Optimization of a Reconfigurable Manipulator with Lockable Cylindrical Joints

Gabriel Zeinoun

A Thesis

in

The Department

of

Mechanical and Industrial Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master Applied Science (Mechanical Engineering) at

Concordia University

Montreal, Quebec, Canada

August 2013

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: Gabriel Zeinoun

Entitled: Optimization of a Reconfigurable Manipulator with Lockable Cylindrical Joints

and submitted in partial fulfillment of the requirements for the degree of

M.A.Sc Mechanical Engineering

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Dr. Muthukumaran Packirisamy Chair

Dr. Youmin Zhang Examiner

Dr. Amir Aghdam Examiner

Dr. Ramin Sedaghati Supervisor

Dr. Farhad Aghili Supervisor

Approved by Dr. Sivakumar Narayanswamy, MASc Program Director
Department of Mechanical and Industrial Engineering

Dr. Christopher Trueman, Interim Dean
Faculty of Engineering & Computer Science

Date 20 September 2013

ABSTRACT

Optimization of a Reconfigurable Manipulator with Lockable Cylindrical Joints

Gabriel Zeinoun

This thesis presents a global optimization methodology to find the optimal Denavit-Hartenbeg parameters of a serial reconfigurable robotic manipulator maximizing a cost function over a pre-specified workspace volume and given lower and upper bounds on the design parameters. Several cost functions are investigated such as the manipulability measure, maximum force/torque capability of the manipulator at its end-effector, and maximum velocity capability of the manipulator, therefore improving the general kinetostatic performance of the manipulator. A modified global and posture-independent parameter of singularity (MPIPS) is presented, and a generic global optimization approach is proposed, using combined genetic algorithm (GA) and sequential quadratic programming (SQP). Different case studies are provided for a 3-DOF and a 6-DOF reconfigurable manipulator. Finally, a weighted objective function that balances between the opposing actions of the end effector velocity and force is proposed. The results are illustrated to demonstrate the performance of the generated manipulators, and are validated. Post-optimality analysis has also been conducted to investigate the sensitivity of the index to the variation in optimal parameters.

Keywords: Global optimization, manipulability, isotropy, kinetostatic performance, parameter of singularity.

ACKNOWLEDGMENTS

This thesis is made possible only through the help and strong support of my teachers, family, and friends. I would like to sincerely express my gratitude towards the following advisors and contributors:

Professor Ramin Sedaghati, for his persistent help, invaluable information and critique on optimization, and thorough feedback on the logic and organization of the paper.

Professor Farhad Aghili for his accurate guidance and suggestions throughout the research, and novel ideas in robotics and formulation of the optimization strategy.

My parents, who never stopped supporting me morally, emotionally, and financially during the course of my study.

Last but not least, I would like to thank my family and friends, who always provided me with encouragement and comfort throughout my journey.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

a	Link length
α	Link twist
d	Joint offset
θ	Joint angle
θ_{max}	Maximum joint angle
θ_{min}	Minimum joint angle
θ_{IK}	Inverse kinematic solution
$\dot{\theta}$	Joint rate
$\dot{\theta}_{max}$	Maximum joint rate
$\dot{\theta}_{min}$	Minimum joint rate
x	Design variable vector
R	Rotation matrix
P	Position of frame
T	Transformation matrix
v	Linear velocity of the end-effector
ω	Angular velocity of the end-effector
v	Cartesian velocity of the end-effector
J	Jacobian matrix
J_T	Translational Jacobian sub-matrix
J_R	Rotational Jacobian sub-matrix

$J_{T,A}$	Translational Jacobian sub-matrix of the manipulator's arm
$J_{T,W}$	Translational Jacobian sub-matrix of the manipulator's wrist
$J_{R,A}$	Rotational Jacobian sub-matrix of the manipulator's arm
$J_{R,W}$	Rotational Jacobian sub-matrix of the manipulator's wrist
τ	Joint torque
τ_{max}	Maximum joint torque
τ_{min}	Minimum joint torque
F	Force at the end-effector
N	Moment at the end-effector
$\mathcal{F}$	General force at the end-effector
W	Manipulability index
W_{max}	Maximum manipulability in the workspace
σ	Singular value of the Jacobian matrix
σ_{max}	Maximum singular value of the Jacobian matrix
σ_{min}	Minimum singular value of the Jacobian matrix
Σ	Singular value matrix
U, V	Orthogonal matrices
I	Identity matrix
κ	Condition number
κ_{min}	Minimum condition number in the workspace
η	Global Conditioning Index
W_{PS}	Modified parameter of singularity
$\mathcal{W}_{PS}$	Modified posture-independent parameter of singularity

