Reduced workspace is the main parallel robot disadvantage. It is generally due to the robot configuration, mainly the platform orientation. The present work intends to find the maximum sphere within the orientation workspace, *i.e.*, the singularity-free orientation regions. These regions are related to the platform orientation through Roll-Pitch-Yaw angles. Therefore, a genetic algorithm optimization is used to determine the initial platform orientation corresponding to the highest sphere volume. In this algorithm, the geometrical parameters and the direct and inverse singularities are the optimization constraints. The geometrical constraints are studied using vectorial analysis. The reciprocity property from screw theory is implemented to analyze the direct and inverse kinematics. In the optimization problem, the sphere volume, *i.e.*, the angular displacement of the moving platform around any axis is the objective function to be maximized. Thus, the genetic algorithm individuals explore all feasible regions looking for an optimal solution. In this work, it is used as a methodology to verify the singularity closeness measure associated with direct kinematic. This measure is related to the rate of work done by each leg upon the platform twist. To determine how close is the parallel robot to a direct singularity an index value is proposed, which is calculated using a dynamic software simulation. It is considered that the passive joints reachable regions may be limited by a cone, whereby the cone symmetrical axis is the same than the passive joint axis. For a planar parallel robot 3-$\underline{\text{R}}$RR case the platform may travel *35.6930mm* in the plane *x-y* and reach orientation $\psi = 35.6930^o$ from the sphere origin without falling into singularities while in the parallel robot Stewart-Gough case the platform could reach these orientations $\theta_{x,y,z} \leq 0.646rad$ without falling into singularities.

Adviser: Alexandre Campos Bonilla

Joinville, 2018

YEAR
2018

LUIS EDUARDO GARCIA GONZALEZ| MAXIMAL SINGULARITY-FREE ORIENTATION SUBREGIONS ASSOCIATED TO INITIAL PARALLEL MANIPULATOR CONFIGURATION

UDESC

SANTA CATARINA STATE UNIVERSITY – UDESC
COLLEGE OF TECHNOLOGICAL SCIENCE – CCT
MECHANICAL ENGINEERING GRADUATE PROGRAM – PPGEM

MASTER THESIS

MAXIMAL SINGULARITY-FREE ORIENTATION SUBREGIONS ASSOCIATED TO INITIAL PARALLEL MANIPULATOR CONFIGURATION

LUIS EDUARDO GARCIA GONZALEZ

JOINVILLE, 2018

LUIS EDUARDO GARCIA GONZALEZ

MAXIMAL SINGULARITY-FREE ORIENTATION SUBREGIONS ASSOCIATED TO INITIAL PARALLEL MANIPULATOR CONFIGURATION

Master thesis submitted to the Mechanical Engineering Department at the College of Technological Science of Santa Catarina State University in fulfillment of the partial requirement for the Master's degree in Mechanical Engineering.

Adviser: Dr. Eng. Anibal Alexandre Campos B.

JOINVILLE - SC
2018

GARCIA GONZALEZ, LUIS EDUARDO
MAXIMAL SINGULARITY-FREE ORIENTATION SUBREGIONS ASSOCIATED TO INITIAL PARALLEL MANIPULATOR CONFIGURATION / LUIS EDUARDO GARCIA GONZALEZ. - Joinville , 2018.
113 p.

Orientador: ANIBAL ALEXANDRE CAMPOS BONILLA
Dissertação (Mestrado) - Universidade do Estado de Santa Catarina, Centro de Ciências Tecnológicas, Programa de Pós-Graduação em Engenharia Mecânica, Joinville, 2018.

1. PARALLEL MANIPULATOR. 2. SCREW. I. CAMPOS BONILLA, ANIBAL ALEXANDRE. II. Universidade do Estado de Santa Catarina. Programa de Pós-Graduação. III. Título.

Maximal Singularity-Free Orientation Subregions Associated to Initial Parallel Manipulator Configuration

por

Luis Eduardo Garcia Gonzalez

Esta dissertação foi julgada adequada para obtenção do título de

MESTRE EM ENGENHARIA MECÂNICA

Área de concentração em "Modelagem e Simulação Numérica"
e aprovada em sua forma final pelo

CURSO DE MESTRADO ACADÊMICO EM ENGENHARIA MECÂNICA
DO CENTRO DE CIÊNCIAS TECNOLÓGICAS DA
UNIVERSIDADE DO ESTADO DE SANTA CATARINA.

Banca Examinadora:

Prof. Dr. Anibal Alexandre Campos Bonilla
CCT/UDESC (Orientador/Presidente)

Prof. Dr. Ricardo de Medeiros
CCT/UDESC

video conferencia

Prof. Dr. Roberto Simoni
UFSC

Joinville, SC, 14 de setembro de 2018.

To Evelyn, Luis, Karina, Karen and Yina, my family.

ACKNOWLEDGMENTS

The author would like to express the following acknowledgments.

To GOD.

To Evelyn, Luis, Karina, Karen and Yina, my family for the love and support.

To Colombiaville, LAMEC group, great friends for the studying and desconcentratio moments.

To Lenz, Ricardo and Pablo, professor and great friends, for teaching and guiding.

To Alexandre Campos, professor and advisor, for teaching, guiding and helping the development of this work.

The authors would like to express the following acknowledgments. To CAPES-BRAZIL, TECNOLOGIA ASSISTIVA PGPTA 59/2014-3686/2014, for the financial support.

"Learning without thought is labor lost; thought without learning is perilous"

Confucio

ABSTRACT

GARCIA GONZALEZ, Luis Eduardo, MAXIMAL SINGULARITY-FREE ORIENTATION SUBREGIONS ASSOCIATED TO INITIAL PARALLEL MANIPULATOR CONFIGURATION. 2018. f. Master Dissertation (Master in Mechanical Engineering - Area: Numerical Modeling and Simulation) – Santa Catarina State University. Post-Graduation in Mechanical Engineering Joinville 2018.

Reduced workspace is the main parallel robot disadvantage. It is generally due to the robot configuration, mainly the platform orientation. The present work intends to find the maximum sphere within the orientation workspace, $i.e.$ the singularity-free orientation regions. These regions are related to the platform orientation through Roll-Pitch-Yaw angles. Therefore, a genetic algorithm optimization is used to determine the initial platform orientation corresponding to the highest sphere volume. In this algorithm, the geometrical parameters and the direct and inverse singularities are the optimization constraints. The geometrical constraints are studied using vectorial analysis. The reciprocity property from screw theory is implemented to analyze the direct and inverse kinematics. In the optimization problem, the sphere volume, $i.e.$, the angular displacement of the moving platform around any axis is the objective function to be maximized. Thus, the genetic algorithm individuals explore all feasible regions looking for an optimal solution. In this work, it is used as a methodology to verify the singularity closeness measure associated with direct kinematic. This measure is related to the rate of work done by each leg upon the platform twist. To determine how close is the parallel robot to a direct singularity an index value is proposed, which is calculated using a dynamic software simulation. It is considered that the passive joints reachable regions may be limited by a cone, whereby the cone symmetrical axis is the same than the passive joint axis. For a planar parallel robot $3-\underline{R}RR$ case the platform may travel $35.6930mm$ in the plane $x-y$ and reach orientation $\psi = 35.6930°$ from the sphere origin without falling into singularities while in the parallel robot Stewart-Gough case the platform could reach these orientations $\theta_{x,y,z} \leq 0.646rad$ without falling into singularities.

Key-words: Parallel Manipulator, Singularity, Workspaces, Passives Joints

RESUMO

GARCIA GONZALEZ, Luis Eduardo, MÁXIMA SUB-REGIÕES DE ORIENTAÇÃO LIVRE DE SINGULARIDADE RELACIONADAS À CONFIGURAÇÃO INICIAL DO ROBÔ PARALELO . 2018. f. Dissertação (Mestrado em Engenharia Mecânica - Área: Modelagem e Simulação Numérica) – Universidade do Estado de Santa Catarina. Programa de Pós-Graduação em Engenharia Mecânica Joinville 2018.

O espaço de trabalho reduzido é uma das principais desvantagens dos robôs paralelos, porque devido a configuração do robô, há uma restrição do movimento da plataforma. O objetivo do presente trabalho é encontrar a maior esfera dentro do espaço de trabalho das orientações, ou seja, as regiões livres de singularidades. Essas regiões são associadas á plataforma por meio dos ângulos Roll-Pitch-Yaw. Por conseguinte, um algoritmo genetico de otimização é usado para determinar a orientação inicial da plataforma, relacionada ao maior volume da esfera. Este algoritmo apresenta dois tipos de restrições, geométricas e cinemáticas. A primeira pode ser determinada usando a análise vetorial, enquanto que para a segundo pode ser usada a teoria dos helicoides. No problema de otimização, o volume da esfera, ou seja, o máximo deslocamento angular da plataforma entorno de um eixo, é a função objetivo. Assím, os indivíduos do algoritmo genético exploram todo o espaço em procura da solução ótima. Al|ém disso neste trabalho é utilizado uma metodologia para medir a proximidade da singularidade associada a cinemática direta. Essa medida é relacionada ao trabalho feito por cada perna sobre o heligiro da plataforma. Um índice é proposto para determinar quão proximo está o robô da singularidade direta, o valor deste indice é calculado usando um software de simulação dinâmica. É considerado que a região de alcance da junta passiva poderia ser limitada por um cone, onde o eixo de simetria deste cone e da junta passiva são as mesmas. Para o caso do robô paralelo planar $3-\underline{R}RR$ a plataforma pode viajar $35.6930mm$ no plano $x-y$ e alcançar a orientações de $\psi = 35.6930°$ da origem da esfera sem cair em singularidades, enquanto no robô paralelo Stewart-Gough a plataforma poderia alcançar orientações de $\theta_{x,y,z} \leq 0.646rad$ sem cair em singularidades.

Palavras-chave: Manipulador paralelo, Singularidade, Espaço de Trabalho, Jntas Pasivas

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

According to their structural topology, a parallel robot consists of two platforms (fixed and moving), connected through serial (open-loops) kinematic chains (TSAI, 1999). The fixed platform is called base and the moving platform is called platform. The parallel robot presents advantages in terms of dynamic properties, load carrying capacity, high accuracy and stiffness, which are widely used in the industry (GAO; ZHANG, 2011). However, compared with the serial robot workspace, the parallel robot workspace is reduced. This disadvantage is analyzed aiming to improve the robot orientation capacity using the kinematics and geometrical robot parameters. Each of these parameters is related to workspace by means of screw theory used, to numerically optimize the parallel robot orientation range. The present work intends to maximize the sphere within the orientation workspace.

1.1 Motivation

For parallel robot kinematic analysis, synthesis and application planning, the workspace is an essential property. A general workspace is a six dimensional volumetric space, and its characterization is difficult due to its complicated geometry. Therefore, to replace it by a convex shape, $i.e,$ the sphere could be of high importance in kinematic optimization (POTT, 2018). Different methods have been presented related to workspace with a sphere. In Bayani et al. (2014), a procedure to obtain the maximal area ellipse, and the maximal volume ellipsoid within the feasible workspace of the cable driven parallel robot wrench, using convex optimization is proposed. In this procedure, the workspace boundaries equations are relaxed by means of Weierstrass and Chebyshev approximation theorems. The wrench feasible workspace is the set of postures of the moving platform for which the cables can balance any wrench for a given set of wrenches. To approximate the workspace to a convex geometry (sphere) it is necessary to describe precisely all the singularity free regions.

Recently, researchers have shown an increased interest in the orientation workspace analysis (BOHIGAS et al., 2013a; JIANG; GOSSELIN, 2009; HUANG et al., 2012; MERLET, 2006; KARIMI et al., 2014). Bohigas et al. (2013a) used the Stewart platform movement to describe the reachable workspace from a known initial configuration. This description provides a whole motion range picture of the robot. On the other hand, for Jiang and Gosselin (2009) the orientation workspace at a prescribed position can be defined by up to 12 workspace surfaces. Therefore, to obtain the maximal singularity-free orientation workspace at a prescribed position of the Stewart–Gough platform, an algorithm to determine these 12 workspace surfaces is developed. In this case, the Roll–Pitch–Yaw angles are used to describe the platform orientation. A second algorithm is develop to compare the maximal singularity-free sphere with the maximal orientation workspace, considering any type of Stewart–Gough platform configuration. Bohigas et al. (2012) proposed a method for identifying the workspace boundary on general robot configuration by mean of a technique named branch-and-prune. In this case, a set of output is isolated and classified according to movement restrictions. Thus, a workspace map is obtained, recognizing the workspace limit. Such method may be applied in serial and parallel robots, as well as for planar or spatial robot.

The orientation workspace is limited due to kinematic and geometrical constraints. The latter constraints are related to robot legs configuration and the platform location through the screw theory. To implement the screw theory. First, the platform motion is described by mean of the orientation matrices which relate the platform orientation with fixed coordinates system by the use of three angles. These angles correspond to three or more successive rotations about the base frame axes (BONEV; RYU, 2001).

Monsarrat and Gosselin (2003) optimized the parallel robot design through the workspace analysis using the tilt and torsion angles to describe the platform orientation. For that purpose, an optimization procedure to obtain the higher workspace volume is implemented for initial platform location. Such procedure consists in two parts: initially, the workspace volume is determined considering the platform initial location to the coordinates subset x, y, and torsion angle φ. Subsequently, the same analysis is done considering the remaining coordinates subset (z, θ,ϕ).

Sun et al. (2012) used a mathematical commercial software, to get the reachable workspace for a 3-DOF PUS$\&$S parallel robot, applied in the large fuselage or wing assembly of aircraft manufacturing. To obtain a proper design based on the workspace representation, the geometrical restriction and kinematic singularities are considered using screw theory and Tilt-and-Torsion angle method. The Tilt-and-Torsion angle method is implemented to describe the platform orientation and the screw theory to explain the actuated joint behavior related to the platform motion.

According to Huang et al. (2012), for each initial platform location exist an orientation workspace. Thus, the robot performance may be improved by defining an optimal initial platform location. To demonstrate such a sentence, the optimal initial platform location for a cubic robot is identified. For that purpose, an algorithm that determines the orientation workspace through the set angles variation is developed using two rotation matrices. The first matrix describes the platform orientation with respect to the base. While the second describes the platform orientation with respect to their initial orientation through a non-conventional matrix based on the modified Euler Angles.

In a parallel robot, the workspace is limited by singular configurations, which may be inverse, direct or combined. Inverse singularity occurs when the robot loses one or more degrees of freedom. Direct singularity occurs when the platform gains one or more degrees of freedom, identify the direct singularity generally is a complex task (TSAI, 1999). Combined singularity appear when the robot falls in inverse and direct singularity. Several methods are proposed to analyze the robot singularities. St-Onge and Gosselin (2000) use the linear decomposition to approach the architecture parameters effect on the nature of the singularity loci. To do this, an algorithm based on the analytical expression for the Jacobian matrix determinant is implemented. the later is possible by two different approaches: linear decomposition and cofactor expansion. The first relates to the architecture parameters with the robot singularities loci, while the second approach reduces significantly the computational complexity of the determinant. On the other hand, Kanaan et al. (2009) introduced a method to analyze singularities geometrically using Grassmann–Cayley algebra (GCA). Where the actuation forces and constraint moments are applied to the platform through their legs therefore, a parallel robot is analyzed, relating legs configuration to the geometric conditions, these conditions are associated with the six Plücker vectors constituting the inverse Jacobian matrix rows. Accordingly, the singularity conditions are obtained in vector form. Besides that Ben-Horin and Shoham (2006) used the Grassmann–Cayley algebra analysis to obtain the geometric conditions of singularities leaning on screw theory Ḟirst, the screw axes for the actuator in each leg chain are determined. Then the Grassmann–Cayley algebra and the associated superbracket decomposition are used. These methods are implemented for determinate the Jacobian matrix condition, in which the screw axes for each leg are contained. Enabling the geometrical interpretation of the singularity condition easily.

Coste and Moussa (2015) analyzed the singularity locus of a Stewart–Gough platform through a surface over the field of rational functions on the group of rotations. In the generic biplanar case, the parallel planes family cut the surface in a linear pencil of conics, and the rotational parametrization is uniform for all generic orientations. They are determined from the geometric properties of this surface.

The optimization algorithms implemented in this problem are mainly based on stochastic concepts due to the parallel robot analysis complexity (KARIMI et al., 2014). These algorithms follow certain characteristics and behavior of biological, molecular, a swarm of insects, and neurobiological systems. The main advantage of this algorithms is that they not require derivatives (RAO; RAO, 2009).

Karimi et al. (2014) studied the parallel mechanism workspace implementing several algorithms based on the convex optimization. These algorithms intend to obtain the maximum ellipsoid or sphere volume into Stewart–Gough platform singularity-free subregions. To find the maximum volume ellipsoid, an iterative procedure, referred to as Improved Lower Bound Semi-definite Programming is proposed. Additionally, an approach based on the sum of squares method is proposed to solve the singularity-free subregions problem for a general Stewart–Gough platform. it is considered the actuator limits and any geometrical parameters. These parameters are used to a polynomial optimization problem.

Stan et al. (2009) analyzed a 2-DOF medical parallel robot kinematic aim to obtain the maximal workspace area by mean of genetic algorithms (GAs). To find the optimal solution, the optimization algorithm explores all feasible parallel robot configurations. In this optimization problem, the kinematics singularities are the constraints, while the geometric parameters are the input data. To calculate the optimal singularity-free cylindrical workspace and to determine continuous singularity-free zones. Abbasnejad et al. (2012) developed an algorithm that detects the optimal singularity-free cylindrical workspace for any prescribed orientation ranging from an initial orientation angle in the platform. In this case, the algorithm is implemented in a 3-RPR planar robot, using robot structural parameters as constraints. The Particle Swarm Optimization (PSO) algorithm is used to determinate the closest point on the singularity surface to the axis of the cylinder.

1.2 Objective and Scope

The main objective of this work is to develop an optimization algorithm to found the initial platform orientation related to maximum volume sphere within the singularity-free subregions. Considering that, the platform motion is described through RPY angles. Therefore, the singularity-free subregions are limited to the platform orientations

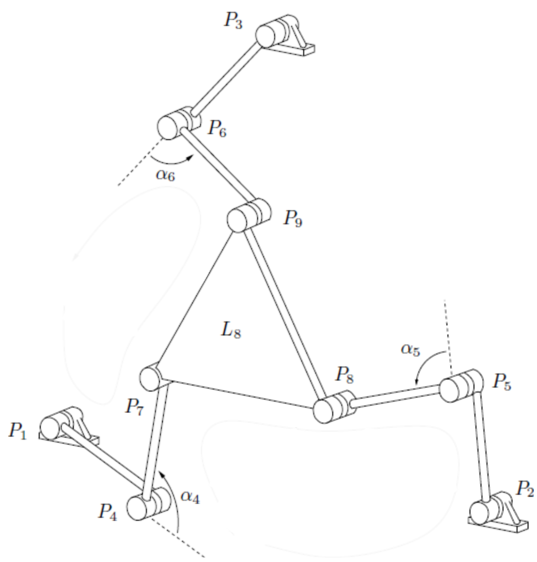
$i.e.$ orientation workspace. The platform mobility is restricted by the kinematics and geometrical constraints, which are related with the parallel robot configuration considering the platform motion using the screw theory and vectorial analysis. To identify the optimal sphere size, the platform mobility constraints should be computed. Owing to the orientation workspace volume is limited by these constraints. The geometrical constraints are measure through the vectorial analysis, while the kinematics singularities are determined by the mean of the Jacobian arrays. In this case, the Jacobian matrix rows are formed by joints twist and reciprocity wrench from screw theory analysis. Thus, to obtain the maximal sphere volume a genetic algorithm is developed. In this algorithm, the kinematical singularities and geometrical restrictions ($e.g,$ limb longitude) are the optimization constraints. Whereas, the maximal sphere volume size, $i.e,$ the largest orientation workspace is the objective function. The specific objectives of this work are described as:

Geometrical Constraints: establishe a mathematical formulation to describe vectorially each constraint relate to the parallel robot geometric parameters.

Kinematic Singularities: measure how close the parallel robot is of kinematic singularities. Specifically, with respect to the direct singularity.

The present work intends to study the planar parallel robot 3-<u>R</u>RR and the Stewart-Gough (S-G) platform. The planar parallel robot 3-<u>R</u>RR consists of a moving platform linked to the fixed base by means of three legs, where each leg is a three-revolute chain (see Fig.1.1). The first rotational joint, attached to the base, is the actuated one in every kinematic chain.

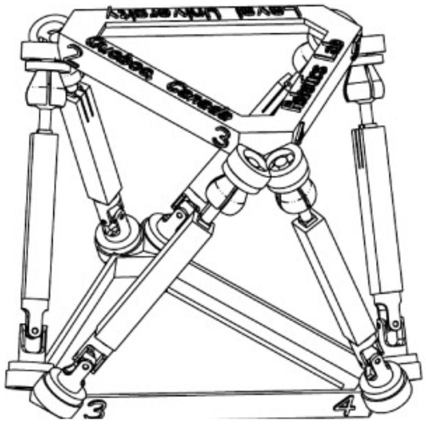
Figure 1.1 – Planar parallel robot 3-<u>R</u>RR .



Source:(BOHIGAS et al., 2013a)

The S-G platform is a spatial parallel robot with 6-DOF (see Fig 1.2). The moving platform, from now on called platform, and the fixed base are connected by six extensible (prismatic joint) and identical legs. These connectors consist of one universal and one spherical joint. Therefore each leg is a U$\underline{\text{P}}$S (universal, prismatic and spherical) kinematic chain, where the underline indicates the actuated joint. It should be noted that for the S-G platform the prismatic joint is actuated.

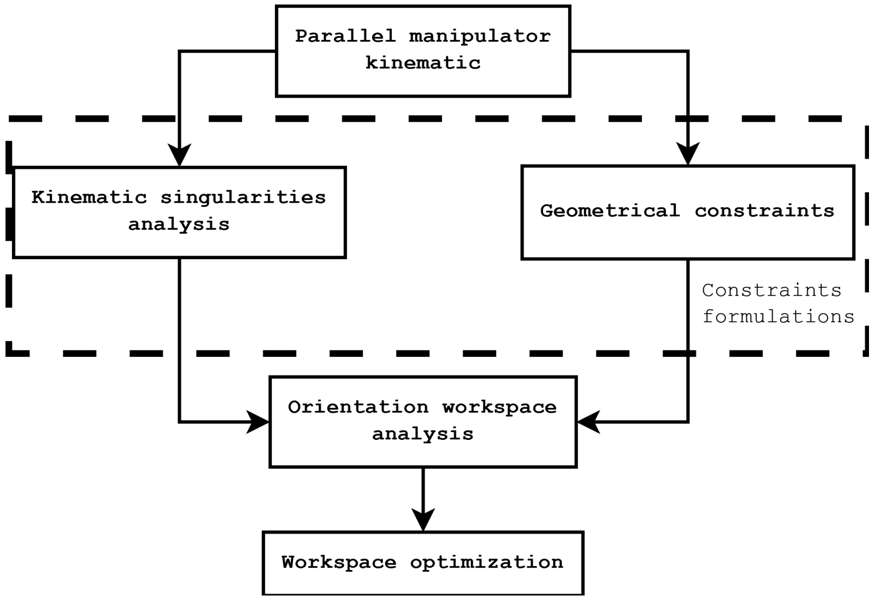
Figure 1.2 – Stewart-Gough platform.



Source:(GAO et al., 2010)

The parallel robots have been widely used as motion simulators, medical robots, industrial robots, nano-manipulators, and micro-manipulators, to name only a few. For the parallel robot analyze an parallel robot kinematic model is developed using multibody dynamic software called Msc ADAMS (Automatic Dynamic Analysis of Mechanical Systems), which it is a widely used tool for the mechanical systems analysis. To verify the parallel robot direct kinematic some dynamic simulations are made in ADAMS. These simulations consist in to rotate and movement the platform about the z axis and in the plane $x - y$ respectively, in a lapse of time t for the 3-$\underline{\text{R}}$RR study, while for the S-G case the simulations consistin in to rotate the platform about the $x - y - z$ axes in a lapse of time t. For both case the actuated joints force are measured by the dynamic software.

