cable Interferences Between Two 6-DOF Foot Platforms in a Cable-Driven Locomotion Interface” IEEE Transactions on Systems, Man, And Cybernetics, VOL. 39, NO. 3, (2009)

P. Lambert, P.Estevez, S. Khan, H. Langen, R. Munnig Schmidt, M. Porta, M. Tichem, U. Staufer, I. Polat, C. Scherer “Progress on Haptic Teleoperation for Micro-assembly,” (2009).

P. Lambert, P.Estevez, S. Khan, H. Langen, R. Munnig Schmidt, M. Porta, M. Tichem, U. Staufer, I. Polat, C. Scherer “ Haptic teleoperation for Micro-assembly ” (2008).

P. Lambert, I. Polat, M van den Berg, S. Liu, H. Langen, C Scherer, “ Haptic Devices for Tele-Microassembly ” (2007).

Patent

Patrice Lambert, “High Speed Parallel Robot With 5 Degrees of Freedom”, PCT phase started on 28th January 2011