$\mathrm{W}_{PS-\mathcal{F}}$	Modified parameter of singularity for force maximization
$\mathcal{W}_{PS-\mathcal{F}}$	Modified posture-independent parameter of singularity for force maximization
$f(\mathrm{x})$	Cost function or objective function
$c_j(\mathrm{x})$	Inequality constraints
$d_k(\mathrm{x})$	Equality constraints

CHAPTER 1

INTRODUCTION

1.1 Motive and Problem Statement

Robotic manipulators are generally required to be versatile when performing tasks in extreme environments, and they are often desired to reconfigure themselves in order to satisfy the constraints of different environments (Castano, Behar, & Will, 2002), (Farritor, Dubowsky, Rutman, & Cole, 1996), (Aghili & Parsa, 2009). Specifically in space applications, it is efficient to employ a single robot that can perform different tasks such as assembly, inspection, and pick and place operations (Farritor, Dubowsky, Rutman, & Cole, 1996). In general, the optimal operation of each of these tasks demands a specific manipulator design. For instance, robots with a low condition number are suitable for tasks requiring high accuracy (Merlet, 2006a), and robots with maximum manipulability measure are appropriate for dexterous tasks (Merlet, 2006b). Also, some tasks demand large forces at the end effector, while others require high velocity.

The concept of reconfigurable robots to add versatility to the robot was presented in (Paredis & Khosla, 1993), reconfigurable manipulators for space applications in (Farritor, Dubowsky, Rutman, & Cole, 1996) and (Shibata & Ohkami, 2002), and reconfigurable manipulators that can take different shapes in (Castano, Behar, & Will, 2002). A reconfigurable manipulator with lockable passive cylindrical joints was also introduced in (Aghili & Parsa, 2009). The latter, which is our manipulator type of interest, is illustrated in Figure 1.1. This kind of robotic arm is able to reconfigure its kinematic

design parameters, i.e., the Denavit-Hartenberg (DH) parameters, by forming a closed kinematic chain with a fixed point, unlocking the required passive joints, and achieving the desired parameters. Upon converging to these values, the brakes are locked again, and a new configuration is born. A detailed study on the mechanism, singularity analysis, and kinematics and control during reconfiguration, can be found in (Aghili & Parsa, 2009).

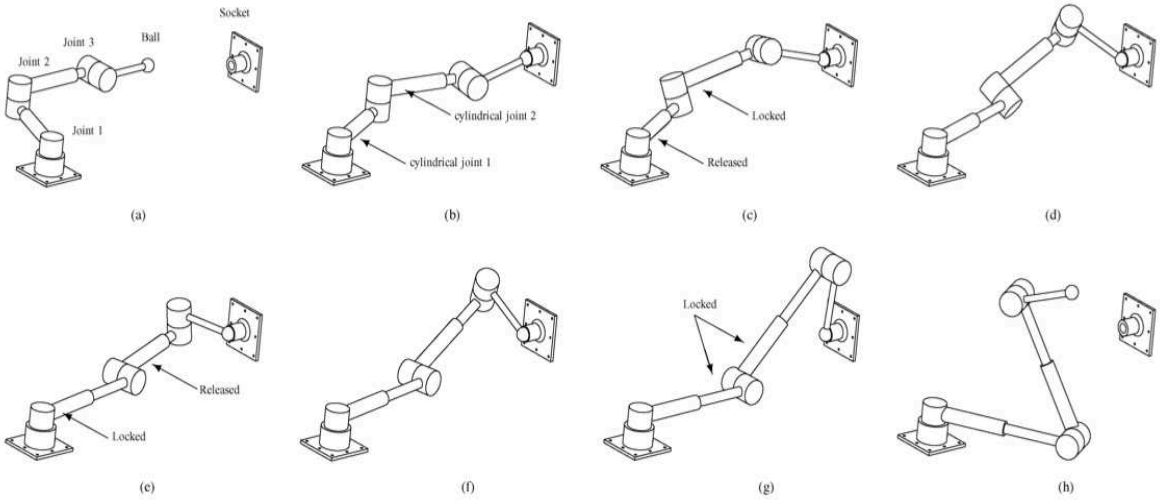


Figure 1.1 Simplified model of a robotic arm during reconfiguration. (a) Initial configuration. (b) Grasping fixed point to form a closed kinematic chain and constrain x-y-z motion. (c)-(e) Releasing brake, reconfiguring, and then relocking first passive cylindrical joint. (f)-(g) Releasing brake, reconfiguring, and then relocking second passive joint. (h) Final configuration is attained. (Aghili & Parsa, 2009)