Figure 1.3 – Outline of the proposed methodology.



Source:Author

The Fig. 1.3 presents the methodology implemented in this work. Initially, in the parallel robot kinematic analysis is described the robot motion using screw theory. This theory allows associate the platform and joints velocities in a simplest form through the Jacobian matrices J_q and J_x.Consequently, the constraints formulation is done, which may be divided in two parts: the first part is aiming to identify the three singularities types. It allows detect if the robot gains or loses degree of freedom, and the second parts identify the constraints associated to the parallel robot physical parameters. Thus, the workspace boundaries may be studied through parallel robot constraints. The next step consist in described the workspace using a geometric methods, discretization method, and numerical method. Finally, the workspace optimization is done by mean of genetic algorithm, where the sphere size is the fitness function.

1.3 Outline of the Dissertation

The dissertation is divided into three parts. Part 1 introduces the concepts related to the parallel robot mobility, their constraints and how it could be analyzed. Part 2 presents the methodology implemented in the proposed problem and the obtained results. Finally, part 3 provides the conclusions, the future works, and the literature review.

- Chapter 1 : Explains the motivation of this research and provides its main objectives and scope.

- Chapter 2 : Shows the essential mathematical tools. Beginning by the screw, the screw vector and the screw algebra. Later the kinematic analysis is introduced, which explains how the rigid body motion is related to screw movement. Subsequently, the relation between the force acting in a body and the screw is presented in the static section. Finally, the reciprocal screw and the orientation matrix are exposed.

- Chapter 3 : Introduces the principals parallel robots concepts and configurations. Consequently, the screw theory applied to parallel robot kinematics is submitted. This theory allows associating the platform and joints velocities in a simple form using the Jacobian matrices J_x and J_q.

- Chapter 4 : Presents the constraint formulation related to the parallel robot in the study. Initially, the kinematics constraints are explained and later the geometrical constraints. In the first part, a methodology called Closeness Measures to verify the singularity related to Direct kinematic is proposed. Whereas, in the second part the constraints associated with the physical robot parameter as the joints range is exhibited.

- Chapter 5 : Introduces different methodologies to decribe the orientation workspace and introduces the genetics algorithm main concepts used to determine the optimal solution.

- Chapter 6 : Proposes an algorithm to locate the initial platform configuration bounded by the higher sphere within the workspace. Its performance is demonstrated with the planar parallel robot 3-$\underline{\text{R}}$RR.

- Chapter 7 : Proposes an algorithm to locate the initial platform orientation bounded by the higher sphere within the orientation workspace. Its performance is demonstrated with the Stewart-Gough (S-G) parallel robot.

- Chapter 8 : Summarizes the dissertation contributions and outlines to possible future researches.

Chapter 2

Mathematical Tools

2.1 Screw

The theory of screws has been used to analyze finite and instantaneous motions of rigid bodies over the past few centuries (BANDYOPADHYAY; GHOSAL, 2009). Initially, Chasles (1830) proposed the concept of the twist motion of a rigid body, which was further developed by Poinsot (1848). Plücker proposed a screw expression (HUANG et al., 2012). However, Ball (1998) established formally the theory of screws and applied it to the analysis of rigid-body motions of multiple degrees-of-freedom (BANDYOPADHYAY; GHOSAL, 2009).

This section shows the basic screw theory concepts. Beginning by the line equation reaching to the Plücker line coordinates analysis in three-dimensional space, later the screw vector and the algebra screw are introduced. Finally, the reciprocal screw is studied.

2.1.1 Line Equation

Two points $A(x_1, y_1, z_1)$ and $B(x_2, y_2, z_2)$ in the three-dimensional space may form a three-dimensional line. Therefore, a vector **S** embedded in the line formed by this two points may be described as (DAVIDSON; HUNT, 2004; HUANG et al., 2012)

$$\mathbf{S} = (x_2 - x_1)\,\hat{\mathbf{i}} + (y_2 - y_1)\,\hat{\mathbf{j}} + (z_2 - z_1)\,\hat{\mathbf{k}}, \tag{2.1.1}$$

where $\hat{\mathbf{i}}, \hat{\mathbf{j}}$ and $\hat{\mathbf{k}}$ are unit vectors corresponding to each coordinate axis. Let

$$\begin{aligned} L &= (x_2 - x_1)\,, \\ M &= (y_2 - y_1)\,, \\ N &= (z_2 - z_1)\,. \end{aligned} \tag{2.1.2}$$

Then **S** may be rewritten as

$$\mathbf{S} = L\hat{\mathbf{i}} + M\hat{\mathbf{j}} + N\hat{\mathbf{k}}, \tag{2.1.3}$$

L, M and N are denominated the directions ratios (HUANG et al., 2012), let

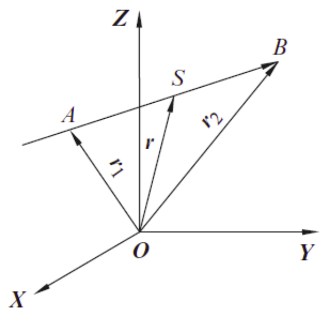
$$\begin{aligned} l &= L/\left|\mathbf{S}\right|, \\ m &= M/\left|\mathbf{S}\right|, \\ n &= N/\left|\mathbf{S}\right|. \end{aligned} \tag{2.1.4}$$

Where $|\mathbf{S}|$ is the vector norm two,

$$|\mathbf{S}| = \sqrt{L^2 + M^2 + N^2}. \tag{2.1.5}$$

In Fig 2.1 the line formed by the points A, B may be described by means of its direction and a point on it.

Figure 2.1 – The line description



Source:(HUANG et al., 2012)

Then, the line equation is written as

$$\mathbf{r} - \mathbf{r}_1 = t\mathbf{S}, \tag{2.1.6}$$

then,

$$(\mathbf{r} - \mathbf{r}_1) \times \mathbf{S} = 0, \tag{2.1.7}$$

the Eq. (2.1.7) can also be expressed as

$$\mathbf{r} \times \mathbf{S} = \mathbf{S}_0, \tag{2.1.8}$$

where $\mathbf{S}_0$ is the geometric moment of the line about the origin O, which is described as (CAMPOS, 2004),

$$\mathbf{S}_0 = \mathbf{r}_1 \times \mathbf{S}. \tag{2.1.9}$$

Expanding Eq 2.1.9 leads to

$$\mathbf{S}_0 = \begin{bmatrix} \hat{\mathbf{i}} & \hat{\mathbf{j}} & \hat{\mathbf{k}} \\ x_1 & y_1 & z_1 \\ L & M & N \end{bmatrix}. \tag{2.1.10}$$

Then the vector $\mathbf{S}_0$ may be expressed in the form,

$$\mathbf{S}_0 = P\hat{\mathbf{i}} + Q\hat{\mathbf{j}} + R\hat{\mathbf{k}}, \tag{2.1.11}$$

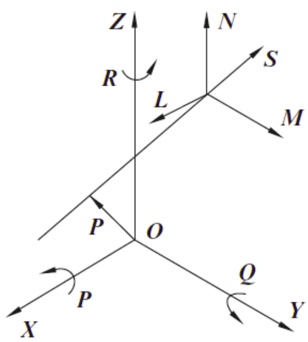
where,

$$\begin{aligned} P &= y_1 N - z_1 M, \\ Q &= x_1 N - z_1 L, \\ R &= x_1 M - y_1 L. \end{aligned} \tag{2.1.12}$$

The Plücker coordinates are really a defined vector by means of six coordinates *(L, M, N; P, Q, R)*, how is shown in Fig. 2.2, this vector satisfies the orthogonality condition (HUANG et al., 2012).

$$\mathbf{S} \cdot \mathbf{S}_0 = 0. \tag{2.1.13}$$

Figure 2.2 – Plücker Coordinates of a line



Source: (HUANG et al., 2012)

The coordinates (L, M, N) consists of the direction ratios of the line and (P, Q, R) are the x, y and z components of the line moment about the origin (HUANG et al., 2012).

2.1.2 The Screw

When the two vectors **S** and $\mathbf{S}_0$ do not satisfy the orthogonality condition $\mathbf{S}\cdot\mathbf{S}^0 \neq 0$, the dual vector is called a screw, and it is denoted by

$$\$ = \begin{bmatrix} \mathbf{S} \\ \mathbf{S}_0 \end{bmatrix}, \tag{2.1.14}$$

the **S** is a constant vector, and $\mathbf{S}_0$ is origin-dependent (HUANG et al., 2012), $e.g.$ if the origin is shifted from point O to point A (see Fig. 2.3), then the moment of **S** about r_A can be written as

$$\mathbf{S}_A = \mathbf{r}_A \times \mathbf{S}. \tag{2.1.15}$$

Where $\mathbf{r}_A$ is a vector that may be described as $\mathbf{r}_A = \mathbf{r} + \overline{\mathbf{OA}}$, thus the Eq.(2.1.15) is expressed as

$$\mathbf{S}_A = [\mathbf{r} + \overline{\mathbf{OA}}] \times \mathbf{S}. \tag{2.1.16}$$

Multiplying both sides this equation by **S**,

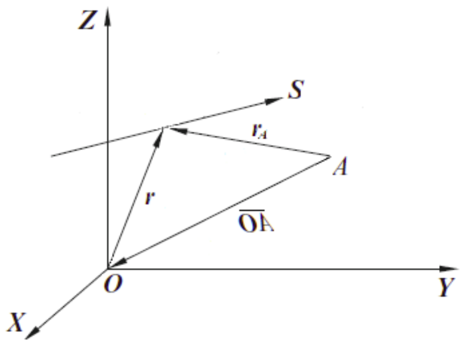
$$\mathbf{S}_A \cdot \mathbf{S} = \mathbf{S}^0 \cdot \mathbf{S}. \tag{2.1.17}$$

Thus, the Eq.(2.1.17) shown that $\mathbf{S}^0 \cdot \mathbf{S}$ is not origin-dependent. Then, if $\mathbf{S} \neq 0$, obtain an origin-independent variable is possible, it may be written as,

$$h = \frac{\mathbf{S}^0 \cdot \mathbf{S}}{\mathbf{S} \cdot \mathbf{S}}, \tag{2.1.18}$$

where h is called the pitch of a screw.

Figure 2.3 – Vector and Screw

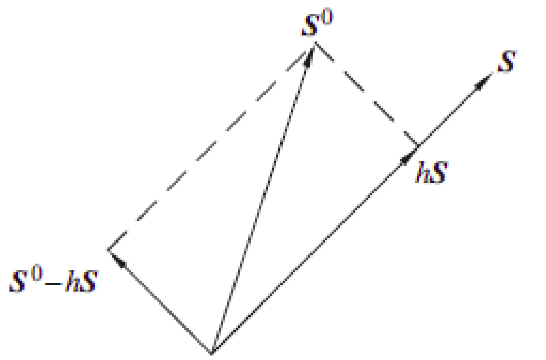


Source: Author's production.

The screw may be described in the origin-independent form. To this end, $\mathbf{S}^0$ is decomposed in two parts, which are parallel and perpendicular to **S**, as shown in Fig. 2.4. Where $\mathbf{S}^0 - h\mathbf{S}$ is perpendicular to **S** and $\mathbf{S}^0 - h\mathbf{S} = \mathbf{S}_0$ (HUANG et al., 2012), this may be described as

$$\mathbf{r} \times \mathbf{S} = \mathbf{S}^0 - h\mathbf{S}. \tag{2.1.19}$$

Figure 2.4 – Screw Axis



Source: Huang 2012

Based on Eq. (2.1.19) is possible rewritten $\mathbf{S}^0$ as

$$\mathbf{S}^0 = \mathbf{r} \times \mathbf{S} + h\mathbf{S}. \tag{2.1.20}$$

Thus, the screw is expressed as

$$\$ = \begin{bmatrix} \mathbf{S} \\ \mathbf{r} \times \mathbf{S} + h\mathbf{S} \end{bmatrix}. \tag{2.1.21}$$

2.1.3 Screw Algebra

Screw Sum Two screw sum $\boldsymbol{\$}_1 = [\mathbf{S}_1; \mathbf{S}_1^0]$ and $\boldsymbol{\$}_2 = [\mathbf{S}_2; \mathbf{S}_2^0]$ is defined as

$$\boldsymbol{\$}_1 + \boldsymbol{\$}_2 = [\mathbf{S}_1 + \mathbf{S}_2; \mathbf{S}_1^0 + \mathbf{S}_2^0]. \tag{2.1.22}$$

Product of a Scalar and a Screw The product of a scalar λ and a screw product is defined as

$$\lambda\$ = [\lambda\mathbf{S}; \lambda\mathbf{S}^0]. \tag{2.1.23}$$

2.1.4 Reciprocal Screw Product

Consider two screws,

$$\begin{aligned} \$_1 &= [\mathbf{S}_1; \mathbf{S}_1^0], \\ \$_2 &= [\mathbf{S}_2; \mathbf{S}_2^0]. \end{aligned} \tag{2.1.24}$$

The reciprocal is defined as follows,

$$\$_1 \circ \$_2 = \mathbf{S}_1 \cdot \mathbf{S}_2^0 + \mathbf{S}_2 \cdot \mathbf{S}_1^0. \qquad (2.1.25)$$

Where the symbol $\circ$ denoted the reciprocal product. The reciprocal of two screws is not origin-dependent (HUANG et al., 2012).

2.2 Kinematic Analysis

The general rigid body motion can be represented as a motion translational, rotational or a combination of these. This motions may be expressed by mean of a rotation and translation on a single axis labeled screw motion shown in Fig. 2.5. Where, for the rigid body motion the screw pitch is established as (CAMPOS, 2004)

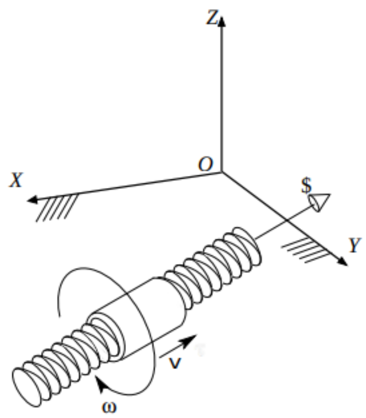
$$h = \frac{\mathbf{v}}{\boldsymbol{\omega}}. \qquad (2.2.26)$$

Where $\mathbf{v}$ and $\boldsymbol{\omega}$ are the forward speed and the angular velocity on the body. Thus, the rigid body motion is described by the twist, this is composed by the screw given in Eq. (2.1.21) and the intensity q. Then, the twist may be expressed

$$\$ = \dot{q}[\mathbf{S}; \mathbf{r} \times \mathbf{S} + h\mathbf{S}]. \qquad (2.2.27)$$

Where $\dot{q} = \omega$ for a rotative motion, and $\dot{q} = v$ for a translation motion.

Figure 2.5 – Screw Motion: Rotational and Translational Motion on a Single Axis



Source: (CAMPOS, 2004)

2.2.1 Instantaneous Translation

For a pure translation motion, the pitch $h = \infty$ due to the angular velocity absence $(\boldsymbol{\omega} = 0)$ (TSAI, 1999). Thus, this motion is described by a prismatic joint with a velocity **v**. For convenience, a vector **S** is drawn through the joint center line (HUANG et al., 2012). Shown Fig. 2.6. From Eq (2.1.21), this motion may be expressed

$$\mathbf{v} = v\$ = v[0; \mathbf{S}], \tag{2.2.28}$$

where $\$$ is given by the pükler coordinates$[0; \mathbf{S}]$, and $v = \|\mathbf{v}\|$ is called a linear velocity intensity.

2.2.2 Instantaneous Rotation

For a pure rotation motion, the pitch $h = 0$ due to angular velocity absence $\mathbf{v} = 0$ (TSAI, 1999). Thus, this motion is described by a rotative joint with an angular velocity $\boldsymbol{\omega}$. For convenience, a vector **S** is drawn through the joint centerline (HUANG et al., 2012). Shown Fig. 2.6. From Eq. (2.1.21) this motion may be described as

$$\boldsymbol{\omega} = \omega\$ = \omega[\mathbf{S}; \mathbf{r} \times \mathbf{S}]. \tag{2.2.29}$$

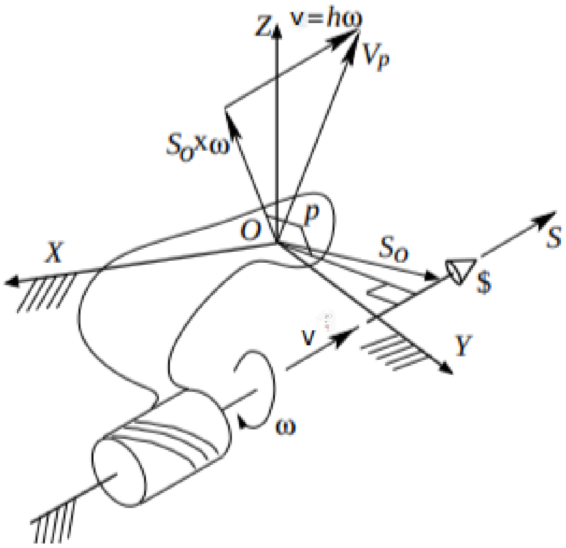
where $\$$ is given by the pükler coordinates $[\mathbf{S}; \mathbf{r} \times \mathbf{S}]$ and $\omega = \|\boldsymbol{\omega}\|$ is called an angular velocity intensity. The term $(\mathbf{r}\times\mathbf{S})$ in the Eq. (2.2.28) is the velocity of a point coincident with the origin that may be written as (TSAI, 1999)

$$\omega\mathbf{r} \times \mathbf{S} = \mathbf{r} \times \boldsymbol{\omega} = \mathbf{v}_0. \tag{2.2.30}$$

Then the Eq. (2.2.28) may be rewritten as

$$\boldsymbol{\omega} = \omega\$ = [\boldsymbol{\omega}; \mathbf{v}_0] \tag{2.2.31}$$

Figure 2.6 – Screw Motion: Rotational and Translational Motion on a Single Axis



Source: (CAMPOS, 2004)

2.3 Statics

The fact that a set of forces and couples acting on the rigid body may be reduced to a resultant force and couple is widely known (see Fig. 2.7). This force may be represented as $\mathbf{f} = f\mathbf{S}$. Where $\mathbf{S}$ is the force direction and f is the force intensity. Thus, the couple due to this resultant force on an instantaneous point in the origin may be described as $f(\mathbf{r} \times \mathbf{S}_r)$ (TSAI, 1999). Therefore, the resultant force acting on a rigid body may be described as

$$\$_r = f[\mathbf{S}_r; \mathbf{r} \times \mathbf{S}_r], \tag{2.3.32}$$

where, for the statics analysis, $\$_r$ is called the wrench. The statics and instantaneous kinematics analysis are analogous, $e.g.$ the Eq. (2.3.32) is related to the rotation motion twist (CAMPOS, 2004). The same way, for Zhao et al. (2009), the couple is described as

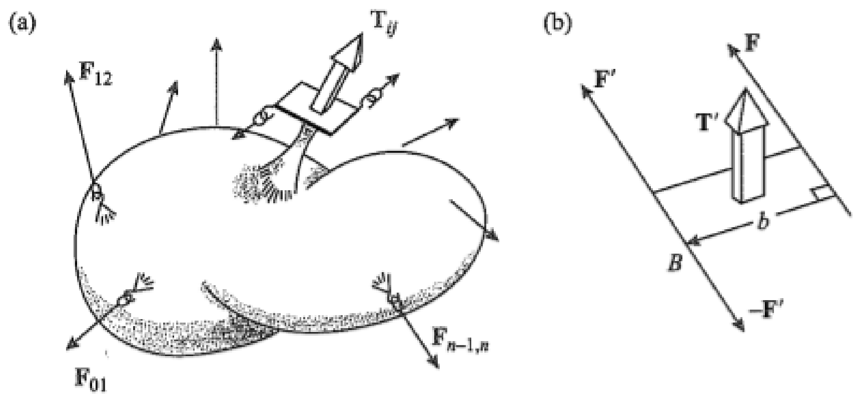
$$\$_r = c[0; \mathbf{S}_r]. \tag{2.3.33}$$

The analogy to describe the couple is similar to the translational motion. The pitch for the static analysis is given by

$$h = c/f. \tag{2.3.34}$$

Where $h = 0$ for a pure force, and $h = \infty$ for a pure couple (DAVIDSON; HUNT, 2004).

Figure 2.7 – (a) Forces and Couples Acting on a Rigid Body (b)Resulting Couple Due to a Binary



Source: (DAVIDSON; HUNT, 2004)

2.4 Reciprocal Screw

The reciprocal screw product is shown in Eq 2.1.25 . Physically is know as the instantaneous work of the force on the rigid body motion (HUANG et al., 2012). This product is written as

$$\delta W = \$ \circ \$_r = \mathbf{f} \cdot \mathbf{v}_0 + \boldsymbol{\omega} \cdot \mathbf{c}_0. \tag{2.4.35}$$

Where the $\$$ is the twist and $\$_r$ is the wrench, if the wrench does not perform work while the rigid body is in motion due to infinitesimal twist $(i.e, \quad \delta W = 0)$, the two screw are said reciprocal screw (TSAI, 1999).

2.5 Orientation Matrix

The matrix orientation columns are mutually orthogonal and their magnitude is one (1), imagine R as a matrix whit three columns , where each column is a unitary vector of any rotate frame analyze from a reference system. In which these unitary vectors are perpendicular to each other. Then, six constraints are defined as (CRAIG, 2012).

$$\begin{aligned} |\hat{\mathbf{X}}| = 1, \quad \hat{\mathbf{X}} \cdot \hat{\mathbf{Y}} = 0; \\ |\hat{\mathbf{Y}}| = 1, \quad \hat{\mathbf{Y}} \cdot \hat{\mathbf{Z}} = 0; \\ |\hat{\mathbf{Z}}| = 1, \quad \hat{\mathbf{Z}} \cdot \hat{\mathbf{X}} = 0. \end{aligned} \tag{2.5.36}$$

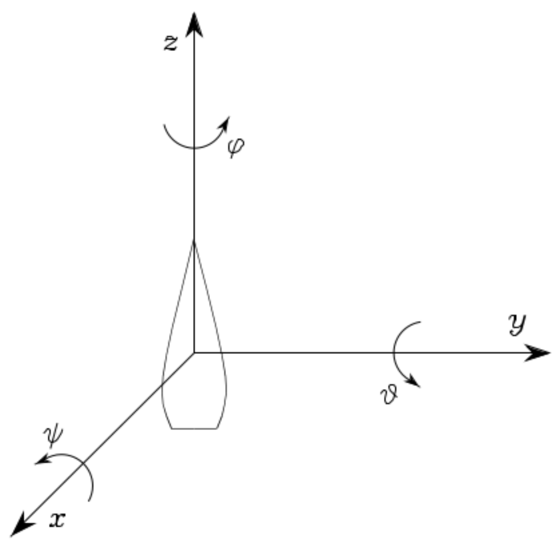
In other words, the orientation matrix is an operator that maps out a rotated system to a fixed system. Considering this, different methods have been proposed to describe the rigid body orientation in a reference system, $e.g.$ RPY Angles.

2.5.1 RPY Angles

Another set of Euler Angles are the ZYX angles (see Fig 2.8), also named Roll-Pitch-Yaw, which are originated from an orientation representation in the aero (aircraft) field. In this case, these angles represent rotations with respect to a fixed frame (SICILIANO et al., 2010). In this method the platform changes from its initial orientation $o - xyz$ to the final orientation $o - x\prime y\prime z\prime$ by mean of following elementary rotations compositions:

- Rotate the reference frame by the angle φ about axis z $(roll)$; this rotation is described by the matrix $R_Z(\varphi)$ which is formally defined in Eq. (2.5.37).
- Rotate the current frame by the angle ϑ about axis y $(pitch)$; this rotation is described by the matrix $R_y(\vartheta)$ which is formally defined in Eq. (2.5.37).
- Rotate the current frame by the angle ψ about axis x (yaw); this rotation is described by the matrix $R_x(\psi)$ which is again formally defined in Eq. (2.5.38).