Clearly, prior to performing each type of task, it is essential to determine the ideal kinematic design parameters that satisfy the performance criteria of the robot. Finding these values, which can be achieved by an optimization methodology, is the main motive of this study, where this methodology will generate the ideal robot configuration that maximizes its performance criteria in the desired workspace. Evidently, the robot then reconfigures to the new configuration and operates as required. The problem can therefore be stated as follows:

Find the optimal Denavit-Hartenberg parameters of a serial robotic manipulator subject to lower and upper bounds, to maximize the (1) manipulability index within a pre-specified workspace volume, (2) Euclidean norm of the end-effector (EE) force and torque given the constraint of actuator torque in the manipulator joints, while maintaining a uniform joint torque distribution within a pre-specified workspace volume, and (3) Euclidean norm of the EE linear and angular velocity given the constraint of actuator rate in the manipulator joints, while maintaining a uniform joint rate distribution within a pre-specified workspace volume.

1.2 Literature Review

Here a pertinent literature review concerned with optimization of serial reconfigurable robotic manipulator has been presented.

The optimization of a robot workspace over its link lengths, as the design parameters, is reported in (Kumar & Waldron, 1981) and (Gupta & Roth, 1982), while optimization of kinematic parameters and criteria for fault tolerance are discussed in (Jing & Yi, 2007) and (Abdi & Nahavandi, 2012). Design of serial manipulators that satisfied certain kinematic specifications with obstacle avoidance was presented in (Paredis & Khosla, 1993), while Snyman and Van Tonder discussed link length design to minimize average torque requirement (Snyman & Van Tonder, 1999). Serial manipulator design for optimal dynamic performance was shown in (Shiller & Sundar, 1991), whereas (Zhang, Liu, & Ding, 2012) and (Krefft & Hesselbach, 2006) proposed a dynamic optimization method for parallel manipulators. Multi-objective optimization for different criteria was discussed in (Carbone, Ottaviano, & Ceccarelli, 2008) and (Barissi & Taghirad, 2008). For instance, the former examined a brief study that comprised of position and orientation

workspace optimization, singularity avoidance, optimal path planning and travelling time, stiffness criterion, lightweight design, and power consumption. Optimal design of medical robots for minimally invasive surgery with multiple objectives was also discussed in (Konietschke, Ortmaier, Weiss, Engelke, & Hirzinger, 2003) and (Du, Zhang, & Zou, 2007).

Huo and Baron (Huo & Baron, 2008) used joint limits and their proposed parameter of singularity (which will be later discussed in detail) to optimize the joint space trajectory of a six-rotation-axis (6-DOF) industrial robot to avoid singularity. In their optimization methodology, they decomposed the desired instantaneous end effector twist (electrode) into two orthogonal components, the relevant task subspace and the redundant task subspace. The projection matrices are then task-dependent and vary with the instantaneous geometry of the task at hand (Huo & Baron, 2008) (Zargarbashi, Khan, & Angeles, 2012).

Genetic algorithms were employed for synthesizing and globally optimizing robot configurations in (Kim & Khosla, 1993), (Leger, 1999), and (Khatami & Sassani, 2002). The latter considered the measure of isotropy as a performance index to design the manipulator. On the other hand, Sobh and Toundykov (Sobh & Toundykov, 2004), and Bagchi (Bagchi, 2007), utilized local methods by optimizing manipulator configurations that maximized the measure of manipulability using the steepest descent method and sequential quadratic programming, respectively. Sobh and Toundykov's approach was task-based, where the robot was optimized for a pre-specified task (trajectory). They solved for the inverse kinematics by including the joint angles in the objective function, and performed an optimization that minimized the distance between the EE and the

desired points in the trajectory, as well as generated the optimal design parameters. Bagchi followed a similar method for determining the inverse kinematic solution.

A design strategy with constraint satisfaction, using the interval-based algorithm was used in (Oetomo, Daney, & Merlet, 2009), and a task-based design of serial robot manipulators with trajectory planning was employed using direct search non-gradient optimization in (Al-Dois, Jha, & Mishra, 2013).

1.3 Problem Approach

The optimization techniques mentioned above mainly rely on a single or limited optimality criteria, and also in many cases the employed numerical methods are only locally convergent. Therefore there is a need to develop a global optimization strategy that incorporates multiple optimality criteria, allowing the designer to optimize the robot such that it performs best in different situations.

Considering this, this thesis presents a formal optimization methodology to find the ideal design parameters that satisfy the multiple performance criteria as follows:

- Maximize the general force or velocity ellipsoid of the end effector.
- Make the EE's general force and velocity as uniform as possible.
- Maximize the manipulator's distance from singularity.
- Increase positioning accuracy.

The key concept to this methodology is optimizing the manipulator's worst-case performance in the desired workspace, thereby meeting the optimality criteria. These performance criteria are correlated in a way or another, and all depend on some major performance indices applied in robotics. Some of the most popular indices are based on

the *manipulability measure* (Yoshikawa, 1985a), *condition number* (Salisbury & Craig, 1982), and *minimum singular value* (Klein & Blaho, 1987). Other less popular indices include the *service angle* (Vinogradov, Kobrinski, & Stepanenko, 1971) (Angeles & Lopez-Cajun, 1992), and the *joint rate availability* (Liegeois, 1977) (Klein & Blaho, 1987).

The numerical optimization methods that are used to meet our required criteria are based on combined Genetic Algorithm (GA) and Sequential Quadratic Programming (SQP), guaranteeing a precise global solution to the problem as will be justified later.

1.4 Thesis Outline

Chapter 2 provides a brief introduction to the kinematics of robotic manipulators, as well as a definition of the reconfigurable manipulator to be optimized. A review of manipulator categories, joint types, DH parameters, manipulator kinematics extended to reconfigurable manipulator kinematics, Jacobian matrix, and EE force and velocity is discussed.

In Chapter 3, we investigate various performance indices such as the parameter of singularity (W_{PS}), the manipulability measure, and condition number of the Jacobian matrix. The modified posture-independent parameter of singularity (MPIPS) is introduced, rendering it to a global index in the robot's desired workspace.

In Chapter 4, a novel and precise optimization methodology is proposed, based on optimizing the robot's worst-case performance inside its desired workspace. The problem formulation and constraints are described based on some assumptions on the manipulator links, joints, and workspace. A brief overview of stochastic based optimization algorithm

using GA and gradient-based nonlinear optimization algorithm based on SQP, is also presented, providing a better understanding as to why combined GA and SQP were chosen to perform the optimization.

Chapter 5 illustrates various case studies for maximizing the performance of 3-DOF and 6-DOF reconfigurable manipulators. The necessity of GA is demonstrated, and the optimization is validated by its comparison with another method. Different case studies are also provided where the EE force or velocity is maximized, given that the maximum joint torque and joint rate are constrained.

CHAPTER 2

KINEMATICS OF RECONFIGURABLE MANIPULATORS

2.1 Introduction

A robotic manipulator is defined by a set of rigid links that are connected together by joints (Craig, 2005). In this chapter, first basics of robotic manipulators are briefly reviewed and then extended to provide the general description of *n*-DOF reconfigurable manipulator to be optimized.

2.2 Serial and Parallel Manipulators

Manipulators can be classified in two main categories: (1) *serial manipulators*, whose links are connected consecutively. Among the most utilized serial manipulators in the industry, one can mention the SCARA-type, Unimation PUMA 560, and Yasukawa Motoman L-3 robots; (2) *Parallel manipulators*, which on the contrary utilize several serial chains directly connected to a single EE or platform. Of the most popular parallel manipulators is the Stewart Platform hexapod, which is extensively used in flight simulators (Angeles, 2007) (Stewart, 1965) (Krefft & Hesselbach, 2006).

Throughout this thesis, the manipulator of interest is a serial and reconfigurable manipulator similar to the one proposed by Aghili and Parsa in (Aghili & Parsa, 2009), Figure 1.1.

2.3 Types of Joints

Manipulator joints are generally composed of prismatic or revolute joints. The former are defined by linear motion along a single axis, while the latter are defined by relative motion about an axis (rotation). Each of these classes has only a single DOF, and

the manipulator as a whole can be considered to have as much DOFs as the number of joints (Craig, 2005). Joints can sometimes be a combination of both prismatic and revolute joints, making them cylindrical. Other hybrid types are planar, screw, and spherical joints.

It is noteworthy to indicate that the *n*-DOF reconfigurable manipulator that will be studied in this thesis includes revolute joints, cylindrical joints, and a spherical joint at the wrist. The revolute joints are treated as active joints, and are equipped with actuators, while the cylindrical joints are passive, and have neither actuators nor sensors (Aghili & Parsa, 2009). The latter are normally locked when the robot operates in free-space, and only unlocked when the robot makes a closed kinematic chain and needs to reconfigure. Figure 2.1 shows a reconfigurable manipulator as described by (Hebert, Tatossian, Cairns, Aghili, & Parsa, 2007). It is composed of: 3 active revolute joints each with an angle θ, 2 passive cylindrical joints each with a twist α and a linear relative displacement l, and a passive spherical wrist.