Figure 2.8 – RPY Representation



Source: (SICILIANO et al., 2010)

where R_Z,R_y and R_x are orientation matrix relate to the rotations in the axis $z, y,$ and x. These orientation matrixes are described as

$$R_y(\vartheta) = \begin{vmatrix} cos(\vartheta) & 0 & sin(\vartheta) \\ 0 & 1 & 0 \\ -sin(\vartheta) & 0 & cos(\vartheta) \end{vmatrix}, \quad R_z(\varphi) = \begin{vmatrix} cos(\varphi) & -sin(\varphi) & 0 \\ sin(\varphi) & cos(\varphi) & 0 \\ 0 & 0 & 1 \end{vmatrix}, \tag{2.5.37}$$

and,

$$R_x(\psi) = \begin{vmatrix} 1 & 0 & 0 \\ 0 & cos(\psi) & -sin(\psi) \\ 0 & sin(\psi) & cos(\psi) \end{vmatrix}. \tag{2.5.38}$$

The end orientation may be described by mean of computation via post-multiplication of each matrix of elementary rotation.

$$R(\varphi, \vartheta, \psi) = R_z(\varphi)R_y(\vartheta)R_x(\psi). \tag{2.5.39}$$

Then,

$$R(\varphi, \vartheta, \psi) = \begin{vmatrix} c\varphi c\vartheta & c\varphi s\vartheta s\psi - s\varphi c\psi & c\varphi s\vartheta c\psi + s\varphi s\psi \\ s\varphi c\vartheta & s\varphi s\vartheta s\psi + c\varphi c\psi & s\varphi s\vartheta c\psi - c\varphi s\psi \\ -s\vartheta & c\vartheta s\psi & c\vartheta c\psi \end{vmatrix}, \tag{2.5.40}$$

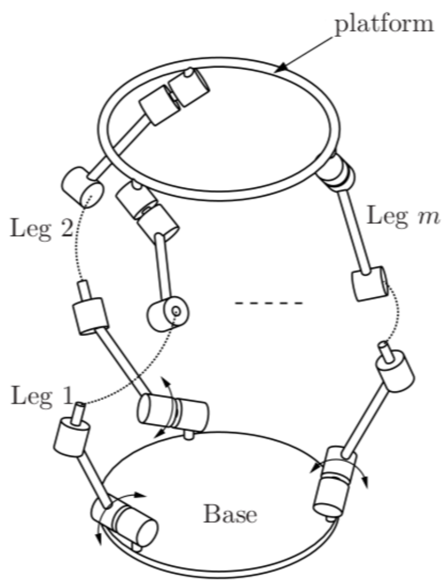
where $c\varphi = cos(\varphi)$, and $s\varphi = sin(\varphi)$.

Chapter 3

Parallel Robot

A parallel robot (PRs) is composed of a platform and a base connected by means of at least two serial kinematic chains (KCs) shown in Fig. 3.1. These serial KCs are called legs (or limbs), each of those limbs contains at least one simple actuator. Thus, these robots are considered multi-degree-of-freedom (multi-Dof) mechanism (KONG; GOSSELIN, 2007). Over the last two decades, parallel robots (PRs) evolved from rather marginal machines to widely used mechanical architectures (KONG; GOSSELIN, 2007). Their higher payload to weight ratio, accuracy, and stiffness allowing their applications in several fields (ANGELES, 2013). Current applications of PRs cover motion simulators, industrial robots, nano-robots, and micro-robots, among others. However, their principal disadvantage is the limited robot workspaces (ABBASNEJAD et al., 2012).

Figure 3.1 – General Parallel robot Structure

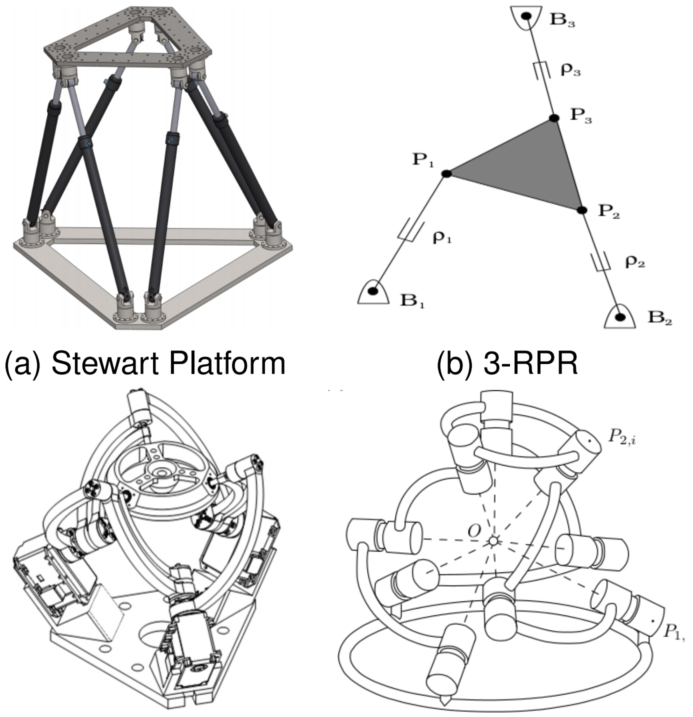


Source: (KONG; GOSSELIN, 2007)

3.1 Classification

According to the parallel robot motion, these could be classified as planar, spherical and spatial. The parallel robot, with planar motion (see Fig. 3.2b), may be described by three motions. Two translational and one rotational. Where, the later is produced around an axis perpendicular to the translational plane, if this one is content by the plane $x - y$, then the rotation axis is parallel to z-axis as shown (TSAI, 1999). In the spherical parallel robot configuration (see Fig. 3.2c-d) the axes joints intersect in a common point, termed to as rotation center. Thus the platform motion is confined on the surface of a sphere centered in this point (SHINTEMIROV et al., 2016). The spatial robot (see Fig. 3.2a) are widely implemented in the industry due to they own high mobility. Originally these robots were designed to have 6-Dof proving to be versatile in complex tasks. While that for less-complexity tasks they turned out to be relatively expensive and redundant. Owing to some applications only a subset of the 6-Dof sufficed (MERLET, 2006). On the other hand, it is possibly classified the PRs by mean of their structural. When the structural limbs configuration is identical, the actuate joints location is the same in all limbs and lastly, if the joints and the robot degree of freedom number are equal. The PRs is considered symmetrical otherwise is considered asymmetrical (TSAI, 1999).

Figure 3.2 – Different Parallel robot Structure
.



(a) Stewart Platform (b) 3-RPR

(c) Agile Eye (d) Spherical PM

Source: (CHRISTENSEN, 2014)(a), (LI, 2005)(b), (KONG; GOSSELIN, 2007)(c-d).

Based on Tsai (1999)the parallel robot can be called symmetric when satisfy the following three rules:

- The joints numbers and its configuration are the same on all the robot legs.
- The number and actuated joints location are the same in each robot leg.
- Parallel manipulator Dof is equal to limbs number.

The other form, it is namely asymmetric.

3.2 Screw Theory

It is possible to express the rigid body infinitesimal displacement as a translation and a rotation about a unique axis, which is called screw displacement (MURRAY et al., 1994). The screw theory is a mathematical tool commonly implemented in the parallel robot analysis. This theory may be applied to indicate the position and orientation of a spatial body, which it may conveniently be represented by two three-dimensional vectors as (HUANG et al., 2012).

$$\hat{\$} = \begin{bmatrix} s \\ s_0 \times s + hs \end{bmatrix}, \tag{3.2.1}$$

where the unit vector s is along the axis screw and s_0 is the position vector between the origin frame and any point on the screw axis, $s_0 \times s$ may be defined as the geometric moment of the screw axis about the origin reference frame (CAMPOS, 2004). The pitch h is the relation between the linear and angular displacement, $h = d/\theta$. Therefore, for a prismatic joint is $h = \infty$ and for a revolute joint is $h = 0$. However, for describing completely the displacement have to specify the screw intensity (TSAI, 1999). if let $\dot{q}$ be the intensity, the screw may be written

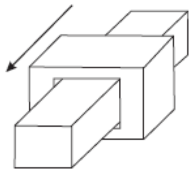
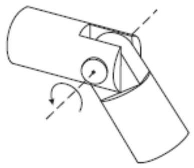
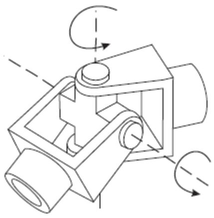
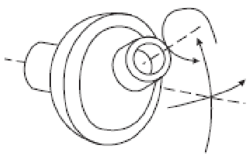
$$\$ = \dot{q}\hat{\$}, \tag{3.2.2}$$

where,

$$\dot{q} \begin{cases} \dot{\theta}\ for\ a\ revolute\ joint \\ \dot{d}\ for\ a\ prismatic\ joint \end{cases} \tag{3.2.3}$$

The screw described in the Eq. (3.2.2) is called the joint twist. As it is shown in Fig. 3.3, each joint can be represented by a twist or twist linear combination.

Figure 3.3 – Common Joints Twist systems and their linear combinations, where $\dot{d}$, $\dot{\theta}_i$, and $\hat{\$}$ are the twits intensity and twist directions respectively.

Pair	Shape	Unit Twist	Liner Combination
Prismatic		$\hat{\$} = \begin{bmatrix} 0 \\ s \end{bmatrix}$	$\dot{d}\hat{\$}$
Revolute		$\hat{\$} = \begin{bmatrix} s \\ s_0 \times s \end{bmatrix}$	$\dot{\theta}\hat{\$}$
Universal		$\hat{\$}_1 = \begin{bmatrix} s_1 \\ s_0 \times s_1 \end{bmatrix}$ $\hat{\$}_2 = \begin{bmatrix} s_2 \\ s_0 \times s_2 \end{bmatrix}$	$\dot{\theta}_1\hat{\$}_1 + \dot{\theta}_2\hat{\$}_2$
Universal		$\hat{\$}_1 = \begin{bmatrix} s_1 \\ s_0 \times s_1 \end{bmatrix}$ $\hat{\$}_2 = \begin{bmatrix} s_2 \\ s_0 \times s_2 \end{bmatrix}$ $\hat{\$}_3 = \begin{bmatrix} s_3 \\ s_0 \times s_3 \end{bmatrix}$	$\dot{\theta}_1\hat{\$}_1 + \dot{\theta}_2\hat{\$}_2 + \dot{\theta}_3\hat{\$}_3$

Source: (SIMONI et al., 2010), (BOHIGAS et al., 2013b)

The platform motion may be described by a twist $(\$_p)$. The linear velocity v_p of a selected point P_0 on the platform and the platform angular velocity ω_p are given according to the task requirements (see Eq. (3.2.4)).

$$\$_p = \begin{bmatrix} \omega_p \\ v_0 \end{bmatrix};\ v_0 = v_p + r_{p0} \times \omega_p, \tag{3.2.4}$$

where r_{p0} is a vector from fixed frame to point P_0 and v_0 is a linear velocity of an instantaneous point in the origin.

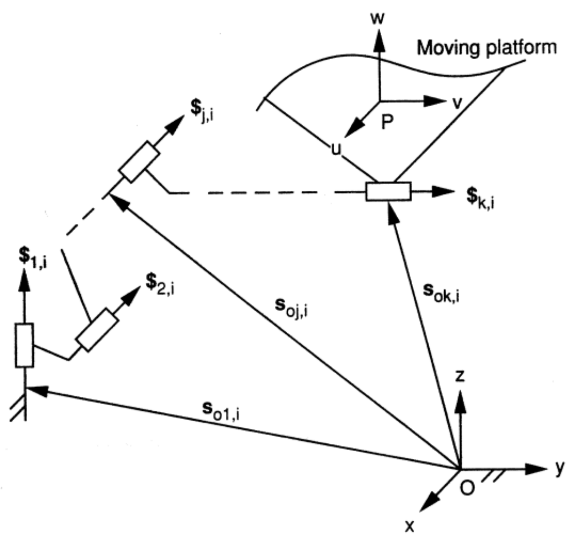
3.2.1 Kinematics

The parallel robots kinematic chains are confirmed by passive and active joints. The active joints are associated with the actuated joint, while the remaining joints are passive. Assuming that parallel robot is symmetric, each kinematic chain could be analyzed as a serial manipulator. Where the platform is the final effector (see Fig. 3.4). Thus, the screw theory may describe the platform resulting motion (twist) by mean of robot joints twist linear combination (TSAI, 1999).

$$\$_p = \sum_{j}^{m} \dot{q}_{i,j} \hat{\$}_{i,j}, \qquad for \quad i = 1...n, \tag{3.2.5}$$

where the unitary twist $\hat{\$}_{j,i}$ first subscript is linked to the joints number and the second to the limbs number.

Figure 3.4 – Platform Twist Related to Each Joints Twist on ith Limb



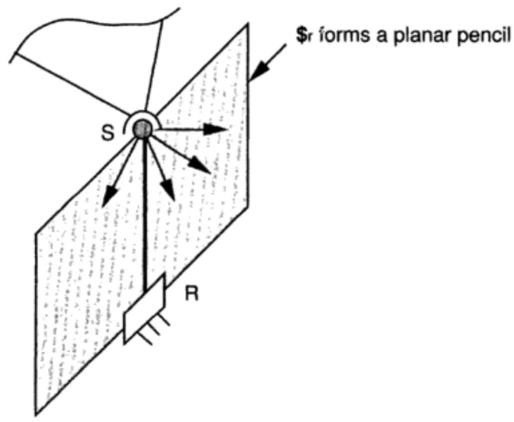
Source: Tsai 1994

3.2.2 Reciprocal Screw

A parallel robot may be confirmed by different kinematic chains types, for each KCs configuration exist a reciprocal unitary screw. In this section show how may be found the reciprocal unitary screw for any typical KCs.

Rotative-Spherical the rotational joint may be represented by a twist with zero-pitch $(h = 0)$, for this joint type the reciprocal screws lie on all the planes containing the axis joint (ZHAO et al., 2009). While the spherical joint may be represented by mean of three twists with zero-pitch $(h = 0)$, the reciprocal screws for this joints conform a three screw system that passes through the joint origin (TSAI, 1999). Therefore, for this KCs configuration the reciprocal screw lie on the plane that containing the axis joint and pass through the spherical joint origin (see Fig. 3.5).

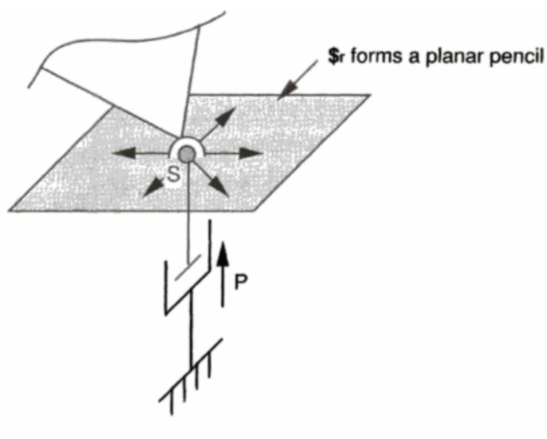
Figure 3.5 – Rotative-Spherical Kcs



Source: (TSAI, 1999)

Prismatic-Spherical the prismatic joint may be represented by a twist with infinity-pitch $(h = \infty)$, for this joint the reciprocal screw lie on the perpendicular plane to the sliding motion (ZHAO et al., 2009). For this KCs configuration the reciprocal screw lie on the perpendicular plane to the prismatic joint sliding motion and pass through the spherical joint origin (see Fig. 3.6)(TSAI, 1999).

Figure 3.6 – Prismatic-Spherical KCs



Source: (TSAI, 1999)

Universal-Spherical the universal joint may be represented by two twists with zero-pitch $(h = 0)$, the reciprocal for this joint pass through the joint origin or lie on the plane formed by the joint twist system(ZHAO et al., 2009). For this KCs configuration, the reciprocal screw passes through the joints origin.

3.2.3 Jacobian Based on Screw

Exist a unitary reciprocal wrench $\hat{\$}_{r,i}$ associated to each kinematic chain, which is reciprocal to all the passive joints (unactuated) $i.e.$ only perform work on the actuated joint. Thus, to eliminate the passive joints velocities (twists) from Eq. (3.2.5), the reciprocal product is done (BONEV, 2002), which it may be expressed as

$$\$_p \circ \hat{\$}_{r,i} = \sum_{j}^{m} \dot{q}_{i,j} \hat{\$}_{i,j} \circ \hat{\$}_{r,i}, \qquad for \quad i = 1...n, \tag{3.2.6}$$

Meaning that the platform velocities may be related only to the actuated joint velocities (DAVIDSON; HUNT, 2004). The Eq. (3.2.6) may be written in the matrix form as

$$J_x \$_p = J_q \dot{q}. \tag{3.2.7}$$

Where J_x is given by

$$J_x = \begin{bmatrix} r \times S_{r,i} + hS_r & S_{r,i} \\ \vdots & \vdots \\ r \times S_{r,n} + hS_{r,n} & S_{r,n} \end{bmatrix}, \tag{3.2.8}$$

and J_q is a diagonal matrix. Where, the diagonal values are conformed by the reciprocal product between the active joint unitary twist $\hat{\$}_{qj,i}$ and the unitary reciprocal wrench $\hat{\$}_{r,i}$ to $i = 1 - n$, it may be written as

$$J_q = \begin{bmatrix} \hat{\$}_{q1,1} \circ \hat{\$}_{r,1} & 0 & \cdots & 0 \\ 0 & \hat{\$}_{q2,2} \circ \hat{\$}_{r,2} & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \hat{\$}_{qm,n} \circ \hat{\$}_{r,n} \end{bmatrix}. \tag{3.2.9}$$

The last term in Eq. (3.2.7) is conformed by the ith intensity $\dot{q}_i$ of each actuated joint that may be described as a vector $\dot{q}^T = [\dot{q}_1 \ldots \dot{q}_n]$. Thus, the Eq. (3.2.7) may be re-written as

$$\begin{bmatrix} \omega_p \\ v_p \end{bmatrix} = J\dot{q}, \tag{3.2.10}$$

where $J = J_x^{-1} J_q$ is called the overall Jacobian. Therefore, J_x and J_q are two separate Jacobian matrices.

Chapter 4

Constraints Formulation

In the proposed analysis the constraints are composed by kinematical (parallel robot singularities) and geometrical restrictions. The first is the particular platform locations, where the PRs may gain, direct singularities, or lose, inverse singularities, degrees of freedom. Thus, the singularity analysis determines the conditions under which singularities occur and how to avoid them (ZLATANOV; GOSSELIN, 2003). The second, geometrical constraints, are due to the parallel robot structural nature. The PRs mobility may be limited by constraints associated with the physical robot parameters.

4.1 Kinematic Singularities

According to Zlatanov and Gosselin (2003), there are three types of kinematic singularities, each with a different physical interpretation. If the matrix J_q is singular, the singularity is inverse. If the matrix J_x is singular, the singularity is direct. And, if two matrices J_q and J_x become singular, the singularity is mixed (see Eq. (3.2.7)).

4.1.1 Inverse Kinematic Singularity

The inverse kinematic singularity type is caused due to the legs serial nature (see Fig. 4.1(a)). It may occur at a workspace boundary or on internal boundaries within the workspace regions (LI, 2005). Zlatanov and Gosselin (2003) proposed an example where the inverse singularity is obtained among workspace regions of a planar parallel robot. This singularity occurs when the determinant of J_q goes to zero.

$$Det(J_q) = 0. \tag{4.1.1}$$

It means that there is a zero platform twist for non zero actuated joint velocities (MERLET, 2006). In other words, for a given non-null velocity, the platform remains immobile, $i.e.$ $\dot{q}$ represents the nonempty null space of the singular matrix J_q (LI; ANGELES, 2018). When the parallel robot is close to inverse singularity, present small velocities on the platform associated to the large actuated joints velocities, turning to the parallel robot accurate. However, this interesting characteristic is difficult to use because operate near the workspace boundaries (MERLET, 2006).

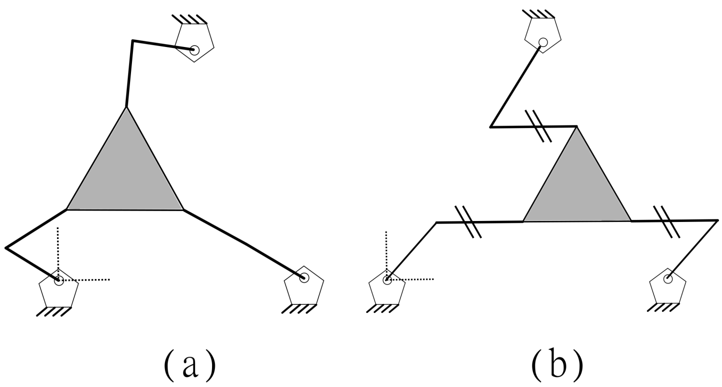
4.1.2 Direct Kinematic Singularity

Direct singularity is more complex than inverse singularity because it appears inside the workspace. In this case, the platform is not controllable (see Fig. 4.1(b)), which means that the parallel robot may gain one or more degrees of freedom. The parallel robot is in direct singularity when the matrix is singular $i.e.$

$$Det(J_x) = 0, \tag{4.1.2}$$

where the platform Twist is the non empty null space. That is, even if the actuated joints are locked the platform may move in some directions (WANG et al., 2018). It means that the parallel robot cannot withstand forces in some directions (MERLET, 2006). It is important to note that, the inverse singularity is not always present in parallel robot and it is easily detected. Whereas the direct singularity occurs only in PRs and it is difficult to be defined. Aiming at identifying the direct singularity, Voglewede (2004) proposed a method to determinate how close to the direct singularity the PRs is, using an optimization problem, where the objective function is the least constrained direction. This method is uses the frame invariant concept, $i.e.$ A value that does not vary due to changes in the frame system position and orientation is called frame invariant, $e.g.$ the distance between two points (VOGLEWEDE, 2004).

Figure 4.1 – (a)Inverse Kinematic Singularity: In this singularity type the platform loses Dof, meaning that the platform can not move in some directions, (b)Direct Kinematic Singularity: In this singularity the platform gains Dof.



Source: Autor

4.1.2.1 Requirements For Direct Singularities Closeness Measures

Direct singularity causes a limitation on the robot that may be written as $M(X)$ at a particular configuration (X), $i.e.$ position and orientation (VOGLEWEDE, 2004). This measure (Index) should have the following properties:

- $M(X) = 0$ if only if X is a singular configuration.
- $M(X) > 0$ if X is non-singular.
- $M(X)$ has a clear physical meaning.

Therefore, this measure determines how close the parallel robot is to a direct singularity. A methodology named power measure or work measure used through the screw theory (VOGLEWEDE, 2004). This technique defined the measures of closeness to Direct singularities by mean of an optimization problem. Considering the objective function is expressed as

$$F = \sum_{i=1}^{n} \left(\hat{\$}_{r,i}^{T} \circ \$\right)^2, \tag{4.1.3}$$

where $\$_r$ is a wrench acting uppon the platform, $\$$ is the platform twist, and n is the total number of limbs. The Eq. (4.1.3) is interpreted as the sum of the square work done by each leg upon the platform motion. Hence, the Eq. (4.1.3) may be rewritten as

$$F = \begin{bmatrix} W_1 & W_2 & \cdots & W_n \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_n \end{bmatrix}, \tag{4.1.4}$$

with

$$[W] = J_x[\$]. \tag{4.1.5}$$

Hence, F may be written in the quadratic form (POTTMANN et al., 1998)

$$F = \$^T J_x^T J_x \$ = \$^T G \$, \tag{4.1.6}$$

where $\$^T$ is the twist transpose, it is denoted as $\$^T = [PQRLMN]$ and G is known as the graminiam matrix given by

$$G = \sum_{i=1}^{n} \$_{r,i} \cdot \$_{r,i}^{T}. \tag{4.1.7}$$

Note that $\$_r \cdot \$^T$ is similar to the reciprocal screw product shown in preceding sections. The optimization problem constraint is the invariant norm, which takes the frame-invariant screw portion magnitude (VOGLEWEDE, 2004). It is defined as

$$\begin{aligned} \|\$\| &= \sqrt{\omega \cdot \omega}, \\ &= \sqrt{\$^T D \$}, \end{aligned} \tag{4.1.8}$$

where

$$D = \begin{bmatrix} 1_{(3x3)} & 0_{(3x3)} \\ 0_{(3x3)} & 0_{(3x3)} \end{bmatrix}. \tag{4.1.9}$$

For a pure translation case ($i.e.\quad \omega = 0$). The invariant norm is

$$\|\$\| = \sqrt{v \cdot v} \tag{4.1.10}$$

Considering the Eq. (4.1.6) and Eq. (4.1.8). The optimization problem to measure closeness to direct singularity may be expressed as

$$M(X) = \begin{cases} \min\limits_{\$} \qquad F(\$) = \$^T G \$, \\ h(\$) = \quad \$^T D \$ - 1 = 0. \end{cases} \tag{4.1.11}$$

This constrained problem may be transformed in the unconstrained problem by mean of a Lagrange function (L) by introducing one Lagrange multiplier for each constraint

$$\min_{\$, \lambda} \quad L(\$, \lambda), \tag{4.1.12}$$

where the Lagrange function L is described as

$$L = \$^T G \$ + \lambda(\$^T D \$ - 1). \tag{4.1.13}$$

Differentiating the Lagrange function respect to λ

$$\frac{\partial L(\$, \lambda)}{\partial \lambda} = \$^T D \$ - 1 = 0, \tag{4.1.14}$$

and differentiating the Lagrange function with respect to $\$$, and using the fact that D and G are symmetrical yields

$$\frac{\partial L(\$, \lambda)}{\partial \$} = (G - \lambda D)\$ = 0. \tag{4.1.15}$$

Note that $\partial L(\$, \lambda)/\partial \lambda$ is the optimization constraint exposed in Eq. (4.1.11). Whiles for $\partial L(\$, \lambda)/\partial \$$, the matrix expression in the parenthesis has to be singular for a non-trivial solution. In other words

$$det((G - \lambda D)) = 0. \tag{4.1.16}$$

The Eq. (4.1.16) is the corresponding eigenvalue problem that may be rewritten as

$$det(\xi I - G^{-1}D), \tag{4.1.17}$$

where $I_{(6\times6)}$ is an identity matrix, and $\xi = 1/\lambda$ is the eigenvalue of $[G^{-1}D]$. Thus, the minimal function value is related to the minimal eigenvalue (VOGLEWEDE, 2004). It may be proven rewriting the Eq.(4.1.14) as

$$G\$ = \lambda D\$. \tag{4.1.18}$$

Substituting the Eq.(4.1.18) into the objective function (F) and using the constraint h

$$F = \$^T G\$ = \lambda \$^T D\$ = \lambda. \tag{4.1.19}$$

Thus the minimization problem may be written as

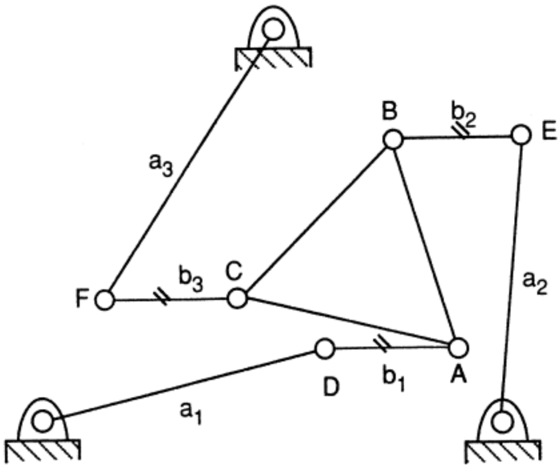
$$\min_{\$,\lambda} \quad L(\$, \lambda) = \lambda_{min}. \tag{4.1.20}$$

Since the objective function is non-negative, due to G is a square symmetric positive semi-definite matrix, $\lambda \geq 0$ is non-negative (VOGLEWEDE, 2004).

4.1.3 Combined Singularity

The parallel robot may fall in combined singularity, $i.e.$ these singularities appear if both J_q and J_x become simultaneously singular (TSAI, 1999). In this situation, the platform experiments infinitesimal motions in some directions, even if the actuated joints are locked (see Fig. 4.2). Hence, in such a case, the robot cannot be controlled (LI, 2005).

Figure 4.2 – Planar robot Combined Kinematic Singularity



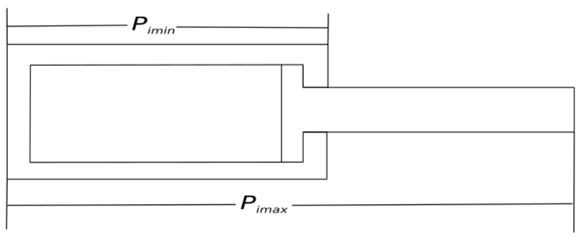
Source:(TSAI, 1999)

4.2 Geometrical Constraint

The geometrical constraints occur due to a collision between kinematic chains or due to a violation of joint limits, which are specific for each joint according to its range of motion. These restriction types generally may be expressed explicitly.

Prismatic Joint In this case, the limitations are given by the joint sliding range (see Fig. 4.3). The minimum length of ith link is denoted by ρ_{imin} and the maximum length by ρ_{imax}.

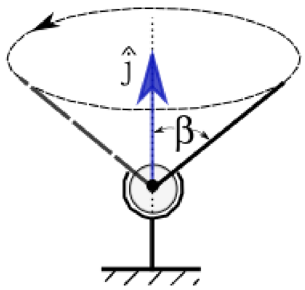
Figure 4.3 – Prismatic Joint Mobility



Source: Author

Spherical Joint Spherical joint limitations may be represented through a cone, which defines the joint mobility or its range of motion (see Fig. 4.5). In this case, the cone angle β is the maximum misalignment angle of the joint (assumed to be less than $90°$), and $\hat{\mathbf{j}}$ is the unit vector along the axis of symmetry (BONEV; RYU, 2001).

Figure 4.4 – Spherical Joint Mobility



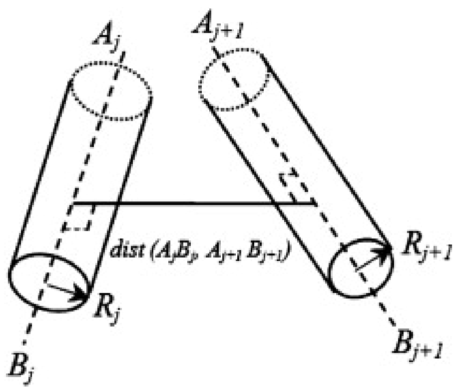
Source: (BONEV; RYU, 2001)

4.2.1 Kinematic Chain Collision

The kinematic chain collision may be described as an intersection between two line. It is important to notice that, the distance between these lines is given by their common normal line magnitude. Assume that the axes of two cylindrical segments (usual link geometry) A_jB_j and $A_{j+1}B_{j+1}$ with radius $R_j \; and \; R_{j+1}$ do not collide if their common normal line norm follows the condition:

$$dist(A_jB_j, A_{j+1}B_{j+1}) \geq R_j + R_{j+1}. \tag{4.2.21}$$

Figure 4.5 – Distance Between two lines



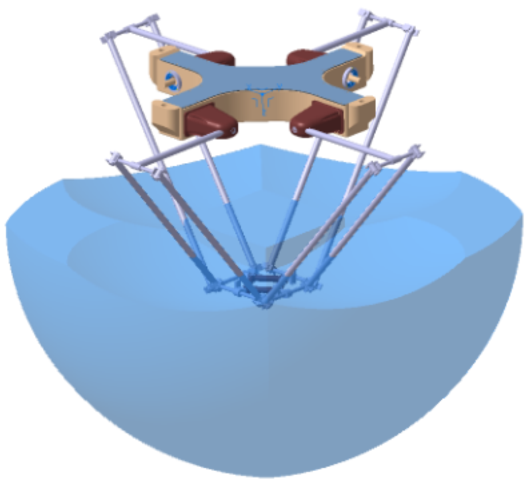
Source: (KELAIAIA et al., 2012)

Chapter 5

Workspace

The workspace is the achievable region by a central point on the platform (see Fig 5.1). For the majority of the parallel robot, the platform motions may be translational, rotational, or a combination of two. Considering this, the workspace may be classified into two principals class: Translational Workspace and Orientation Workspace (JIANG, 2008).

Figure 5.1 – 3T1R Parallel robot Workspace



Source: (MARTÍNEZ, 2013)

5.1 Translational Workspace

The translational workspace is considered the attainable region by a central point on the platform in the space when the platform maintains a constant orientation (BONEV, 1998). There are some works that present different methodology intended for the translational workspace analysis (JIANG, 2008; BONEV, 1998; MERLET, 2006).

5.2 Orientation Workspace

The orientation workspace is more complex than the translational workspace. It is defined as the orientations set reached by the platform around a point (BONEV, 1998). Generally this point represents the coordinates system origin fixed in the platform. there are different alternatives to analysis the orientation workspace (JIANG, 2008). In this case, the RPY Angles are used to represents the orientation workspace (see Eq.(2.5.40)).

5.3 Workspace Calculation Methods

There are various methods to calculate the parallel robot orientation workspace, these methods may be divided into three main groups: Numerical, geometrical and discretization method (MERLET, 2006).

Geometric Method The geometrical method determines geometrically the orientation workspace boundary (MERLET, 2006). Considering this, to describe geometrically the constraints related to the workspace boundary is necessary (SAPUTRA et al., 2015). Thus, the formed space by the intersection of the geometrics functions describes all platform possible positions that satisfy the geometrical constraints (MERLET, 2006). This method is an efficient and accurate mapping of the workspace boundary. However, The restriction may not be always geometrically represented (SAPUTRA et al., 2015).

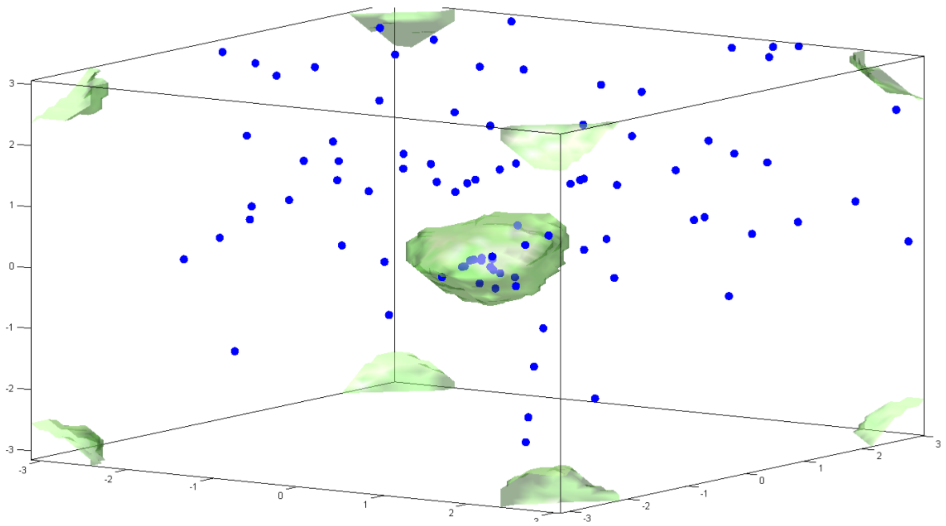
Discretization Method In this discretization approach, the workspace is covered by a regular grid, where each node size is specified as a sampling step (MERLET, 2006). These nodes are tested to see whether is pertain to the workspace. Due to this, it needs a longer computation time. another disadvantage is the accuracy depends on the sampling step that is used to create the grid and generally fails to detected voids (SAPUTRA et al., 2015).

Numerical Method Several researchers have suggested different approximation method to calculate the orientation workspace. The Jacobian method proposed by (JO; HAUG, 1989), introduced a new variables set called generalized variables to transform the inequalities constraints in equalities (MERLET, 2006). In this case, the inequalities functions are kinematic constrains equations that describe the attainable motion range by the PRs (SAPUTRA et al., 2015). Let be J a Jacobian matrix formed by this equations. Then, If the Jacobian rank is lower than its dimension, the corresponding configuration is at the orientation workspace boundary. However, the introduction of other constraints limiting the workspace is a drawback. Due to would lead to so larger Jacobian matrix as to render the procedure quite difficult to manage (MERLET, 2006).

5.4 Workspace Optimization

In the present study, the workspace optimization aims at identifying the higher singularity free sphere, also called the optimal sphere, which represents the reachable orientations by the platform in the three-dimensional space. But due to the orientation workspace geometry, this task is complex. Thus, to find the optimal sphere, a genetic algorithm is implemented. Those algorithms are a robust type of evolutionary algorithms, which explore all the space avoiding fall in local minimums (see Fig. 7.17). The optimization analysis takes into account the kinematic and geometric robot constraints.

Figure 5.2 – Individuals Explroring the 6 UPS Orientation Workspace.



Source: Author

5.4.1 Genetic Algorithm (GAs)

A Genetic Algorithm is an increasingly popular method of optimization being applied to many fields. Motivated by the “survival of the fittest” concept and Darwin’s theory of natural selection. Therefore, this algorithm uses processes analogous to biological evolution to promote the better genes of a population. In the GAs search process, only the function values are used to make progress toward a problem solution. The problem functions differentiability is neither required for the algorithm calculations. Therefore, it may be applied to all kinds of problems: discrete, continuous, and nondifferentiable. For this reason, the GAs is widely used in different practical engineering problem (ARORA, 2004).

The algorithm starts by generating an initial population of random candidates solutions. Each individual, in the population, is then awarded a score based on its performance. For this purpose, the candidates are represented by binary strings, and the GAs population size is fixed (WEILE; MICHIELSSEN, 1997). The essential elements of natural genetics are reproduction, crossover, and mutation, which are used in the genetic search procedure (RAO; RAO, 2009).

5.4.1.1 Design Variables Representation

Each design variable may be coded in a binary string of length q. Therefore, if the optimization problem is represented by n design variables x_n, the design vector is represented using a string of total length nq (RAO; RAO, 2009). Thus, to decodify the design variable from binary string to decimal number the Eq.(5.4.1) and Eq.(5.4.2) are used.

$$y = \sum_{k=0}^{q} 2^k b_k, \tag{5.4.1}$$

where $b_k = 0$ or 1. The Eq.(5.4.1) represents the equivalent decimal number y (integer), while the Eq.(5.4.2) represents its decimal value as

$$x = x_l + \frac{x_u - x_l}{2^q - 1} \sum_{k=0}^{q} 2^k b_k, \tag{5.4.2}$$

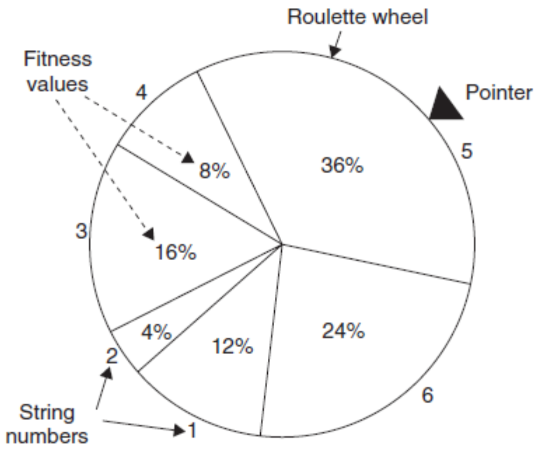
$x_u,\ x_l$ are the upper and lower bounds of design variable x. Thus to represent a variable with high accuracy $\triangle x$, a large value of q is used in its binary representation. q may be calculated as

$$2^q \geq \frac{x_u - x_l}{\triangle x} + 1. \tag{5.4.3}$$

5.4.1.2 Reproduction

Reproduction operation is also called the selection operator due to it select good strings from the current population to form a mating pool. This operation is the first operation applied to the GAs population (RAO; RAO, 2009). The reproduction operator is biased toward to pick above-average strings of the current design set (population). Thus, multiple copies of better strings are inserted in the mating pool based on a probabilistic procedure. Usually, a string is selected from the mating pool with a probability proportional to its fitness (see Fig. 5.3), $i.e.$ those with higher fitness should have a greater chance of selection (MCCALL, 2005).

Figure 5.3 – Roulette-Wheel Selection Scheme.



Source: (RAO; RAO, 2009)

Thus if F_i denotes the fitness of the ith string in the population of size n, its probability of selection is calculated as

$$P_i = \frac{F_i}{P_f}; \quad P_f = \sum_{j=1}^{n} F_j \quad i = 1, 2, ..., n. \tag{5.4.4}$$

5.4.1.3 Crossover

Crossover is an operation to introduce a variation into a population. It creates a new string by exchanging information among strings of the mating pool (RAO; RAO, 2009). Thus for the crossover operation are selected two individual strings, which are known as parent string. The latter are picked at random from the mating pool generated by the reproduction. Then, an information portion is exchanged between the parents giving a resultants string known as child strings (MCCALL, 2005). With the intention to preserve some good strings for the next generation, a crossover probability, p_c, is used. Thus only $100p_c$ of the strings in the mating pool will be used and $100(1-p_c)$ percent of the strings will be retained. Let $x_1,\ x_2$ be two design vectors with a string length of 10

(parent 1) $x_1 = \{0\ 1\ 0\ 1|\ 1\ 1\ 1\ 0\ 1\ 1\}$,
(parent 2) $x_2 = \{0\ 1\ 1\ 1|\ 0\ 1\ 0\ 0\ 1\ 0\}$.

When the crossover site is 4 the result is

(child 1) $x_3 = \{0\ 1\ 0\ 1|\ 0\ 1\ 0\ 0\ 1\ 0\}$,
(child 2) $x_4 = \{0\ 1\ 1\ 1|\ 1\ 1\ 1\ 0\ 1\ 1\}$.

5.4.1.4 Mutation

Mutation introduces traits which are not in the original population, modifying a certain percentage of the bits in the list of chromosomes. The latter keeps the GA from converging too fast before sampling the entire variable space. Therefore, the mutation operation permit to search outside the current region of variable space (SYAHPUTRA, 2017). In practice, a mutation is applied to the new strings with a specific small mutation probability, pm. This operation modifies the binary digit 1 to 0 and vice versa. Consider the following population

$$\{0\,1\,0\,1\,1\,1\,1\,0\,1\,1\},$$
$$\{0\,1\,1\,1\,0\,1\,0\,0\,1\,0\}.$$
$$\{0\,1\,0\,1\,0\,1\,0\,0\,1\,0\},$$
$$\{0\,1\,1\,1\,1\,1\,1\,0\,1\,1\}.$$

The next population as result of mutation operator may be given by

$$\{0\,1\,1\,1\,0\,1\,1\,0\,1\,1\},$$
$$\{0\,1\,1\,1\,0\,1\,0\,0\,1\,0\}.$$
$$\{0\,1\,0\,1\,0\,1\,0\,0\,1\,0\},$$
$$\{0\,1\,1\,1\,1\,1\,0\,0\,1\,1\}.$$

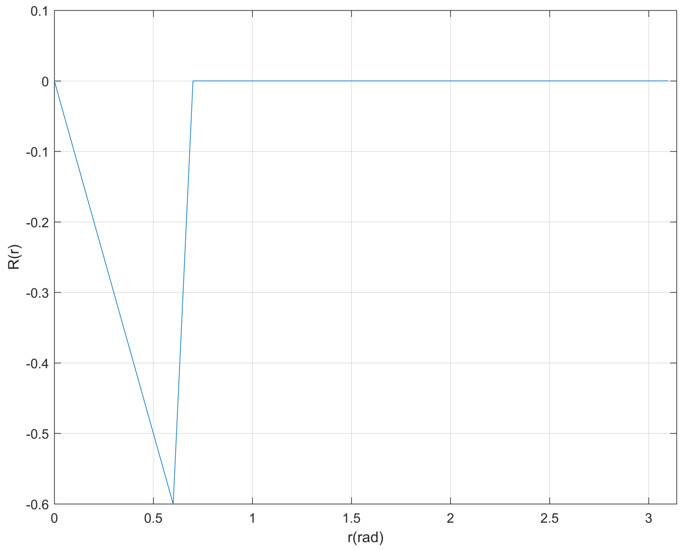
5.4.1.5 Objective Function

The objective function, also called the fitness function, is related to the parameter to be maximized. In the present optimization analysis, the parameter is the sphere radius, which is dependent on the sphere origin location. Therefore, to determine the optimal solution, the GA algorithm explore different origin positions into the orientation workspace, $i.e.$ the origin sphere locations. However, for each origin position, an optimal radius (r) exists. In view of this, the optimization analysis is split into two parts. In the first part, the maximal radius is determined by means of the golden search procedure. While in the second part, the optimal sphere origin is identified by GAs. Thus, the fitness function is given by:

$$f(o_x, o_y, o_z) = abs(min(R(o_x, o_y, o_z, r))). \tag{5.4.5}$$

On the right side of Eq. 5.4.5 the sphere radius (r) is a variable and sphere origin coordinates (o_{x1}, o_{x2}, o_{x3}) are fixed values. Thus, the function $R(r)$ determines the feasible (r) values associated with a specific sphere origin, whose value could range from 0 to $-r$ in the span $0 \leq r \leq r_{max}$. In this work, r_{max} is associated to the feasible maximum sphere radius within the workspace. For the S-G platform, the analysis is given by $0 \leq r \leq \pi$, as shown in Fig. 5.4.

Figure 5.4 – $R(x)$ Funcntion value with $o_{x1} = 0,\ o_{x2} = 0,\ o_{x3} = 0$ for a S-G platform



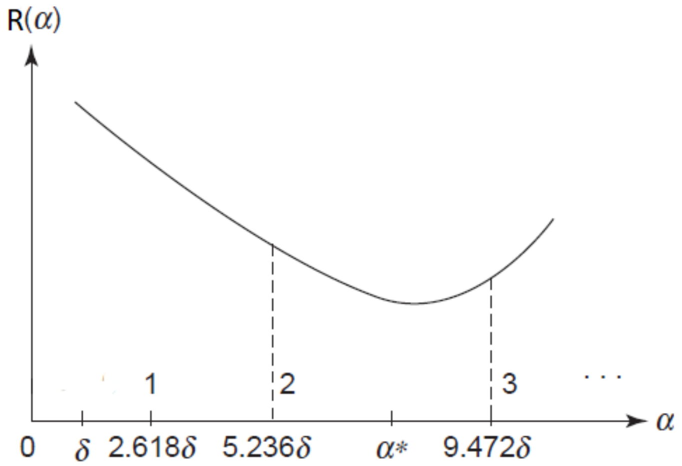
Source: Author

Golden Section The golden section method is used to obtain the minimum $R(r)$ function value. The method evaluates the function at predetermined points α_j as it is shown in Fig. 5.5. The procedure starts Bracketing a function interval I using a sequence of larger increments based on the golden ratio, in which an increment $\delta > 0$ is selected to bracket. The interval I, given by $I = \alpha_u - \alpha_l$, where $\alpha_l = 0,\ \alpha_u = \pi$ are the interval limits initially. Due to do not know the function value at the minimum, the increment δ is multiplied by a constant $r_c > 1$. In the explained methodology, the r_c value is selected as 1.618, which it is known as the golden ratio (ARORA, 2004). Thus, the function may be evaluated at the n points initially, which these are given by Eq 5.4.6 .

$$\alpha_j = \sum_{j=0}^{n} \delta(1.618)^j. \tag{5.4.6}$$

Subsequently, two points α_a and α_b are established, such that they are locate at (τI) and $(1-\tau)I$ as is shown in Fig.5.6(A), where $\tau = 0.618$. Then, the interval I is reduced by mean of evaluating and comparing the function at points α_a and α_b in each step (see Fig.5.6).