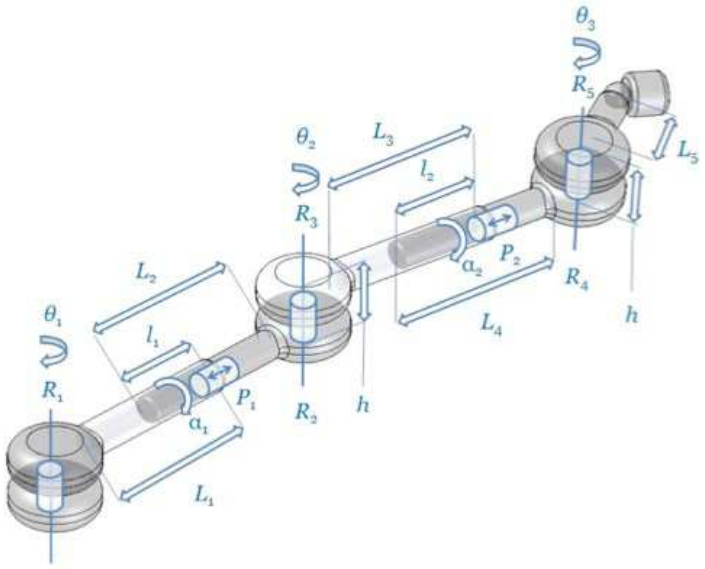


Figure 2.1 Reconfigurable manipulator. (Hebert, Tatossian, Cairns, Aghili, & Parsa, 2007)

2.4 The Denavit-Hartenberg Parameters

Each robotic manipulator is described by its link lengths, twist angles, link offsets, and joint angles, known as the Denavit–Hartenberg parameters (DH parameters). Introduced by Jacques Denavit and Richard Hartenberg, they became associated with a convention used in attaching frames of references to the links of a manipulator (Hartenberg & Denavit, 1964).

The first step in defining a robot is the link-frame attachment procedure, which should be done in a proper way. For the purpose of our study, we will consider a manipulator with *n* links connected serially by *n*-1 joints. The links are numbered from 1 to *n*, starting from the immobile base or the reference frame, which might be numbered as 0. Links 1 to *n* are all mobile, actuated by the joints. A useful summary for assigning the frames is provided by Craig (Craig, 2005) as indicated below, and Figure 2.2 can be used for graphical reference:

- Identify all the joint axes as shown in Figure 2.2, and for the next steps consider two adjacent joint axes *i* and *i*+1.
- Identify the common perpendicular between the adjacent joint axes, and assign a link frame origin at the point of intersection with the *i*th axis.
- Assign the $\mathbf{Z}_i$ axis along the *i*th joint axis.
- Assign the $\mathbf{X}_i$ axis along the common perpendicular. In the case that joint axes intersect, $\mathbf{X}_i$ would be normal to the plane formed by these two axes.
- The $\mathbf{Y}_i$ axis completes the right-hand rule for $\mathbf{X}_i$ and $\mathbf{Z}_i$.

Frame {0} is assigned such that it overlaps with {1} when the first joint variable (d for prismatic joint, or θ for revolute joint) is zero. As for {n}, it is assigned in such a way that sets the most number of DH parameters to zero.

The DH parameters are then defined as follows:

a_i = Link length: distance from original frame Z_i to Z_{i+1} along X_i

α_i = Link twist: angle measured from Z_i to Z_{i+1} along X_i

d_i = Joint offset: distance from X_{i-1} to X_i along Z_i

θ_i = Joint angle: angle from X_{i-1} to X_i along Z_i

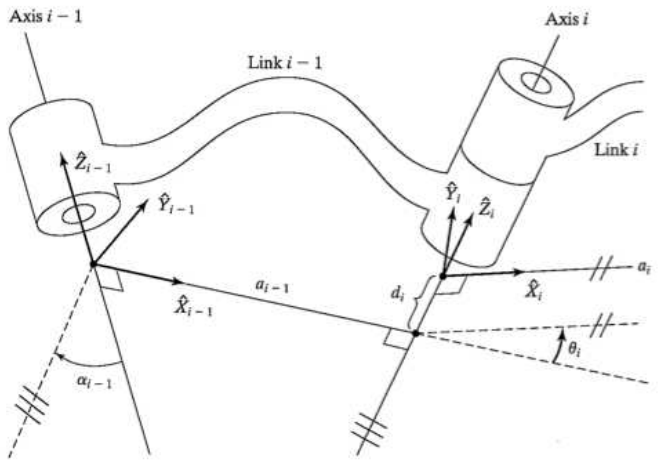


Figure 2.2 DH parameters and frame attachment for a general serial manipulator section. (Craig, 2005)

The DH parameters of a 6-DOF reconfigurable manipulator are represented in Table 2.1, with some assumed *constant* values for joint offsets and the fixed base's link length. The joint-angle column will be represented by stars "*" since these values vary while maneuvering the robot arm. Also, due to design limitations, the joint angles will be bounded by assumed θ_{min} and θ_{max} values. Finally, since the robot has passive cylindrical joints on the links as the ones shown in Figure 1.1 and Figure 2.1, these will be represented by the design variables (α_1, α_2, l_1, l_2) in the DH table since they can vary,

adopting different values that satisfy different goal criteria. Note that the link offsets are not considered as design variables. As a summary, the general topology of the 6-DOF reconfigurable manipulator can be illustrated in Figure 2.3.