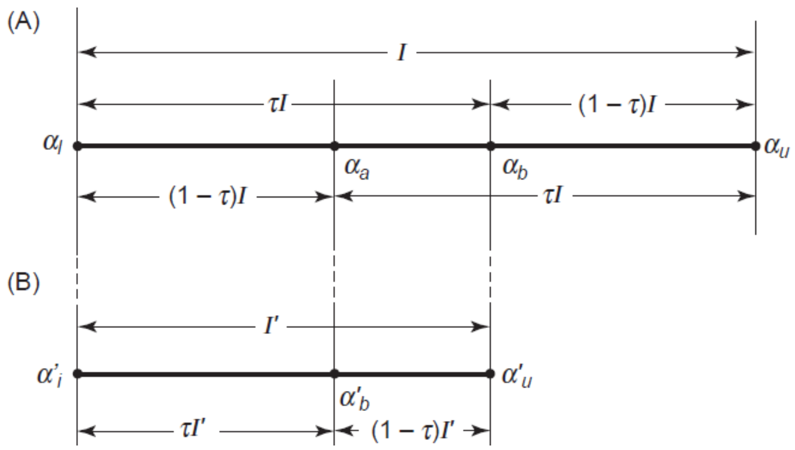
Figure 5.5 – Initial Bracketing in the Golden Section Search



Source: Author

If $R(\alpha_a) < R(\alpha_b)$ the minimum point lies between α_l and α_b. Therefore, the new bracketing becomes $\alpha_l,\ \alpha_u = \alpha_b$. If $R(\alpha_a) > R(\alpha_b)$, then the new bracketing becomes $[\alpha_l = \alpha_a,\ \alpha_u]$. Finally, if $R(\alpha_a) = R(\alpha_b)$ the new bracketing becomes $[\alpha_l = \alpha_a,\ \alpha_u = \alpha_b]$, where $\alpha_a = \alpha_a + 0.382I$ and $\alpha_b = \alpha_l + 0.618I$. The golden section search is applicable to extremizing functions of one variable only (JIA, 2017).

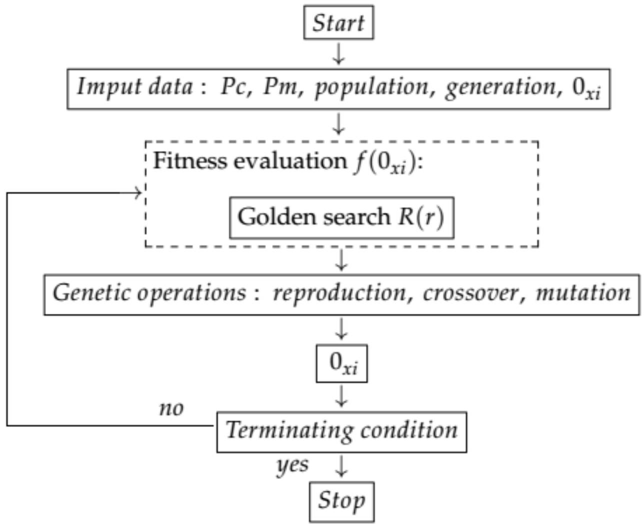
Figure 5.6 – Golden Section Partition



Source: (ARORA, 2004)

If the new interval I is small enough to satisfy a stopping criterion ε, the optimal point is $\alpha = (\alpha_u + \alpha_l)/2$. The Genetic algorithm flowchart is shown in Fig. 5.7.

Figure 5.7 – Genetic algorithm flowchart



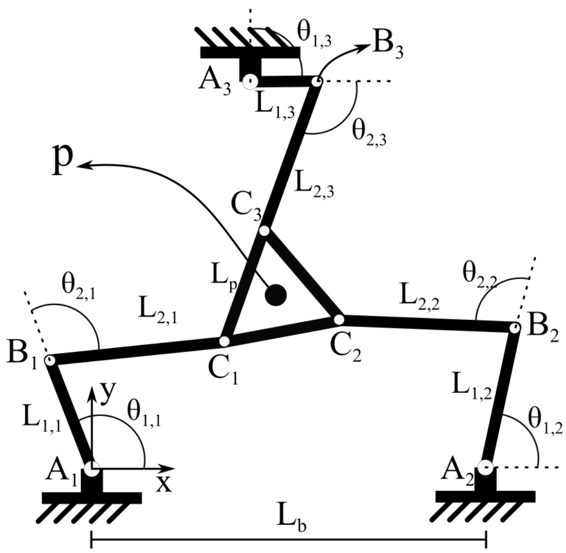
Source: Author

Chapter 6

Planar Parallel Robot 3-RRR Workspace Analysis

The planar parallel consists of a moving platform linked to the fixed base by means of three legs, where each leg is a three-revolute chain (see Fig. 6.1). The first rotational joint, attached to the base, is the actuated one in every kinematic chain. Therefore the parallel robot is named 3-RRR, where the underline indicates the actuated joint.

Figure 6.1 – Planar Parallel Robot 3-RRR



Source: Author

Let $O_1, O_2,$ the fixed coordinate systems, which are located at point P on the platform and at the ith point A_i, in each limb, respectively. Where the frame coordinate system O is fixed on point A_1.

6.1 Kinematics of 3-$\underline{R}$RR

Kinematics of 3-$\underline{R}$RR considers the task of computing joints variable $(\theta_{1,i})$ for a given platform configuration (px, py, ψ), $i.e.$, solve the inverse kinematic problem. Where (px, py) are the platform position in the plane x-y and ψ is the angle associated to the platform orientation, which is described by means of orientation matrix $R(\psi)$ (see Eq.(2.5.37)). Geometrically, for ith leg, the problem may be described as

$$ac_i = op + R(\psi)c_i - oa_i, \tag{6.1.1}$$

where the variables of Eq.(6.1.1) are related to the joint and platform coordinates, which are given in table 6.1 and ac_i is the C_i point position.

Table 6.1 – Joint and Platform Coordinates

oa_i	c_i	op
$[0\ 0\ 0]^T$	$[-L_p/2\ \ -(\sqrt(3)/4)L_p\ 0]^T$	
$[L_b/1.45\ 0\ 0]^T$	$[L_p/2\ \ -(\sqrt(3)/4)L_p\ 0]^T$	$[px\ py\ 0]^T$
$[0\ (\sqrt(3)/2)L_b\ 0]^T$	$[0\ (\sqrt(3)/4)L_p\ 0]^T$	

Source: Author

In this way, to computing the joints variables $\theta_{1,i}$ is necessary computed the joints $\theta_{2,i}$ values, which is described as

$$\theta_{2,i} = cos^{-1}(m_1/m_2) + \pi, \tag{6.1.2}$$

where

$$\begin{gathered} m_1 = ac_{i,x}^2 + ac_{i,y}^2 - L_{1,i}^2 - L_{2,i}^2, \\ m_2 = 2L_{1,i}L_{2,i}; \end{gathered} \tag{6.1.3}$$

In the Eq. (6.1.3) $L_{1,i}$ and $L_{2,i}$ are the length of the links (see Fig. 6.1), while $ac_{i,x}, ac_{i,y}$ are the components of the vector ac_i. Thus, the joints variables $\theta_{1,i}$ may be written as

$$\theta_{1,i} = tan^{-1}(ac_{i,y}/ac_{i,x}) - sin^{-1}(m_3), \tag{6.1.4}$$

where

$$m_3 = L_{2,i}sin(\theta 2, i)/\sqrt{ac_{i,x}^2 + ac_{i,y}^2}. \tag{6.1.5}$$

6.2 Screw and Reciprocal Screw

The screw theory is a mathematical tool which can be used for parallel robot analysis, which is represented by two three-dimensional vectors (HUANG et al., 2012). In this case, for the revolute joint the screw may be written as

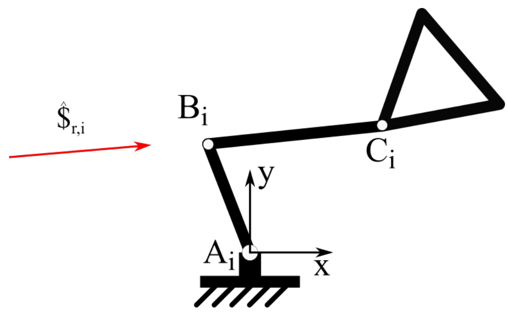
$$\hat{\$} = \begin{bmatrix} \hat{s}_i \\ s_{0,i} \times \hat{s}. \end{bmatrix} \tag{6.2.6}$$

The unit vector $\hat{s}$ is along the screw axis and s_0, which has the same direction that the $z-$ axes. s_0 is the position vector between the origin point on the frame and any point on the screw axis, which is described as

$$s_{0,i} = [L_{1,i}cos(\theta_{1,i})\ L_{1,i}sin(\theta_{1,i})\ 0]. \tag{6.2.7}$$

Based on Bonev (2002), the screw needs to be modified when it is applied to mechanisms with $n < 6$ DOF, $i.e.$ the twists and wrenches involved in the velocity and singularity analysis (BONEV, 2002). The robot considered in this analysis is a 3 DOF. Therefore, the planar twists may be describe as $\$ = [\omega_z, \nu_x, \nu_y]$. Where ω_z is in the $\hat{s}$ direction and ν_x, ν_y are the components of cross product $s_{0,i} \times \hat{s}$

Figure 6.2 – ith kinematic chain for Parallel Robot 3-RRR



Source: Author

There is a wrench that only performs work on the actuated joint, which is called reciprocal (DAVIDSON; HUNT, 2004). For the ith kinematic chain show Fig. 6.2, the unitary reciprocal wrench $(\hat{\$}_r)$ passes through the two rotational joints (ZHAO et al., 2009). Thus, the unit reciprocal direction is presented in Eq. (6.2.8),

$$\hat{\$}_{r,i} = \begin{bmatrix} \hat{s}_{r,i} \\ ac_i \times \hat{s}_{r,i} \end{bmatrix}, \tag{6.2.8}$$

where

$$\hat{s}_{r,i} = ac_i - s_{0,i}. \tag{6.2.9}$$

For the planar case, the wrench is given by $\$_{r,i} = [f_x, f_y, C_z]$.

6.3 Screw-Based Jacobian

Let be $(\$_p)$ the platform twist motion, which may be described as robot joint twist linear combinations using the screw theory, it is presented in the Eq.(6.3.10).

$$\$_p = \sum_{i=1}^{n} \dot{\theta}_{1,i}\$_{1,i} + \dot{\theta}_{2,i}\$_{2,i} + \dot{\theta}_{3,i}\$_{3,i}. \tag{6.3.10}$$

Premultiplying Eq.(6.3.10) by the reciprocal wrench $\$_{r,i}$ and writing in the matrix form

$$J_x\$_p = J_q\dot{\theta}_{1,i}, \tag{6.3.11}$$

where J_x is conformed by the transpose of the reciprocal wrench

$$J_x = \begin{bmatrix} ac_1 \times s_{r,1} & s_{r,1} \\ ac_2 \times s_{r,2} & s_{r,2} \\ ac_3 \times s_{r,3} & s_{r,3} \end{bmatrix}, \tag{6.3.12}$$

and J_q is a diagonal matrix. Where, the diagonal values are conformed by the reciprocal product between the active joint unitary twist $\hat{\$}_{1,i}$ and the unitary reciprocal wrench $\hat{\$}_{r,i}$, which may be written as

$$J_q = \begin{bmatrix} \hat{\$}_{1,1} \circ \hat{\$}_{r,1} & 0 & 0 \\ 0 & \hat{\$}_{1,2} \circ \hat{\$}_{r,2} & 0 \\ 0 & 0 & \hat{\$}_{1,3} \circ \hat{\$}_{r,3} \end{bmatrix}. \tag{6.3.13}$$

6.4 Constraint Formulation

The constraint formulation considered that the 3-RRR mobility is restricted by the Direct and inverse singularities.

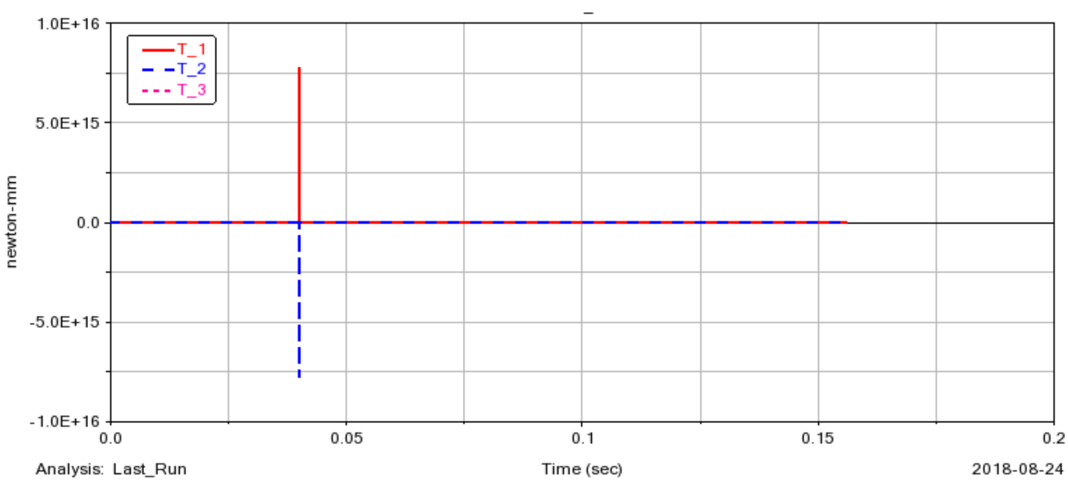
Inverse Singularity To avoid the inverse singularity, the Eq.6.4.14 is considered. This equation is computed by mean an algorithm in MATLAB, where the J_q determinant is evaluated in each platform configuration.

$$L_y > 1^{-10}, \tag{6.4.14}$$

where $L_y = abs(det(J_q))$.

Direct Singularity A kinematic model, in ADAMS, is developed to identify the 3-RRR index value. In this software, different platform configurations are explored aiming to detect possible points within the parallel robot workspace associated with the direct singularity. These points are identified through the measurement of the actuated joints reaction torque τ. It is possible due to legs reaction torque tends to infinity in this singularity type as shown Fig.(6.3).

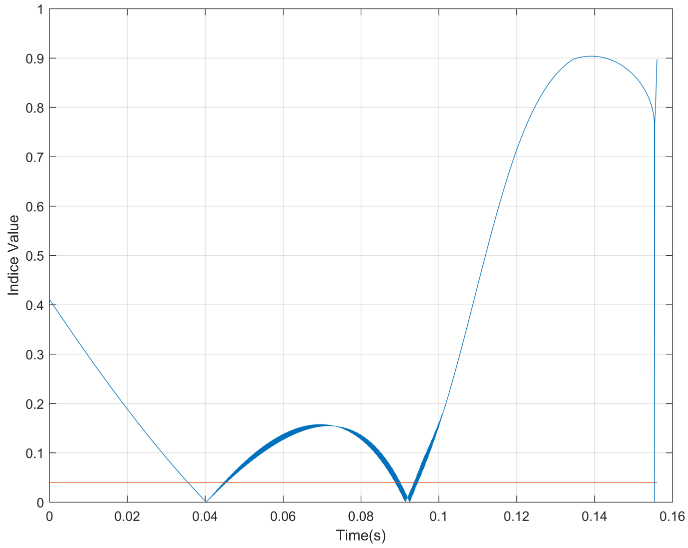
Figure 6.3 – Planar Parallel Robot 3-RRR legs reaction torque measure in MSC ADAMS/View



Source: Author

An algorithm in MATLAB is applied to aim at measuring the robot direct singularity closeness λ in each platform configuration as shown in Fig. 6.4. This figure is presented the indice behavior for a given platform configuration in a lapse of time t.

Figure 6.4 – Masure index associated to the platform configuration



Source: Author

The index limit is associated to the point where the reaction leg torque suffers at high increase, Therefore, the index limit value is calculated measuring the direct kinematic index, where the actuated joints reactions torque τ tends to infinity. In this case, the constraint equation for the direct kinematic singularity may be written as:

$$\sqrt{\lambda_{min}} > 0.04. \tag{6.4.15}$$

6.4.1 Workspace Analysis

In this section the 3-RRR workspace is presented. Considering the kinematic constraints mentioned above. In the proposed analysis the base $(L_b = 260mm)$, the platform $(L_p = 85.73mm)$ and platform initial configuration $(P_x = 97mm,\ P_y = 96mm,\ \psi = 0°)$ are regarded. The links lengths $L_{1,i}$ and $L_{2,i}$ are presented in the Table 6.2. The analysis is reduced initially to the achievable orientations in the plane $y - x$ due to the complex geometry of the orientation workspace.

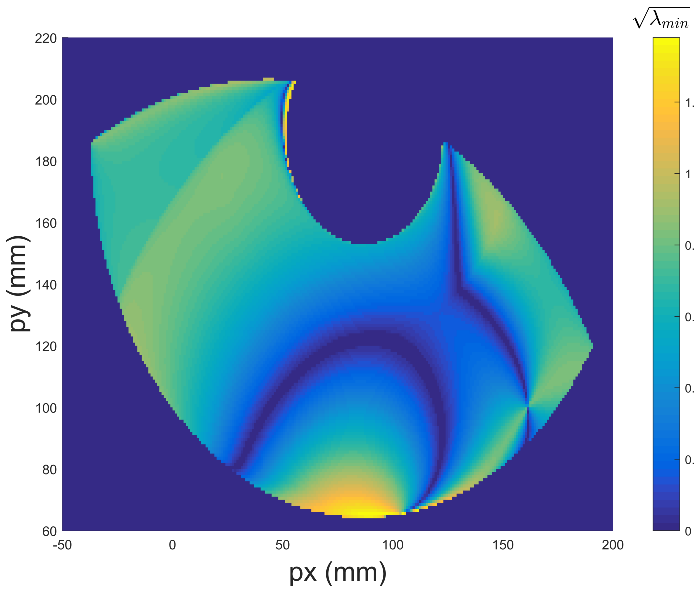
Table 6.2 – links lengths $L_{1,i}$ and $L_{2,i}$ for the ith leg
.

i	$L_{1,i}(mm)$	$L_{2,i}(mm)$
1	80	88.90
2	100	88.90
3	44.4	80

Source: Author

The direct kinematic index value related to each platform configuration $(p_x,\ p_y,\ 0)$ is shown in Fig. 6.5. In this case, the dark blue regions are related to the platform inoperative zones, $i.e.$ $\sqrt{\lambda_{min}} < 0.04$. While the remaining regions are considered feasible. For the parallel robot in the study, there are infeasible regions within the workspace. It means that the platform may become uncontrollable in some task for a given path or to achieve some specific configuration.

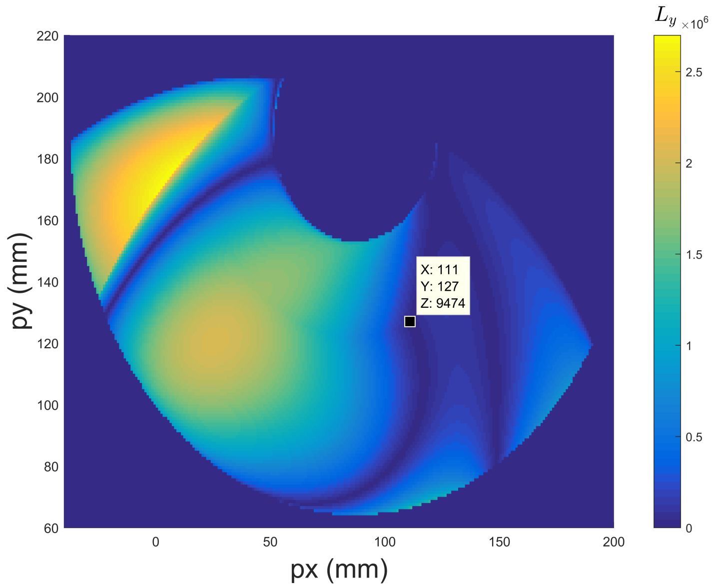
Figure 6.5 – Workspace Associated to Direct Kinematic Index



Source: Author

The inverse kinematic value related to each platform configuration $(p_x,\ p_y,\ 0)$ is shown in Fig. 6.6. Similar to the workspace associated with the direct kinematic index, the dark regions are related to the platform inoperative zones, $i.e.$ $L_y < 1^{-10}$. In this case, there are no infeasible regions within the workspace. Therefore, the dark blue regions within the workspace corresponds to lower values of L_y in relation to the high values reached by platform configuration in other points, which may be observed in Fig. 6.6. These high values are in the order of 10^6.

Figure 6.6 – Workspace Associated to Inverse Singularity

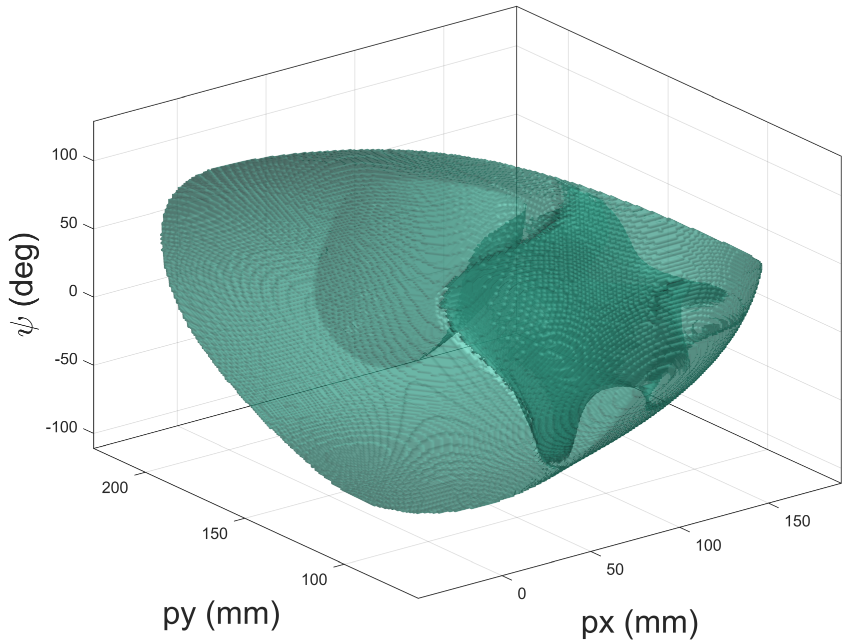


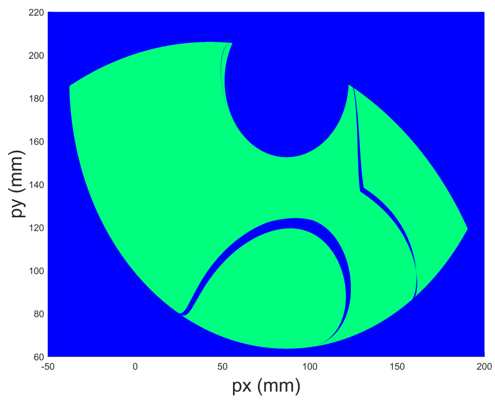
Source: Author

In the Fig. 6.7 is presented the parallel robot 3-<u>R</u>RR workspace, which exhibits different feasible regions (in green) for given platform orientation ψ. These allow understanding the workspace geometry. The Fig. 6.7(a) presents the workspace related to kinematics constraints studied above. In this case, the unfeasible regions (in blue) within the workspace are due to the direct singularity, which is located in a specific zone. It may be observed elaborately in the Fig. 6.7 (a-d).

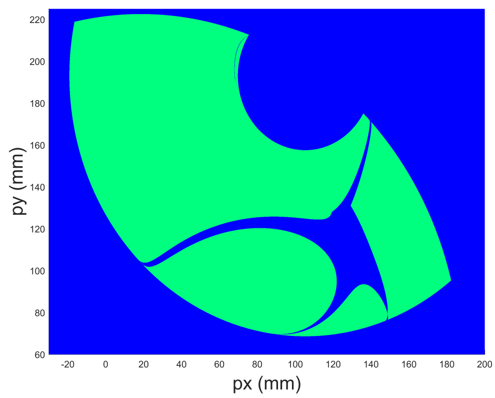
Figure 6.7 – Parallel robot 3-RRR Workspace

.