Table 2.1 DH parameters of a 6-DOF reconfigurable manipulator.

i	***α***	***a (m)***	***d (m)***	***θ***	***Joint***	θ_{min}	θ_{max}
1	90°	0.2	0	*	R	-180°	180°
2	α_1	l_1	0.2	*	R	-108°	108°
3	α_2	l_2	0.2	*	R	-144°	144°
4	-90°	0	0	*	R	-180°	180°
5	90°	0	0	*	R	-180°	180°
6	-90°	0	0	*	R	-180°	180°

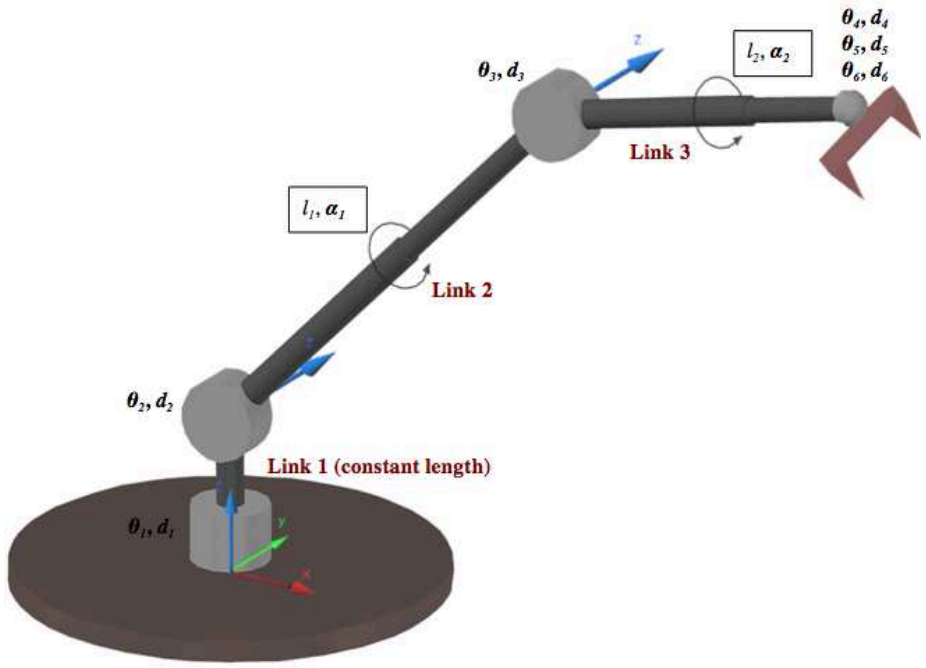


Figure 2.3 Reconfigurable robot to be optimized with DH parameters, drawn on AutoCAD.

2.5 Transformation Matrices and Forward Kinematics

Each frame of reference is described by a 4 x 4 transformation matrix. The matrix is clearly a function of the DH parameters, and is denoted as follows:

$$ {}^{i-1}_{i}T = \begin{pmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & \alpha_{i-1} \\ \sin(\theta_i)\cos(\alpha_{i-1}) & \cos(\theta_i)\cos(\alpha_{i-1}) & -\sin(\alpha_{i-1}) & -\sin(\alpha_{i-1})d_i \\ \sin(\theta_i)\sin(\alpha_{i-1}) & \cos(\theta_i)\sin(\alpha_{i-1}) & \cos(\alpha_{i-1}) & \cos(\alpha_{i-1})d_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.1) $$

where ${}^{i-1}_{i}T$ provides information about the position and orientation of frame $\{i\}$ with respect to frame $\{i\text{-}1\}$. The first 3 rows of the first 3 columns describe the orientation sub-matrix of $\{i\}$ with respect to $\{i\text{-}1\}$, and the first 3 rows of the last column define the x-y-z translation coordinates of $\{i\}$ with respect to $\{i\text{-}1\}$. The last row should have "[0 0 0 1]" elements so that the transformation matrix is orthonormal.