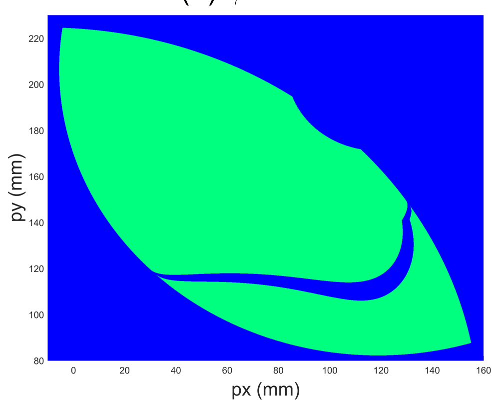




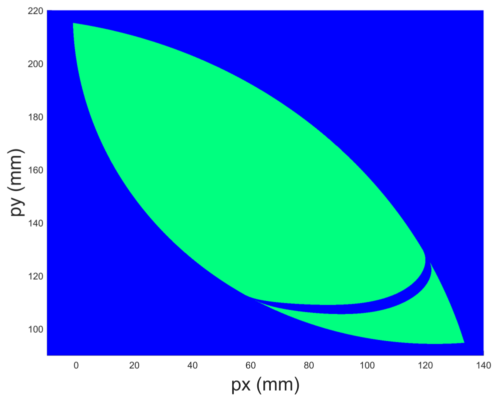
(a) $\psi = 0°$



(b) $\psi = 30°$



(c) $\psi = 60°$



(d) $\psi = 80°$

Source: Author

6.5 Workspace Optimization

The optimization problem may be described as

$$\begin{aligned} maximize \qquad & f(0_{xi}) \\ \\ subject\ to: \qquad & L_y > 1^{-10} \\ & \sqrt{\lambda_{min}} > 0.04, \\ & 0 \leq r \leq 130 \\ & 60 \leq p_x \leq 230 \\ & -50 \leq p_y \leq 200 \\ & -110 \leq \psi \leq 130 \end{aligned} \tag{6.5.16}$$

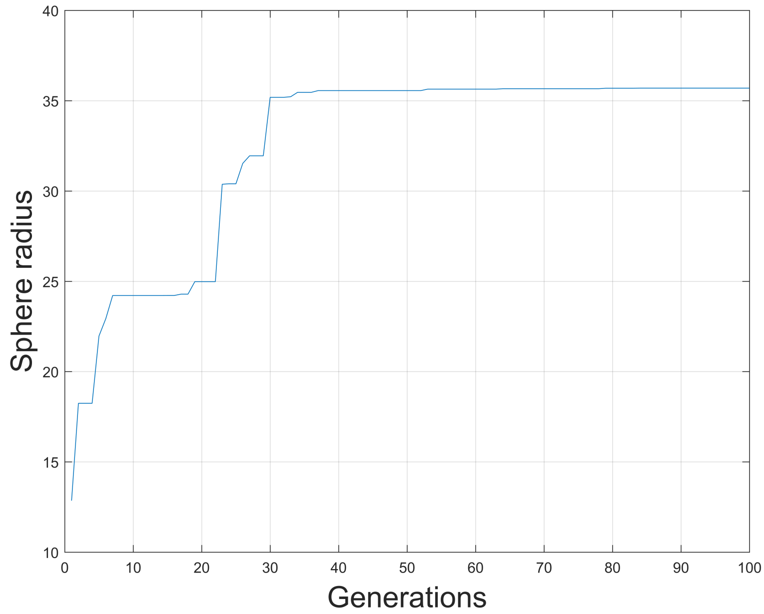
where $0_{xi} = [p_x,\ p_y,\ \psi]$, which represents the sphere origin coordinates. The 0_{xi} components values are based on workspace analysis. The optimization constraints are the kinematics parallel robot characteristics previously mentioned and the sphere radius r. The sphere location is identified by the genetic algorithm, while the optimal sphere radius r is obtained through the golden search procedure. The GAs parameters are presented in Table 6.3, which are selecting based on previous attempts. The Fig. 6.8 presents the objective function convergence, which is associated with the best individual value of each generation.

Table 6.3 – Genetic Algorithm Operation Parameters
.

Population:	80
Generations:	100
Crossover (Pc):	20
Mutation (Pm):	10

Source: Author

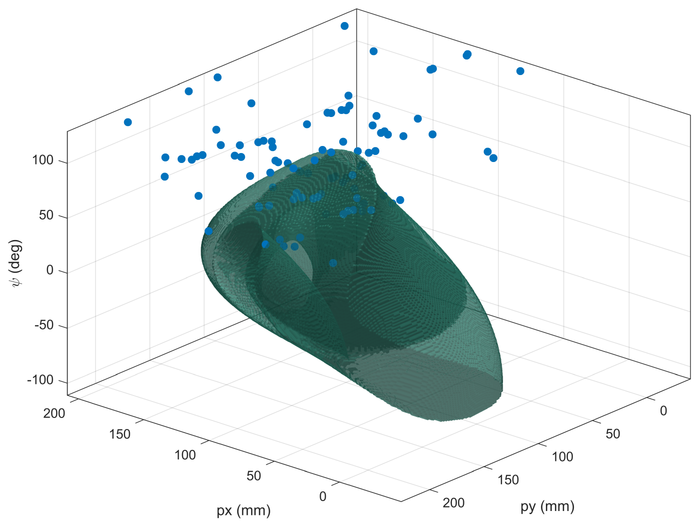
Figure 6.8 – Objective Function Convergence



Source: Autor

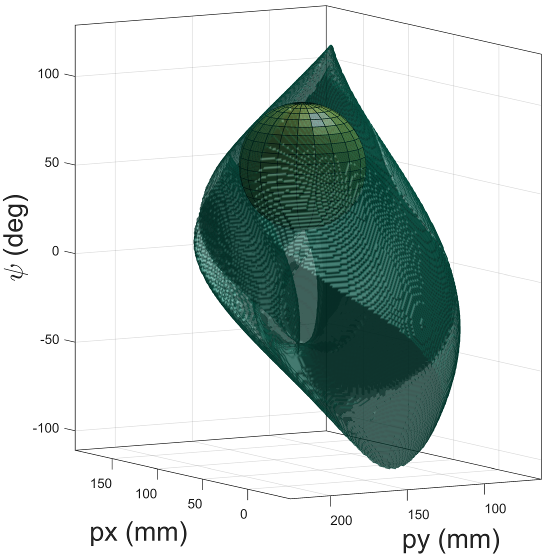
The individuals convergence is shown in Fig. 6.9. For this case the optimal solution is located at point $(60.0363mm,\ 155.2065mm,\ 59.7220°)$, $i.e.$ the sphere origin. The maximal sphere within the workspace is shown in Fig. 6.10, and its radius is $R = 35.6930$. It means that the platform may travel $35.6930mm$ in the plane $x - y$ and reach orientation $\psi = 35.6930°$ from the sphere origin without falling into singularities.

Figure 6.9 – GAs Individuals Searching An Optimal Solution



Source: Autor

Figure 6.10 – Maximal Sphere within Workspace



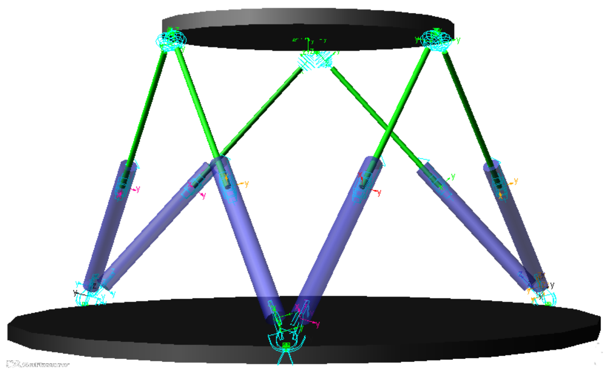
Source: Autor

Chapter 7

Stewart-Gough Platform Orientation Workspace Analysis

The spatial parallel robot used in the proposed analyze is an S-G platform with 6-Dof is shown in Fig. 7.1. Where each leg is connected at points A_i in the base and B_i in the platform by universal and spherical joints respectively (see Fig. 7.2). It should be noted that in the proposed study some assumptions have been considered like all the parallel robot links are rigid, each actuador axes passes through the respective joint centers and the parallel robot home orientation is given by $(\varphi = 0, \vartheta = 0, \psi = 0)$.

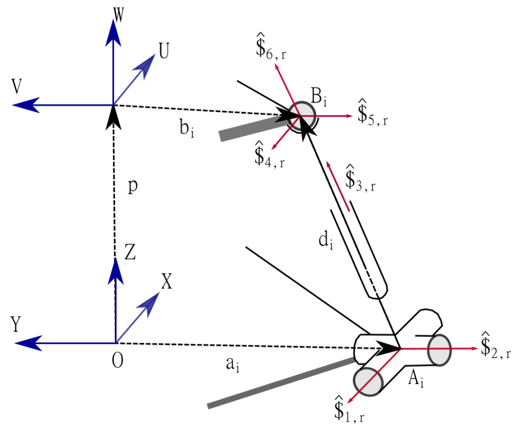
Figure 7.1 – Stewart-Gough Platform in MSC ADAMS/View



Source: Autor

Let O_1, O_2, O_3 be the fixed coordinate systems, which are located at centroid 0, on the base, at the centroid P, on the platform and at the ith point A_i, in each limb, respectively. In the coordinate system 0_3, the z_i axis is located in direction from A_i to B_i , the y_i-axis is parallel to the cross product of two unit vectors along the z_i and z axes, the x_i-axis is defined by the right-hand rule.

Figure 7.2 – Stewart-Gough Platform ith Leg Configuration



Source: (ABEDINNASAB et al., 2012)

Universal joints at point A_i and spherical joints at point B_i lie on the plane $X-Y$ and $U-V$ respectively. Due to this, it is possible to define two position vectors, the first vector a_i describe the Ai position on fixed base frame 0_1, while the second vector ${}^2b_i$ describe the B_i position on the platform frame, which may be described as.

$$a_i = \begin{bmatrix} Rcos(\theta_i) \\ Rsin(\theta_i) \\ 0 \end{bmatrix}, {}^2b_i = \begin{bmatrix} rcos(\theta_i + \theta_o) \\ rsin(\theta_i + \theta_o) \\ 0 \end{bmatrix}; \quad i = 1 \; to \; 3, \tag{7.0.1}$$

where $\theta_o = 60^o$, R is the base radius and r is the platform radius. On the other hand, the vector P shown in Fig.7.2 describes the platform location on the fixed frame. It can be written as

$$P = \begin{bmatrix} x & y & z \end{bmatrix}. \tag{7.0.2}$$

For mapping between XYZ and UVW frames, the orientation matrix ${}^1R_2$ involving the RPY Angles $(\varphi, \vartheta, \psi)$ is used (see Eq.(2.5.40)).

7.1 Reciprocal screw

For the ith KCs configuration shown in Fig.7.2, the reciprocal screw passes through the universal joint center and the spherical joint center. Therefore, the reciprocal screw is a force wrench that only performs work on the actuated joint $(prismatic\,joint)$, as it is explained in $chapter\ 3$. Thus the line vector $\mathbf{S}_r$ that described the reciprocal wrench direction may be expressed as vectorial operation.

$$s_r = \frac{p + b_i - a_i}{\|AB_i\|}, \tag{7.1.3}$$

where $b_i = {}^1R_2\, {}^2b_i$ and a_i denote the spherical and Universal joints position respect the fixed frame in the ith leg. The vector P was previously explained, and $\|AB_i\|$ is the leg length, described as,

$$\|AB_i\| = |p + b_i - a_i| \,. \tag{7.1.4}$$

Then the unitary reciprocal wrench associated to each kinematic chain is a unit screw with zero-pitch $(h = 0)$, it may be written as

$$\hat{\$}_r = \begin{bmatrix} s_r \\ s_{ro} \times s_r \end{bmatrix}. \tag{7.1.5}$$

7.2 Screw-Based Jacobian

For the ith KCs configuration shown in Fig.7.1(b) the relation between the joints twits and the platform twist may be extended as (JAYAKRISHNA; BABU,)

$$\$_p = \dot{\theta}_{(1,i)}\hat{\$}_{1,i} + \dot{\theta}_{(2,i)}\hat{\$}_{2,i} + \dot{d}_{(3,i)}\hat{\$}_{3,i} + \dot{\theta}_{(4,i)}\hat{\$}_{4,i} + \dot{\theta}_{(5,i)}\hat{\$}_{5,i} + \dot{\theta}_{(6,i)}\hat{\$}_{6,i}. \tag{7.2.6}$$

The unit twist direction may be described by the Eq 7.1.3 , $i.e.$ $\mathbf{S}_r = \mathbf{S}$. It should be noted that the wrench is reciprocal to passive joints. Therefore, if premultiply the Eq.(7.2.6) for the unit reciprocal wrench $\hat{\$}_r$

$$\hat{\$}_{r,i} \circ \$_p = \hat{\$}_{r,i} \circ \hat{\$}_{3,i}\dot{d}_{(3,i)}, \tag{7.2.7}$$

where,

$$\hat{\$}_{r,i}^T\hat{\$}_{3,i} = 1. \tag{7.2.8}$$

Extended this analysis for the other legs and writing in the matrix form

$$\begin{bmatrix} s_{ro,1} \times s_{r,1} & s_{r,1} \\ \vdots & \vdots \\ s_{ro,6} \times s_{r,6} & s_{r,6} \end{bmatrix} \$_p = \begin{bmatrix} I \end{bmatrix} \begin{bmatrix} \dot{d}_{(3,1)} \\ \vdots \\ \dot{d}_{(3,6)} \end{bmatrix}, \tag{7.2.9}$$

or

$$J_x \left[\$_p\right] = J_q \left[\dot{d}\right]. \tag{7.2.10}$$

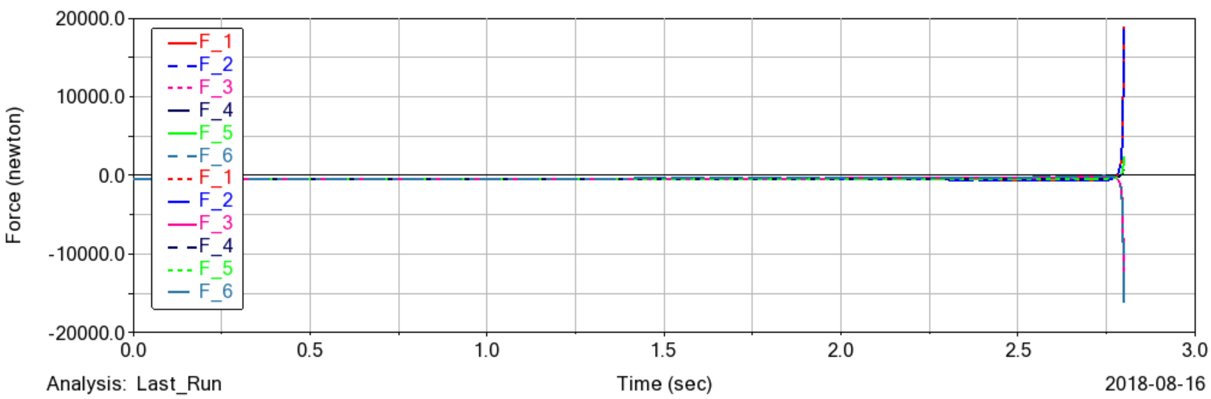
7.3 Constraints formulation

This robot is inverse singularity free due to the J_q matrix is an identity matrix see Eq.(7.2.9). Hence the robot mobility is only constrained by the direct singularity and the geometrical constrains.

Direct Singularity An algorithm in MATLAB was applied to aim at measuring the robot direct singularity closeness λ. For this, the algorithm calculates the wrench value for each instantaneous PRs position. Hesselbach et al. (2005) determinated that the direct singularity takes place when $\sqrt{\lambda}$ falls under 0.03, for a Hexa parallel robot. Therefore, to identify the S-G index value is developed a kinematic model in ADAMS. Where a number of dynamics simulations are made. The simulations consist in rotate the platform about the x, y and z axes with a load F and a moment τ on the platform in a time t, until the parallel robot fall in a direct singularity.

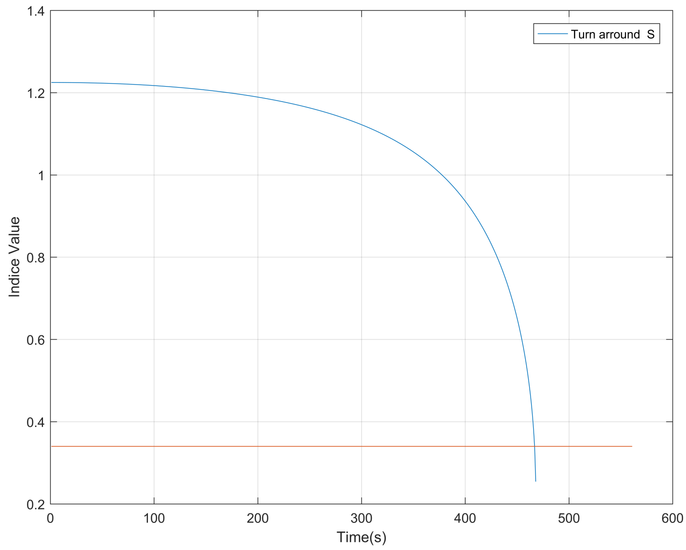
The ADAMS software measures the reaction force in each leg to detect where the parallel robot gets in a direct singularity. It is possible that the leg reaction force tends to infinity in this singularity type, with that in mind the Fig. 7.3 shown the reaction force meassured for a platform turn around any axis S.

Figure 7.3 – Stewart-Gough Platform legs reaction force measure in MSC ADAMS/View



Source: Autor

Figure 7.4 – Masure index associated to the rotation arround S



Source: Autor

The index limit corresponds to the point where the reaction leg force suffers a high increase, which is named breaking point. This point is determined based on the ADAMS measures in each time step $\triangle t$, its value is obtained measuring the direct kinematic indice for the platform orientations associated with the breaking point. Therefore, the constraint equation for the direct kinematic singularity may be written as,

$$\sqrt{\lambda_{min}} > 0.34. \tag{7.3.11}$$

Geometrical Constraints The mechanical constraints considered in this analysis are: (a) Prismatic joint limit, (b) Kinematic chains interference.

$$\rho_{min} \leq l_{pi} \leq \rho_{max}, \tag{7.3.12}$$

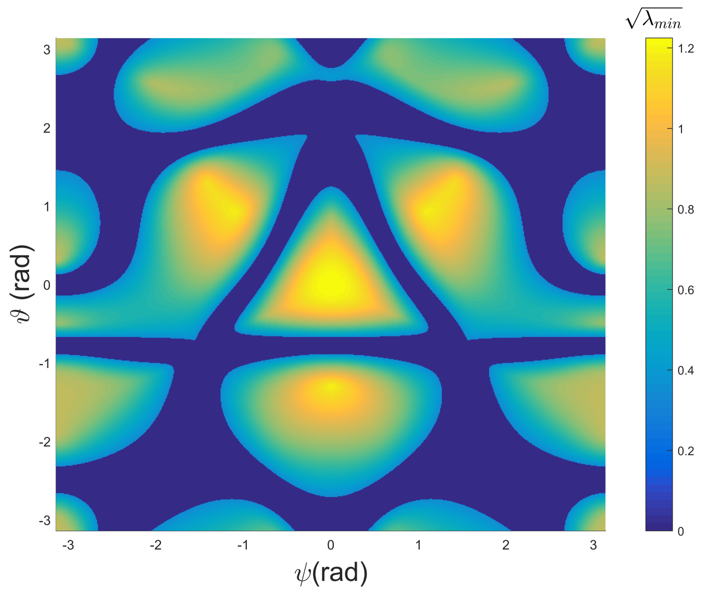
$$d_{ij} \geq D. \tag{7.3.13}$$

where l_{pi} is the motion range for the ith prismatic joint shown in Fig. 7.1(b). $\rho_{min} = 0.7m$ and $\rho_{max} = 1.4229m$ are the minimum and maximum lengths of actuated joints, d_{ij} is the distance between the ith link and the jth link $(i \neq j)$, $D = 0.03m$ is the diameter of each link.

7.4 Workspace Analysis

In this section, the 6 U$\underline{P}$S orientation workspace is presented. Considering the kinematic and the geometric constraints mentioned above. In the proposed analysis the base radius $(r = 1m)$, the platform radius $(R = 0.5m)$ and a platform initial orientation $(\varphi = 0, \vartheta = 0, \psi = 0)$ are regarded. The analysis is reduced to the achievable orientations in the plane $y - x$ due to the complex geometry of the orientation workspace.

Figure 7.5 – Orientation Workspace Associated to Direct Kinematic Index $(\sqrt{\lambda_{min}})$

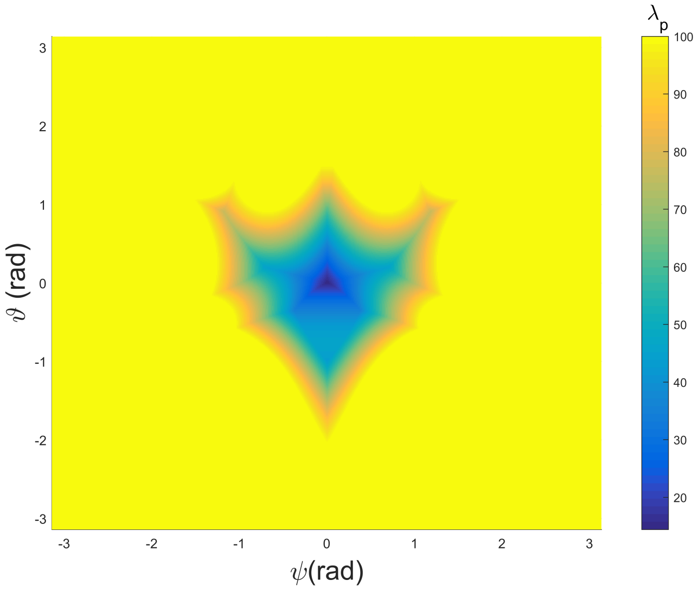


Source: Autor

As the first case study, a fixed platform position $p = [0\ 0\ 0.5]m$ is considered. In Fig. 7.5 is shown the direct kinematic index value associated to each platform orientation $(0, \vartheta, \psi)$, where the infeasible regions are related to the blue regions while the remains regions are regarded as feasibles $i.e.$ the index value is higher than 0.34. Several existing feasible regions where the platform may move as shown in Fig. 7.5, which means that travel between regions in the same plane is possible if an appropriate platform path or platform configuration is established. It is important to note that the term “platform configuration” is related to the different orientations that the platform achieves within the orientation workspace. In addition to direct singularity index, others indices that associate the geometrical constraints with the orientation workspace boundaries are established: prismatic joint λ_p, legs collision λ_c, and passive joint λ_y.

The prismatic joint index represents the higher proximity percentage from the sliding link effector positions to the physical joint limits. Where 100% means that the joint reached its physical limit. For the stewart case this measure restricts the parallel robot orientation workspace approximately 70% in relation to the direct kinematic index, in the case of plane analysis to a central feasible region as shown in Fig. 7.6. Where the infeasible regions are yellow, while the remains regions are regarded as feasibles.

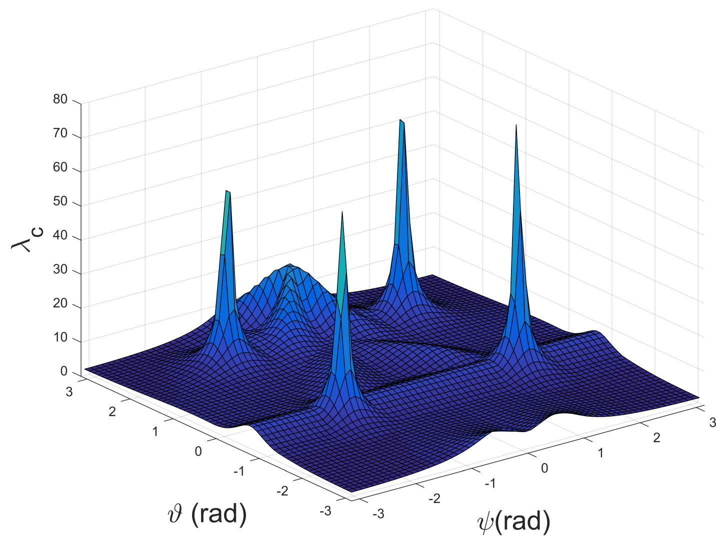
Figure 7.6 – Orientation Workspace Associated to Prismatic joint index (λ_p)



Source: Autor

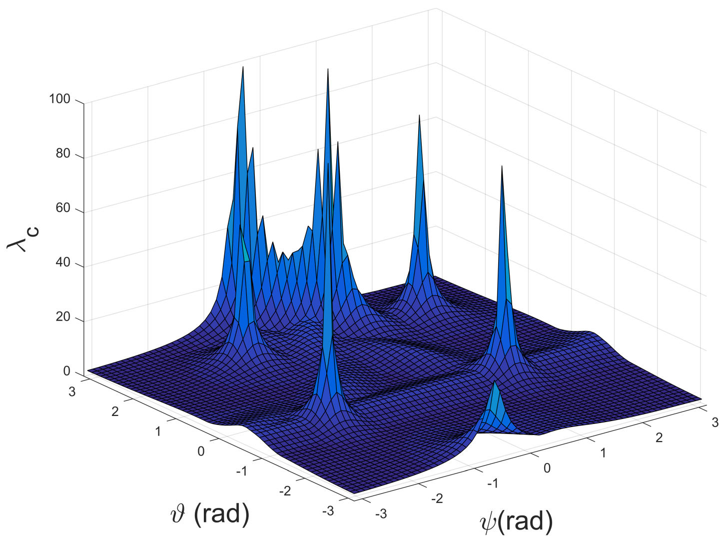
The leg collision index indicates how close the limbs are to impact with each other. This collision is considered when the index value is 100%. For the planar study shown in Fig. 7.7 the higher index is 50%, it means that there are no limitations related to the collision index. In this case, the collision index is limited by the prismatic joint indices and the direct kinematic index. Therefore the orientation workspace in the plane $\psi = 0.0$ is initially restricted by these constraints. The collision analysis is always considered because the workspace could be limited in other planes or in the space, $e.g.$ in the plane $\psi = 0.3$ index value increases to 98% as shown Fig. 7.8.

Figure 7.7 – Orientation workspace associated to legs collision (λ_c), $\psi = 0.0$



Source: Autor

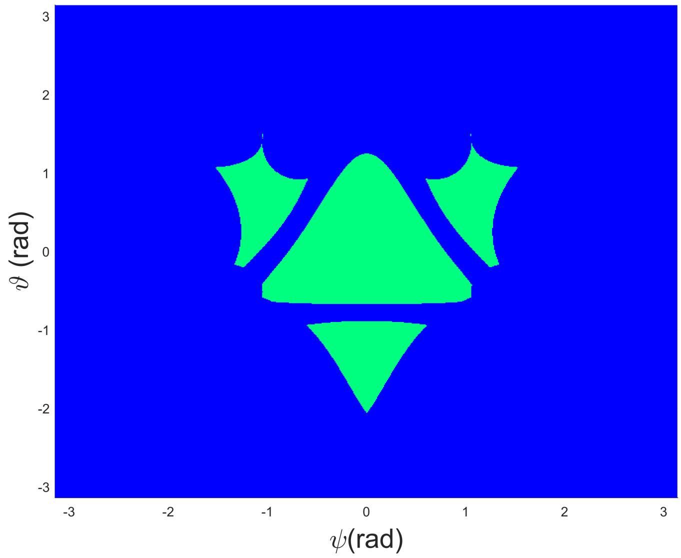
Figure 7.8 – Orientation workspace associated to legs collision (λ_c), $\psi = 0.3$



Source: Autor

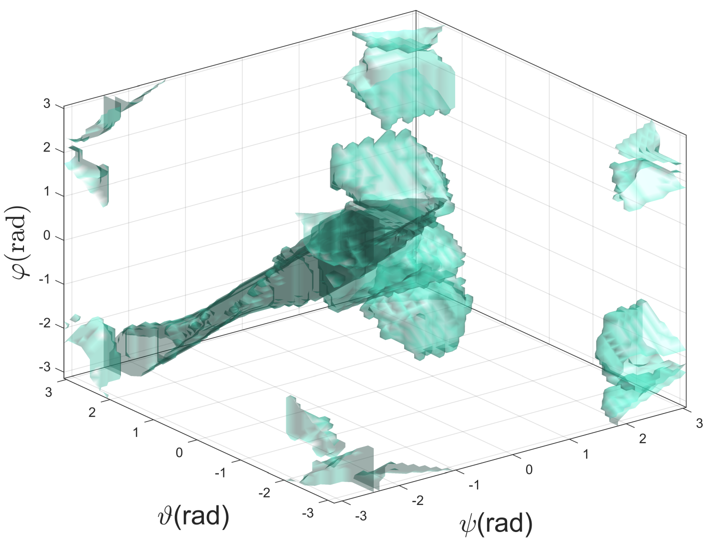
The orientation workspace, shown in Fig. 7.9 exhibits different feasible regions (in green) in which the parallel robot operates without violating the kinematics and prismatic joint contraint, see Fig. 7.10. To achieve major rotations in some specific directions, the platform may be within one of these green regions.

Figure 7.9 – Orientation workspace without passive joints analysis



Source: Autor

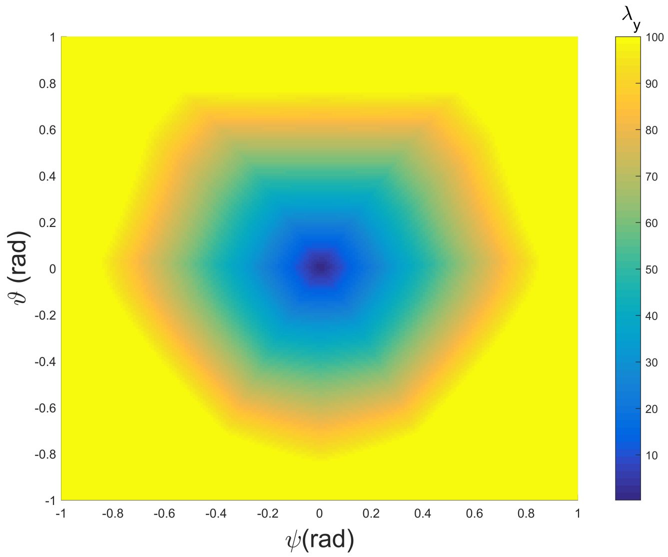
Figure 7.10 – Three-dimensional orientation workspace without passive joints analysis



Source: Autor

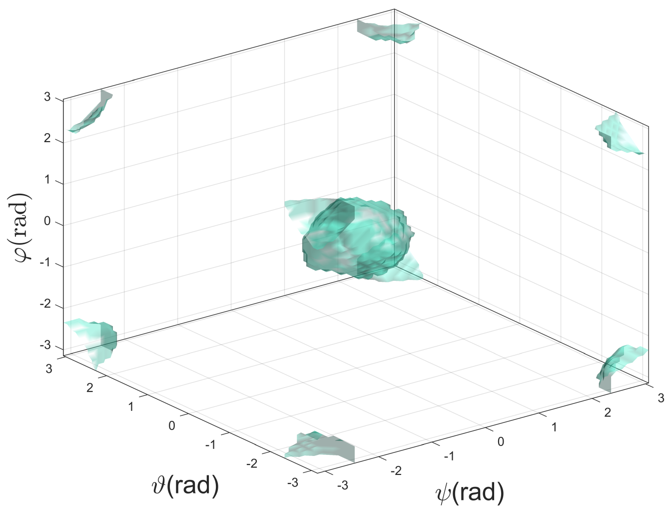
The passive joint index is related to the non-actuated joint motion study, which presents physical restricted motion similar to the actuated joints (prismatic). In the S-G platform case, it refers to allowed movement in universal and spherical joints. The implemented methodology is explained in Section 4.2. The workspace related to passive joints is presented in Fig 7.11. Taking the passive joint analysis into account, the orientation workspace is reduced to a unique feasible region (see Fig 7.12). In this case the β value is based on the Bonev and Ryu (2001) researches.

Figure 7.11 – Orientation workspace associated to passive join index with (λ_y), $\beta \leq 50°$



Source: Autor

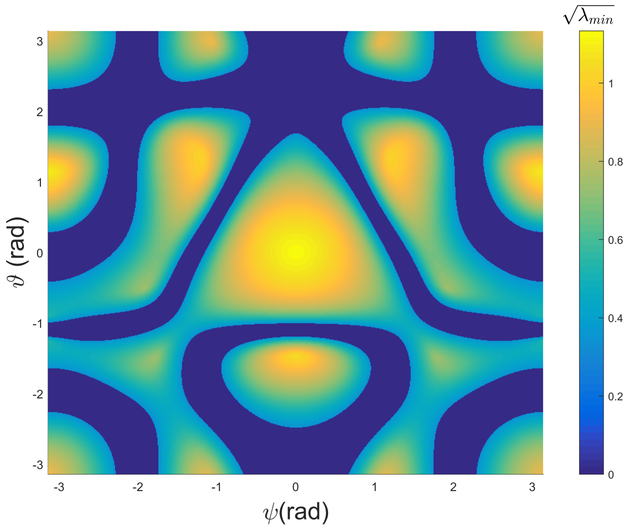
Figure 7.12 – Three-dimensional orientation workspace with passive joints analysis



Source: Autor

For the second case study, a fixed platform position $p = [0\ 0\ 1]m$ is considered. The orientation workspace associated with the direct kinematic index is shown in Fig. 7.13. In this case, the home position is the same than case 1 and the regions related to the home position is not completely delimited by the infeasible regions unlike to the regions shown in Fig. 7.5. Therefore, the platform could be access to different platform configuration maps varying the platform $Z - position$. Therefore, the parallel robot travels between the orientation subregions without falling in singularity.

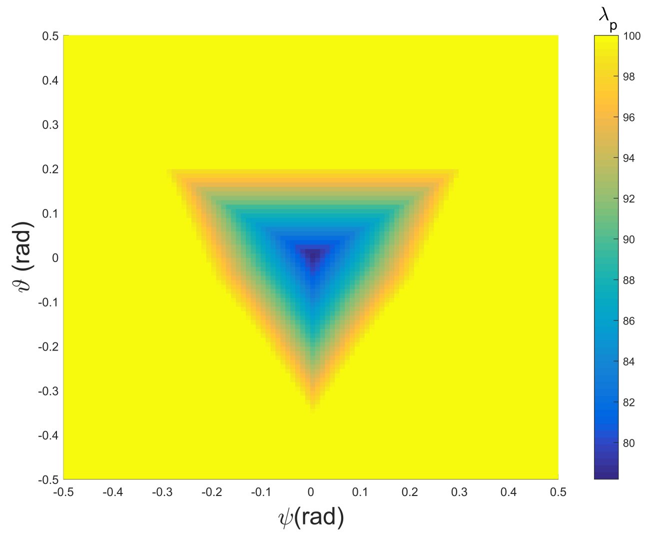
Figure 7.13 – Orientation workspace associated to direct kinematic index $(\sqrt{\lambda_{min}})$



Source: Autor

The orientation workspace associated with the prismatic joint index is shown in Fig. 7.14. This indice related the orientation workspace with the ith prismatic joint mobility range λ_p. Therefore, the reduce orientation workspace presents in Fig. 7.14 indicate that the prismatic joint it is close to its limit ρ_{max}.

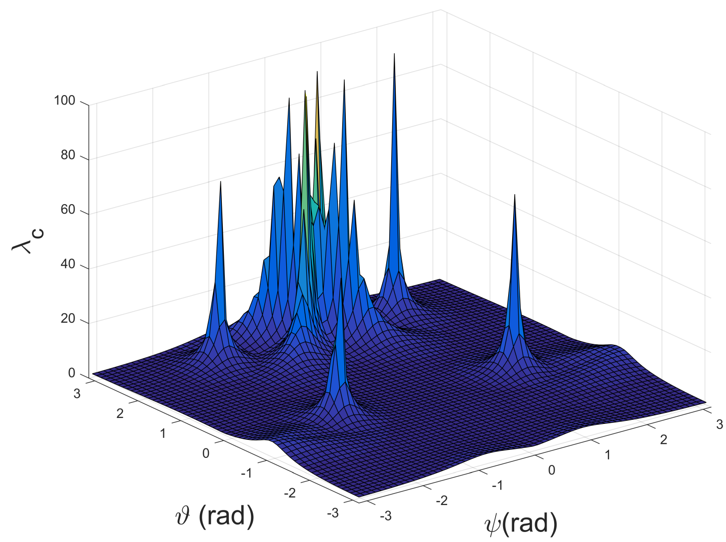
Figure 7.14 – Orientation Workspace Associated to Prismatic joint index (λ_p)



Source: Autor

The translational platform movement in $z-$axis does not affect the orientation workspace associated with passive join index due to parallel robot geometric. The orientation workspace associated with legs collision is shown in Fig. 7.15. In this study, there are a spesific regions where the probability that the legs collision occur is 80%. However, to the analyze position p the critical index is associated with the prismatic joint mobility, which restricts the regions where the index associated with the collision achieve its maximum values.

Figure 7.15 – Orientation workspace associated to legs collision (λ_c)



Source: Autor

7.5 Workspace Optimization

As it is explained in chapter 5, the optimization workspace is related to the maximal singularity-free sphere. The sphere points coordinates are represented by mean of RPY angles, the Genetic algorithm is used to determinate the sphere location, while the golden search procedure is used to obtain the sphere radio. The optimization problem is described as:

$$\begin{aligned} maximize \qquad & f(0_{xi}) \\ subject\ to: \qquad & \beta \leq 50^{\circ} \\ & \sqrt{\lambda_{min}} > 0.34. \\ & \rho_{min} \leq l_{pi} \leq \rho_{max} \\ & d_{ij} \geq D \\ & -\pi \leq 0_{xi} \leq \pi \end{aligned} \tag{7.5.14}$$

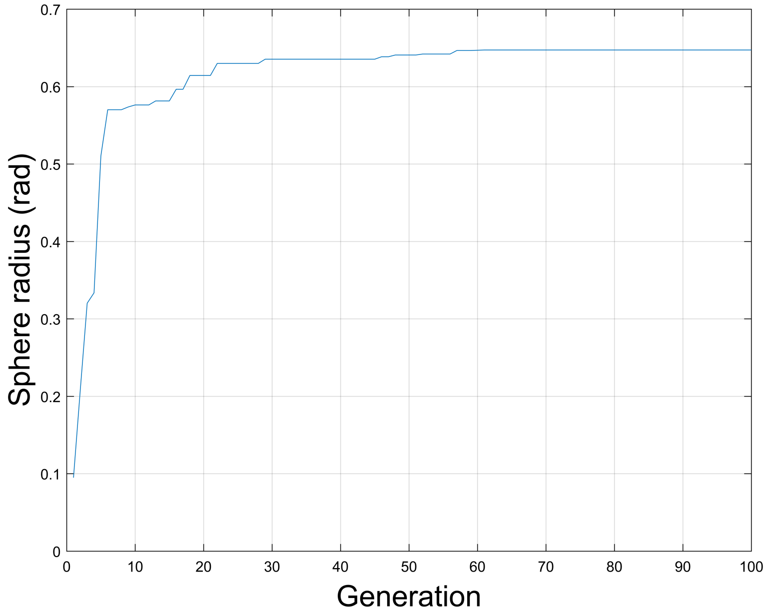
where $0_{xi} = [0_{x1},\ 0_{x2},\ 0_{x3}]$, the optimization contraints are associated with the kinematic and geometrical parallel robot charateristics mentioned below. The objective function convergence is shown Fig. 7.16. This function is associated with the best individual value of each generation. Due to the random GAs nature, there is no specific rule to select the optimization algorithm operation parameters. Thus, they are selected based on previous attempts to find an optimal solution. The GA parameters are shown in Table 7.1.

Table 7.1 – Genetic Algorithm Operation Parameters
.

Population:	80
Generations:	100
Crossover (Pc):	30
Mutation (Pm):	10

Source: Author

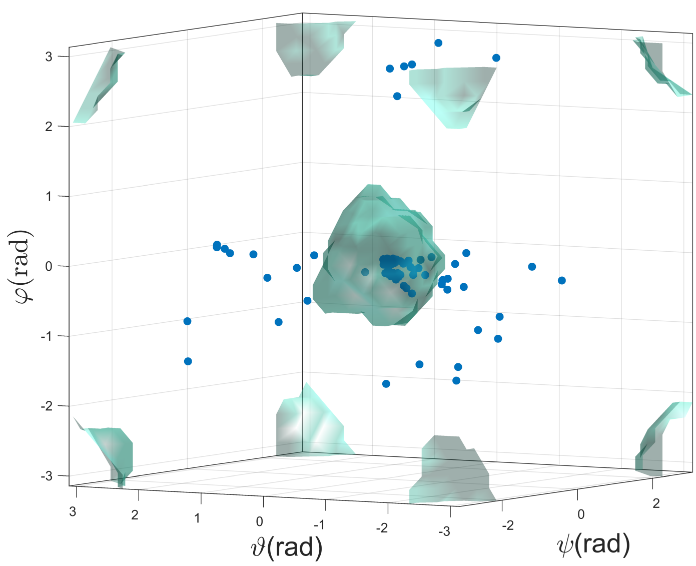
Figure 7.16 – Objective Function Convergence



Source: Autor

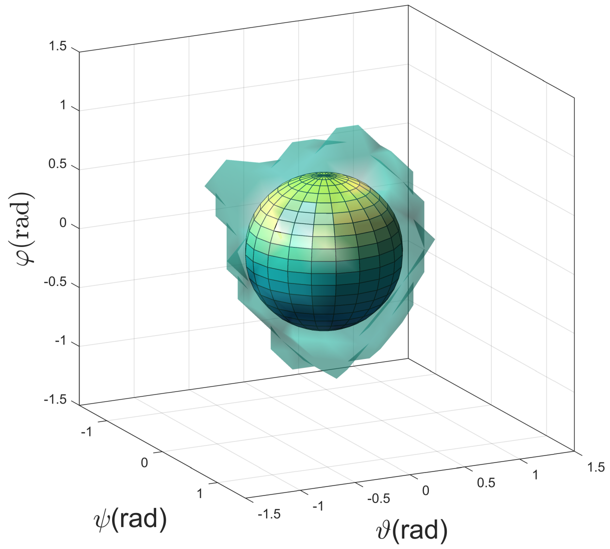
The GA individuals explore all the space detecting feasible regions, while the golden search modifies the sphere size. The individual convergences is shown in Fig. 7.17. For this case the optimal solution is located at point $(-33, 11, 28)10-4$. The maximal sphere within the orientation workspace is shown in Fig. 7.18 and its radius is $R = 0.646rad$. It means that the platform could reach these orientations $\theta_{x,y,z} \leq 0.646rad$ without falling into singularities.

Figure 7.17 – GAs Individuals Searching An Optimal Solution



Source: Autor

Figure 7.18 – Maximal Sphere within Orientation Workspace



Source: Autor

Chapter 8

Conclusions

This research focuses is to develop a methodology for modeling, calculation, mapping, and optimization of parallel robot orientation workspace. The methodology consists in finding the initial platform orientations, where it achieves their higher rotations in all directions, considering the physical and kinematical constraints, which allow knowing the workspace boundaries and its geometry. Due to the workspace complex geometry, the maximal platform orientation workspace is approximated to simplified geometry, $i.e.$ a sphere, where its origin is the initial platform orientation and its radius is the higher rotation magnitude that it may be achieved.

It is provided a method for solving the mobility problems for general parallel robot architecture based on screw theory analysis. This method takes into account the Jacobian matrices and the actuator contributions on the platform motion. Furthermore, the parallel robot physical constraint as geometrical limits are considered. In the present study, it is found a single index power measure value for the Stewart-Gough Platform. The index value is the limit between non-singular and singular subregions, which it is shown in Fig. 7.5 - 7.13. In these figures two platform positions $p_1 = [0\ 0\ 0.5]m$, and $p_1 = [0\ 0\ 1]m$ are considered, where some specific regions are not completely delimited by the infeasible regions, meaning that the platform could be acces to different platform configuration maps varying its position p, especifly the Z component. For the planar parallel robot 3-<u>R</u>RR studied in the present work is found a single index measure power. The planar parallel robot workspace related to this indice is presented in Fig. 6.5. In this case is determined that the platform may become uncontrollable in some task for a given path or to achieve some specific configurationn, which is due to there are infeasibles regions within the workspace.

In the proposed analysis only the planar parallel robot robot 3-RRR fall in the combine singularity. This is due to that unlike to the Stewart-Gough Platform the J_q is not an identity matrix. The points where the planar parallel robot fall in combine singularity can be observed in the Fig. 6.7 and corresponds to all points into the blue regions that are not whitin the planar parallel robot workspace. Therefore, the 3-RRR platform can be in direct singularity without fall in inverse singularity but the opposite is not possible.

It is presented as the orientation workspace is bounded by the parallel robot kinematics and geometrical constraints and how the indices associated to each restriction must be calculated to describe correctly the orientation workspace. The procedure to calculate these indices may be defined in different ways. Based on the results obtained in this work the indices analysis should be initiated by the passives joints index because this measure reduces significantly the orientation workspace, this consideration is aiming to reduce the computational cost.

In the proposed analysis is not considered some geometrical constraints as the collision between legs and platform. The procedure proposed in this study may be extended for the redundant parallel robot. Then the methodology used to describe the kinematic and geometrical constraints would provide a basis to futures searches associated to this parallel robot type. The algorithm must verify that kinematics and geometrical are not violating at specific points (boundaries sphere points) to determine that the sphere is within the orientation workspace. In the presents work this procedure is done iteratively, Thus the algorithm efficiency is reduced due to the computational cost. However, develop a mathematical model linked to the verification points may facilitate the optimization process.

In the workspace may exist different available regions associated with the parallel robot kinematic and geometry characteristics as shown in Fig. 7.9-7.10. Where the platform may attain higher rotations in some specific directions. These rotations are reduced due to passive joints mobility range. However, this disadvantage could be treated with a passive joint optimization, where the passive joints axis angle β is the parameter to be optimized.

Bibliography

ABBASNEJAD, G.; DANIALI, H. M.; KAZEMI, S. A new approach to determine the maximal singularity-free zone of 3-rpr planar parallel manipulator. *Robotica*, Cambridge Univ Press, v. 30, n. 06, p. 1005–1012, 2012.

ABEDINNASAB, M. H.; YOON, Y.-J.; ZOHOOR, H. Exploiting higher kinematic performance-using a 4-legged redundant pm rather than gough-stewart platforms. In: *Serial and Parallel Robot Manipulators-Kinematics, Dynamics, Control and Optimization*. [S.l.]: InTech, 2012.

ANGELES, J. *Fundamentals of robotic mechanical systems: theory, methods, and algorithms*. [S.l.]: Springer Science & Business Media, 2013. v. 124.

ARORA, J. *Introduction to optimum design*. [S.l.]: Elsevier, 2004.

BALL, R. S. *A Treatise on the Theory of Screws*. [S.l.]: Cambridge university press, 1998.

BANDYOPADHYAY, S.; GHOSAL, A. An eigenproblem approach to classical screw theory. *Mechanism and Machine Theory*, Elsevier, v. 44, n. 6, p. 1256–1269, 2009.

BAYANI, H. et al. On the determination of the maximal inscribed ellipsoid in the wrench-feasible workspace of the cable-driven parallel robots. In: IEEE. *Robotics and Mechatronics (ICRoM), 2014 Second RSI/ISM International Conference on*. [S.l.], 2014. p. 422–427.

BEN-HORIN, P.; SHOHAM, M. Singularity condition of six-degree-of-freedom three-legged parallel robots based on grassmann-cayley algebra. *IEEE transactions on robotics*, IEEE, v. 22, n. 4, p. 577–590, 2006.