The next step is to describe the position and orientation of the EE with respect to the base frame. This is done through the multiplication of individual transformation matrices from *i*=1 to *n* and developing what is known as the *kinematic equations*. Thus,

$$ {}^{0}_{n}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T \ldots {}^{n-1}_{n}T \quad (2.2) $$

represents the 4 x 4 transformation matrix that relates the frame $\{n\}$ to the frame $\{0\}$. The elements of this matrix are known as the kinematics of the robot at hand (Craig, 2005).

By referring to Table 2.1, the simplified matrices for a 6-DOF reconfigurable manipulator are shown below:

$$ {}^{0}_{1}T = \begin{pmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & 0.2 \\ 0 & 0 & -1 & 0 \\ \sin(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.3) $$

$$ {}^{1}_{2}T = \begin{pmatrix} \cos(\theta_2) & -\sin(\theta_2) & 0 & l_1 \\ \sin(\theta_2)\cos(\alpha_1) & \cos(\theta_2)\cos(\alpha_1) & -\sin(\alpha_1) & -0.2\sin(\alpha_1) \\ \sin(\theta_2)\sin(\alpha_1) & \cos(\theta_2)\sin(\alpha_1) & \cos(\alpha_1) & 0.2\cos(\alpha_1) \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.4) $$

$$ {}^{2}_{3}T = \begin{pmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & l_2 \\ \sin(\theta_3)\cos(\alpha_2) & \cos(\theta_3)\cos(\alpha_2) & -\sin(\alpha_2) & -0.2\sin(\alpha_2) \\ \sin(\theta_3)\sin(\alpha_2) & \cos(\theta_3)\sin(\alpha_2) & \cos(\alpha_2) & 0.2\cos(\alpha_2) \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.5) $$

$$ {}^{3}_{4}T = \begin{pmatrix} \cos(\theta_4) & -\sin(\theta_4) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sin(\theta_4) & -\cos(\theta_4) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.6) $$

$$ {}^{4}_{5}T = \begin{pmatrix} \cos(\theta_5) & -\sin(\theta_5) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \sin(\theta_5) & \cos(\theta_5) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.7) $$

$$ {}^{5}_{6}T = \begin{pmatrix} \cos(\theta_6) & -\sin(\theta_6) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sin(\theta_6) & -\cos(\theta_6) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.8) $$

Remark that the design parameters are kept as variables (α_1, α_2, l_1, l_2), and are only found among the first 3 transformation matrices. Also notice that the last 3 transformation matrices do not have any translational part in the fourth column since they only represent the spherical wrist of the robot arm.

The kinematic equations that determine the position and the orientation of the EE with respect to the base frame are then obtained by the product of all link transformations in the following order:

$$ {}^{0}_{6}\mathrm{T} = {}^{0}_{1}\mathrm{T}\ {}^{1}_{2}\mathrm{T}\ {}^{2}_{3}\mathrm{T}\ {}^{3}_{4}\mathrm{T}\ {}^{4}_{5}\mathrm{T}\ {}^{5}_{6}\mathrm{T}. \tag{2.9} $$

These are a function of the joint angle vector $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6]$, and the elements of the design parameter vector $\mathbf{x} = [\alpha_1, \alpha_2, l_1, l_2]$, such that:

$$ {}^{0}_{n}\mathrm{T} = f(\boldsymbol{\theta}, \mathbf{x}). \tag{2.10} $$

2.6 Inverse Kinematics

The problem of inverse kinematics (IK) can be stated as follows: for a given *n*-DOF manipulator, if a certain position and orientation of an EE is desired, and the base frame is specified, what are the values of the *n* joint angles that can generate such an EE pose?

In order to answer that question, the transformation matrix ${}^{0}_{N}T$ in Eq. (2.10) can be a starting point. Consider a 6-DOF reconfigurable robot whose passive cylindrical joints are now fixed. ${}^{0}_{6}T$ is a function of the unknown joint variables, and has 16 kinematic equations, 4 of which are trivial. Therefore, 12 equations remain to solve for the 6 joint variables. However, of the rotational part of ${}^{0}_{6}T$, only 3 are linearly independent, added to the other 3 which define the position of the EE (Craig, 2005). The total number of equations is now equal to the number of unknowns, and can be used to solve for θ_1 to θ_6. It is important to mention that the 6 equations are very long and complex, and in general require the implementation of special substitutions for solving transcendental equations, *if* a solution exists (Craig, 2005). The following equations to be

retained for later use, will be used to convert the transcendental equations into polynomial functions in *u*:

$$u = \tan(\frac{\theta}{2})$$
$$\cos(\theta) = \frac{1-u^2}{1+u^2} \tag{2.11}$$
$$\sin(\theta) = \frac{2u}{1+u^2}$$

Different methods can be used to solve for the inverse kinematics and will be indicated thereafter.