BOHIGAS, O.; MANUBENS, M.; ROS, L. A complete method for workspace boundary determination on general structure manipulators. *IEEE Transactions on Robotics*, IEEE, v. 28, n. 5, p. 993–1006, 2012.

BOHIGAS, O.; MANUBENS, M.; ROS, L. A linear relaxation method for computing workspace slices of the stewart platform. *Journal of mechanisms and robotics*, American Society of Mechanical Engineers, v. 5, n. 1, p. 011005, 2013.

BOHIGAS, O.; MANUBENS, M.; ROS, L. Singularities of non-redundant manipulators: A short account and a method for their computation in the planar case. *Mechanism and Machine Theory*, Elsevier, v. 68, p. 1–17, 2013.

BONEV, I. A. Analysis and design of 6-dof 6-prrs parallel manipulators. *MasterÕs Thesis, Kwangju Institute of Science and Technology, Kwangju*, 1998.

BONEV, I. A. *Geometric analysis of parallel mechanisms*. [S.l.]: Citeseer, 2002.

BONEV, I. A.; RYU, J. A new approach to orientation workspace analysis of 6-dof parallel manipulators. *Mechanism and machine theory*, Elsevier, v. 36, n. 1, p. 15–28, 2001.

CAMPOS, A. Cinemática diferencial de manipuladores empregando cadeias virtuais. *Pos-Grad em Eng. Mec*, 2004.

CHRISTENSEN, S. Stewart-gough platform. 2014.

COSTE, M.; MOUSSA, S. On the rationality of the singularity locus of a gough–stewart platform—biplanar case. *Mechanism and Machine Theory*, Elsevier, v. 87, p. 82–92, 2015.

CRAIG, J. J. *Robótica. 3ª edição*. [S.l.]: São Paulo: Editora Pearson, 2012.

DAVIDSON, J. K.; HUNT, K. H. *Robots and screw theory: applications of kinematics and statics to robotics*. [S.l.]: Oxford University Press on Demand, 2004.

GAO, Z.; ZHANG, D. Workspace representation and optimization of a novel parallel mechanism with three-degrees-of-freedom. *Sustainability*, Molecular Diversity Preservation International, v. 3, n. 11, p. 2217–2228, 2011.

GAO, Z.; ZHANG, D.; GE, Y. Design optimization of a spatial six degree-of-freedom parallel manipulator based on artificial intelligence approaches. *Robotics and Computer-Integrated Manufacturing*, Elsevier, v. 26, n. 2, p. 180–189, 2010.

HESSELBACH, J. et al. Direct kinematic singularity detection of a hexa parallel robot. In: IEEE. *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*. [S.l.], 2005. p. 3238–3243.

HUANG, Z.; LI, Q.; DING, H. *Theory of parallel mechanisms*. [S.l.]: Springer Science & Business Media, 2012. v. 6.

JAYAKRISHNA, S. R.; BABU, G. S. Dexterity indices of 6-ups parallel manipulator.

JIA, Y.-B. Nonlinear optimization. 2017.

JIANG, Q. *Singularity-free workspace analysis and geometric optimization of parallel mechanisms*. Tese (Doutorado) — Université Laval, 2008., 2008.

JIANG, Q.; GOSSELIN, C. M. Determination of the maximal singularity-free orientation workspace for the gough–stewart platform. *Mechanism and Machine Theory*, Elsevier, v. 44, n. 6, p. 1281–1293, 2009.

JO, D.-Y.; HAUG, E. J. *Workspace analysis of closed-loop mechanisms with unilateral constraints*. [S.l.]: Center for Simulation and Design Optimization and Department of Mechanical Engineering, University of Iowa, 1989.

KANAAN, D. et al. Singularity analysis of lower mobility parallel manipulators using grassmann–cayley algebra. *IEEE Transactions on Robotics*, IEEE, v. 25, n. 5, p. 995–1004, 2009.

KARIMI, A.; MASOULEH, M. T.; CARDOU, P. Singularity-free workspace analysis of general 6-ups parallel mechanisms via convex optimization. *Mechanism and Machine Theory*, Elsevier, v. 80, p. 17–34, 2014.

KELAIAIA, R.; COMPANY, O.; ZAATRI, A. Multiobjective optimization of a linear delta parallel robot. *Mechanism and Machine Theory*, Elsevier, v. 50, p. 159–178, 2012.

KONG, X.; GOSSELIN, C. M. *Type synthesis of parallel mechanisms*. [S.l.]: Springer Publishing Company, Incorporated, 2007.

LI, H. *The analytical singularity locus equation and the determination of singularity-free zones in the workspace of the general Gough-Stewart platform*. [S.l.]: Université Laval, 2005.

LI, W.; ANGELES, J. Full-mobility 3-ccc parallel-kinematics machines: Forward kinematics, singularity, workspace and dexterity analyses. *Mechanism and Machine Theory*, Elsevier, v. 126, p. 312–328, 2018.

MARTÍNEZ, J. On the basis of workspaces of robotic manipulators (part 1). 2013.

MCCALL, J. Genetic algorithms for modelling and optimisation. *Journal of Computational and Applied Mathematics*, Elsevier, v. 184, n. 1, p. 205–222, 2005.

MERLET, J.-P. *Parallel robots*. [S.l.]: Springer Science & Business Media, 2006. v. 128.

MONSARRAT, B.; GOSSELIN, C. M. Workspace analysis and optimal design of a 3-leg 6-dof parallel platform mechanism. *IEEE Transactions on Robotics and Automation*, IEEE, v. 19, n. 6, p. 954–966, 2003.

MURRAY, R. M. et al. *A mathematical introduction to robotic manipulation*. [S.l.]: CRC press, 1994.

POTT, A. Efficient computation of the workspace boundary, its properties and derivatives for cable-driven parallel robots. In: *Computational Kinematics*. [S.l.]: Springer, 2018. p. 190–197.

POTTMANN, H.; PETERNELL, M.; RAVANI, B. Approximation in line space—applications in robot kinematics and surface reconstruction. In: *Advances in Robot Kinematics: Analysis and Control*. [S.l.]: Springer, 1998. p. 403–412.

RAO, S. S.; RAO, S. S. *Engineering optimization: theory and practice*. [S.l.]: John Wiley & Sons, 2009.

SAPUTRA, V. B.; ONG, S.-K.; NEE, A. Y. A swarm optimization approach for solving workspace determination of parallel manipulators. *Robotica*, Cambridge Univ Press, v. 33, n. 03, p. 649–668, 2015.

SHINTEMIROV, A.; NIYETKALIYEV, A.; RUBAGOTTI, M. Numerical optimal control of a spherical parallel manipulator based on unique kinematic solutions. *IEEE/ASME Transactions on Mechatronics*, IEEE, v. 21, n. 1, p. 98–109, 2016.

SICILIANO, B. et al. *Robotics: modelling, planning and control*. [S.l.]: Springer Science & Business Media, 2010.

SIMONI, R. et al. Contribuições para a enumeração e para a análise de mecanismos e manipuladores paralelos. 2010.

ST-ONGE, B. M.; GOSSELIN, C. M. Singularity analysis and representation of the general gough-stewart platform. *The International Journal of Robotics Research*, SAGE Publications, v. 19, n. 3, p. 271–288, 2000.

STAN, S.-D. et al. Genetic algorithms for workspace optimization of planar medical parallel robot. In: *IEEE International Conference on Emerging Trends in Computing, ICETIC*. [S.l.: s.n.], 2009. p. 8–10.

SUN, T. et al. Optimal design of a parallel mechanism with three rotational degrees of freedom. *Robotics and Computer-Integrated Manufacturing*, Elsevier, v. 28, n. 4, p. 500–508, 2012.

SYAHPUTRA, R. Distribution network optimization based on genetic algorithm. *Journal of Electrical Technology UMY*, v. 1, n. 1, p. 1–9, 2017.

TSAI, L.-W. *Robot analysis: the mechanics of serial and parallel manipulators*. [S.l.]: John Wiley & Sons, 1999.

VOGLEWEDE, P. A. *Measuring closeness to singularities of parallel manipulators with application to the design of redundant actuation*. Tese (Doutorado) — Georgia Institute of Technology, 2004.

WANG, X. et al. Singularity analysis and treatment for a 7r 6-dof painting robot with non-spherical wrist. *Mechanism and Machine Theory*, Elsevier, v. 126, p. 92–107, 2018.

WEILE, D. S.; MICHIELSSEN, E. Genetic algorithm optimization applied to electromagnetics: A review. *IEEE Transactions on Antennas and Propagation*, IEEE, v. 45, n. 3, p. 343–353, 1997.

ZHAO, J. et al. Geometrical method to determine the reciprocal screws and applications to parallel manipulators. *Robotica*, Cambridge Univ Press, v. 27, n. 06, p. 929–940, 2009.

ZLATANOV, D.; GOSSELIN, C. M. Singularity analysis of 3-dof planar parallel mechanisms via screw theory. *ASME Journal of Mechanical Design*, v. 125, p. 573–581, 2003.

Appendix A

Parallel Robots Kinmatics Algorithms in Matlab

A.1 S-G Platform Algorithm

```
function [Di,med,Dmi,Cj]=res2(z,y,x)
deg=pi/180; % convert degree to radiams
np=6; % Legs number
RB=1; % Fixed base radius (mm)
RM=0.5; % Moving platform radius (mm)
SA=50*deg; % Spherical joint rotation angle (radiam)
UA=50*deg; % Universal joint rotation angle (radiam)
CD=0.03; % Cylinder diameter (mm)
UP=1.4; % Upper prismatic joint limit (mm)
LP=0.7; % Lower prismatic joint limit (mm)

%% Robot configuration
tb=0; % angle to locate the points on fixed platform (degree)
ta=0; % angle to locate the points on moving platform (degree)
% Angles to locate the legg's points
% 1 1 2 2 3 3
Ab=[ 60 60+tb 180 180+tb 300 300+tb]*deg; % Points distributions mv platform (0-60)
% 1 2 2 3 3 1
Aa=[0+ta 120 120+ta 240 240+ta 0]*deg; % Points distributions mv platform (0-60)
% fixed base points
a=zeros(3,6);
a(1:2,:)=[RB*cos(Aa(1:6));RB*sin(Aa(1:6))]; % This matrix containts the points coordinates
% moving platform'points
b=zeros(3,6);
b(1:2,:)=[RM*cos(Ab(1:6));RM*sin(Ab(1:6))]; % This matrix containts the points coordinates
% The moving platform position vector
px(1,1:6)=0; % x position (mm)
py(1,1:6)=0; % y position (mm)
pz(1,1:6)=1;%0.5; % z position (mm)
p=[px; py; pz];
c=p(:,:)+b(:,:)-a(:,:); % p(:,:)+Rmm*b(:,:)-a(:,:) but Rmm(0,0,0)=identity
%% passive Joints symetric axis
ES=[c(:,1)/norm(c(:,1)) c(:,2)/norm(c(:,2)) c(:,3)/norm(c(:,3)) c(:,4)/norm(c(:,4)) c(:,5)/norm(c(:,5)) c(:,6)/norm(c(:,6))];

G=zeros(np,np);
```

```
cj=zeros(2,1);
cjA=zeros(np,1);
cjB=zeros(np,1);
UN=zeros(3,np);
lon=zeros(np,1);
si=zeros(3,np);
d=zeros(3,6);

%% orientacion y posicion de la plataforma movil
    Ro=Rpy(z,y,x); %Roll-pitch-yaw orientation;

%% Screw and Screw operations
    for j=1:np
        bi=Ro*b(:,j);
        UN(:,j)=Ro*ES(:,j);
        d(:,j)=p(:,1) + Ro*b(:,j) - a(:,j);
        si(:,j)= d(:,j)/norm(d(:,j));
        cjA(j,1)= real(acos(ES(:,j)'*si(:,j)));
        cjB(j,1)= real(acos(UN(:,j)'*si(:,j)));
        gra=[cruz(bi,si(:,j))' si(:,j)']'*[cruz(bi,si(:,j))' si(:,j)'];
        G=G+gra;
        lon(j,1)=norm(d(:,j));
    end

%% Constrains calculation
% Passive joint
    cj(1,1)=(max(cjB)/SA)*100; % Moving platform passive joints Angle
    cj(2,1)=(max(cjA)/UA)*100; % Fixed base passive joints Angle
    Cj=max(cj); % Higer Angle reached on a passive joint for given orientation and position
% Prismatic joints limits
   l=(UP-LP)/2; % aviable prismatic region
   lomin=min(lon); % Minimum prismatic joint position for a given orientation and position
   lomax=max(lon); % Higher prismatic joint position for a given orientation and position
   m(1,1)=abs((lomin-(LP+l))/l)*100; % Minimum prismatic joint position (porcentagem)
   m(2,1)=abs((lomax-(LP+l))/l)*100; % Maximum prismatic joint position (porcentagem)
   med=max(m);
% Direct kinematic
    D=eye(6,6);
    D(1:3,1:3)=zeros;
    Gm=G\D;
    eval=eig(Gm);
    max_auto= max(max(real(eval)));
    lam=1/max_auto;
    Di=sqrt(lam);
% Colition
    Co=dism(d)/CD;
    Dmi=100/Co;
end
```

A.2 3-RRR Platform Algorithm

```
function [Di,I]=res(Px,Py,tz)
%% Constans
BC=zeros(3,3);
AB=zeros(3,3);
IR=zeros(2,2);
Jq=zeros(3,3);
sr1=zeros(3,3);
s=[0 0 1];
%% Base points locations Ai=[x y](mm)
Lb=260 ; %Base length
%frame system
A1=[0 0 0];
A2=[Lb/1.45 0 0];
A3=[Lb/3 ((sqrt(3))/2)*Lb 0];
%% Platform points location Bi=[x y]
Lp=85.73; %platform length
%Moving system
C1=[-Lp/2 -((sqrt(3))/4)*Lp 0];
C2=[Lp/2 -((sqrt(3))/4)*Lp 0];
C3=[0 ((sqrt(3))/4)*Lp 0];
%% Variables

%Leg length
L1=[80 100 44.45];%44.45; %crank
L2=[88.90 88.90 80]; %rod

% platform position limits
OP=[Px Py 0];

%% Platform orientation
R=rotz(tz); % matriz de rotacion
A=[A1' A2' A3'];
C=[C1' C2' C3'];
G=zeros(3,3);
for i=1:3
    AC= OP' + R*C(:,i) - A(:,i);
    ACX=AC(1);
    ACY=AC(2);
    % analisis vectorial
    a=ACX^2 +ACY^2-L1(i)^2-L2(i)^2;
    b=2*L1(i)*L2(i);
    t2(i)=real(acos(a/b))+((1+(-1)^(1+i))*pi/2);
    if ( t2(i) == 2*pi) || (t2(i) == pi)
        er=1;
        break
    else
      er=0;
      d=L2(i)*sin(t2(i))/(sqrt(ACX^2 +ACY^2));
      t1(i)=atan2(ACY,ACX)- asin(d);
      AB(:,i)=[L1(i)*cos(t1(i)); L1(i)*sin(t1(i)); 0];
        % screw
        BC(:,i)= AC-AB(:,i);
        sr1(:,i)=BC(:,i)/norm(BC(:,i));
        sro(:,i)=cross(AB(:,i),sr1(:,i));
        s0=cross(A(:,i),s);
        Jq(i,i)=[sro(3,i)' sr1(1:2,i)']*[1 s0(1:2)]';
```

```
            gra=[sro(3,i)' sr1(1:2,i)']'*[sro(3,i)' sr1(1:2,i)'];
            G=G+gra;
      end
end
%% Constraints

if er==0
      %% Direct kinematic
      D=zeros(3,3);
      D(3,3)=1;
      Gm=G\D;
      eval=eig(Gm);
      max_auto= max(max(real(eval)));
      lam=1/max_auto;
      Di=sqrt(lam);
    %% Inverse kinematic
      I=abs(det(real(Jq)));
else
        Di=0;
        I=0;
end
end
```

Appendix B

Workspace Analysis and Optimization Algorithms in Matlab

B.1 Genetic Algorithm

```
%% Geneticos
clear
clc
tic
xub=[pi pi pi]; %upper limits of the variables "l"
xlb=[-pi -pi -pi]; %lower limits of the variables "l"
iter=100; %number of iterations
Fb=zeros(1,iter);
Fig=zeros(1,iter);
dx=0.001; %precision
n=80; %number of particles
l=3; %number of variables
pc=30; %crossover percentagen
q=q1(xub,xlb,dx,l); %number of bits
mil=l*q; %population size
pm=10; %mutacion percentagen
G=rand(n,mil); %firts ramdomica population
G=cod(G,n,mil);
for ii=1:iter
    x=deco(G,n,q,l,xub,xlb);
%% reproduction
    prom=pro(x,n);
    [a,b]=max(prom);
    Fb(ii)=fit(x(b,1),x(b,2),x(b,3));
    el=G(b,:);
    G=rep(prom,G,n);
%% Crossover
    G=cro(pc,n,G,mil);
%% Mutation
    G=muta(pm,G,n,mil);
    G(1,:)=el;
    x2=deco(G,n,q,l,xub,xlb);
%% Plot
   disp(ii);
end
```

```
toc
% Play the movie ten times
%movie(Fig,10)
plot(1:iter,Fb);
```

B.1.1 q1(xub,xlb,dx,l)

```
function t=q1(xub,xlb,dx,l)
        q2=zeros(l,1);
        for i=1:l
            q2(i,1)=log(((xub(i)−xlb(i))/dx)+1)/log(2);
        end
        t=ceil(max(q2));
end
```

B.1.2 cod(G,n,mil)

```
function M=cod(G,n,mil)
        for j=1:n
            if j<n
                for i=1:mil
                    if G(j,i)>G(n,i)
                        G(j,i)=0;
                    else
                        G(j,i)=1;
                    end
                end
            end
            if j==n
                for i=1:mil
                    if G(j,i)>G(n−1,i)
                        G(j,i)=0;
                    else
                        G(j,i)=1;
                    end
                end
            end
        end
        M=G;
end
```

B.1.3 deco(G,n,q,l,xub,xlb)

```
function v=deco(G,n,q,l,xub,xlb)
        x=zeros(n,l);
        for j=1:n
            qi=1;
            for k=1:l
                su=0;
                ex=0;
                for i=qi:(q∗k)
```

```
                        su=su+(2^(ex)*G(j,i));
                        qi=qi+1;
                        ex=ex+1;
                    end
                    x(j,k)=xlb(k)+((xub(k)-xlb(k))/((2^q)-1))*su;%(xlb(k)+(xub(k)-xlb(k))/((2^q)-1))*
                end
            end
        v=x;
end
```

B.1.4 pro(x,n)

```
%% Funcion que calcula el porcentage de ser seleccionado
function p=pro(x,n)
        F=zeros(n,1);
        pru=zeros(n,1);
        for i=1:n
            F(i,1)=fit(x(i,1),x(i,2),x(i,3));
        end
        sim=sum(F);
        for i=1:n
            pru(i)=F(i,1)/sim;
        end
        p=pru;
end
```

B.1.5 rep(prom,G,n)

```
%% Reproduction
function re=rep(prom,G,n)
        pr=zeros(n+1,1);
        B=zeros(n,1);
        Gm=G;
        for j=2:n+1
            pr(j)=pr(j-1)+prom(j-1);
        end
        R(1:n,1)=pr(1:n);
        R(1:n,2)=pr(2:n+1);
        for i=1:n
            a=rand(1,1);
            for k=1:n
                if R(k,1)<= a
                    if a<=R(k,2)
                        B(i)=k;
                        break
                    end
                end
            end
        end
 %% Reproduction of the Better genes
        for l=1:n
            Gm(l,:)=G(B(l),:);
        end
        re=Gm;
```

```
end
```

B.1.6 cro(pc,n,G,mil)

```
%% Crossover
function c=cro(pc,n,G,mil)
        co=round((n*pc)/100); % Percentage of the population to crossover
        for j=1:co
            ta=floor(rand(1,1)*(n-1))+1;
            ma=floor(rand(1,1)*mil)+1;
            jo=ta+1;
            ju=floor(rand(1,1)*10)+1;
            if ju < ma
                ko=G(ta,ju:ma);
                G(jo,ju:ma)=ko;
            else
                ko=G(ta,ma:ju);
                G(ta,ma:ju)=G(jo,ma:ju);
                G(jo,ma:ju)=ko;
            end
        end
        c=G;
end
```

B.1.7 muta(pm,G,n,mil)

```
%% Mutacion
function m=muta(pm,G,n,mil)
        mu=round((n*mil*pm)/100);
        for no=1:mu
            b=floor(rand(1,1)*n)+1;
            c=floor(rand(1,1)*mil)+1;
            if G(b,c)==1
                G(b,c)=0;
            else
                 G(b,c)=1;
            end
        end
        m=G;
end
```

B.2 S-G Worspace

B.2.1 Worspace 2D

```
clear
clc
ns=0.01; % Step size
z=0; % Initial orientation
[L1, L2] = meshgrid(−pi:ns:pi);
    j=0;
    for y=−pi:ns:pi;
        j=j+1;
       k=0;
        for x=−pi:ns:pi
            k=k+1;
            L3(j,k)=fv22(z,y,x); %Detec the feasibles points
        end
    end
     surf(L1,L2,L3) % Plot workspace section

ylabel('\vartheta (rad)') % y−axis label
xlabel('\psi(rad)') % x−axis label
zlabel('Index value')
```

B.2.2 Worspace 3D

```
clc
clear
ns=0.1;
lo=length(−pi:ns:pi); %% Platform orientation limits
v=zeros(lo,lo,lo);
i=0;
for z=−pi:ns:pi;
    i=i+1;
    j=0;
    for y=−pi:ns:pi;
       j=j+1;
       k=0;
        for x=−pi:ns:pi
            k=k+1;
            v(i,j,k)=fv22(z,y,x); % Detec the feasibles points
        end
    end
end
%% Isosuperficie
[X Y Z] = meshgrid(−pi:ns:pi);
isosurface(X,Y,Z,v,0);
%% Figure appearance
view(3); axis tight
camlight
alpha(.4)
%% Labels
ylabel('\vartheta(rad)') % y−axis label
xlabel('\psi(rad)') % x−axis label
zlabel('$\varphi(rad)$', 'interpreter', 'latex')
```

B.3 3-RRR Worspace

B.3.1 Worspace 2D

```
%% workspace 2D
clear
clc
ns=1;
z=0; % Initial orientation
lin=-60; % Minimal displacement
lup=250; % Maximum displacement
[L1, L2] = meshgrid(lin:ns:lup);
    j=0;
    for y=lin:ns:lup;
        j=j+1;
       k=0;
        for x=lin:ns:lup
            k=k+1;
            L3(j,k)=fv22(x,y,z); %Detec the feasibles points
        end
    end
     surf(L1,L2,L3) % Plot workspace section
ylabel('py (mm)') % y-axis label
xlabel('px (mm)') % x-axis label
```

B.3.2 Worspace 3D

```
clc
clear
% isosuperficie
vt=0;
ns=1; %step size
nsp=1; % Imagen quality
lm01=[-180 180]; % orientation limits
lm02=[-100 300]; % displacement limits
lo1=length(lm01(1):nsp:lm01(2));
lo2=length(lm02(1):ns:lm02(2));
v=zeros(lo2,lo2,lo1);
i=0;
for y=lm02(1):ns:lm02(2)
    i=i+1;
    j=0;
    for x=lm02(1):ns:lm02(2);
       j=j+1;
       k=0;
        for z=lm01(1):nsp:lm01(2)
            k=k+1;
            v(i,j,k)=fv22(x,y,z); %detec the feasibles points;
        end
    end
    disp(y)
end
[X Y Z] = meshgrid(lm02(1):ns:lm02(2),lm02(1):ns:lm02(2),lm01(1):nsp:lm01(2));
isosurface(X,Y,Z,v,0); %isosuperficie
view(3); axis tight; Box on;
```

```
camlight; lighting gouraud; alpha(0.5); %Figure appearance
ylabel('py (mm)') % y−axis label
xlabel('px (mm)') % x−axis label
zlabel('\psi (deg)') % x−axis label
```