2.6.1 Methods of Solution

Since solving for the joint angles can be a rigorous process, several different methods of solution are available in solving for joint angles. Algebraic, numerical, as well as geometric methods exist, and usually the choice of method depends on the complexity of the manipulator model, and the necessity of an efficient solution. The literature addressing different inverse kinematic problems and general solutions is very rich, and comprises of (Pieper, 1968), (Wang & Chen, 1991), (Raghavan & Roth, 1993), (Mavroidis, Ouezdou, & Bidaud, 1994), (Manocha, & Canny, 1994), (Chapelle & Bidaud, 2004), (Buss, 2004), (Wenz & Worn, 2007), and (Vasilyev & Lyashin, 2010). The IK solution can also be obtained by solving an optimization problem. Since the desired EE positions are known (the points are already defined by the user within the workspace), and the $\boldsymbol{\theta}$ joint values are unknown, we can determine the optimum values of $\boldsymbol{\theta}$ by calculating the error distance between the desired and the present location of the end effector and find a $\boldsymbol{\theta}$ that minimizes the distance between the two. Optimization was used to obtain IK solutions in (Sobh & Toundykov, 2004) (Bagchi, 2007) and (Xu, Wang, & Sun, 2010).

In our case, we will make use of Pieper's method that is very efficient for manipulators with three intersecting axes (EE with a spherical wrist), and a closed-form IK solution will be finally developed as a function of **x**.

Pieper's Method for 3 Intersecting Axes:

By writing the forward kinematics relating the wrist's base (the point where the 3 axes intersect) to the robot's base frame as a function of the design parameters and the first 3 joints (θ_1, θ_2, and θ_3), we determine the position of the EE:

$$ {}^0P_{4ORG} = {}^0_1T\,{}^1_2T\,{}^2_3T\,{}^3P_{4ORG} = \begin{bmatrix} x_d \\ y_d \\ z_d \\ 1 \end{bmatrix}, \tag{2.12} $$

where the subscript "*d*" represents the desired position, pre-specified by the designer. It can also be written as

$$ {}^0P_{4ORG} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ 1 \end{bmatrix} = {}^0_1T\,{}^1_2T\,{}^2_3T \begin{bmatrix} l_2 \\ -d_4\sin(\alpha_2) \\ d_4\cos(\alpha_2) \\ 1 \end{bmatrix}, \tag{2.13} $$

where:

$$ \begin{aligned} f_1 &= l_2\cos(\theta_3) + d_4\sin(\alpha_2)\sin(\theta_3) + l_1 \\ f_2 &= l_2\cos(\alpha_1)\sin(\theta_3) - d_4\sin(\alpha_2)\cos(\alpha_1)\cos(\theta_3) - d_4\sin(\alpha_1)\cos(\alpha_2) \\ &\quad - d_3\sin(\alpha_2) \\ f_3 &= l_2\sin(\alpha_1)\sin(\theta_3) - d_4\sin(\alpha_2)\sin(\alpha_1)\cos(\theta_3) + d_4\cos(\alpha_1)\cos(\alpha_2) \\ &\quad + d_3\cos(\alpha_1) \end{aligned} \tag{2.14} $$

Now, we can write Eq. (2.13) as:

$$ {}^0P_{4ORG} = \begin{bmatrix} \cos(\theta_1)\,g_1 - \sin(\theta_1)\,g_2 \\ \sin(\theta_1)\,g_1 + \cos(\theta_1)\,g_2 \\ g_3 \\ 1 \end{bmatrix}, \tag{2.15} $$

where

$$
\begin{aligned}
g_1 &= \cos(\theta_2)\, f_1 - \sin(\theta_2)\, f_2 + l_0 \\
g_2 &= \sin(\theta_2)\cos(\alpha_0)\, f_1 + \cos(\theta_2)\cos(\alpha_0)\, f_2 - \sin(\alpha_0)\, f_3 - d_2 \sin(\alpha_0) \\
g_3 &= \sin(\theta_2)\sin(\alpha_0)\, f_1 + \cos(\theta_2)\sin(\alpha_0)\, f_2 + \cos(\alpha_0)\, f_3 + d_2 \cos(\alpha_0)
\end{aligned}
\tag{2.16}
$$

Let r denote the squared magnitude of ${}^0P_{4ORG}$. So

$$r = x_d^2 + y_d^2 + z_d^2. \tag{2.17}$$

From Eq. (2.15), it can also be written as

$$r = g_1^2 + g_2^2 + g_3^2. \tag{2.18}$$

From Eqs. (2.15) and (2.16), r can be written differently as

$$r = (k_1 \cos(\theta_2) + k_2 \sin(\theta_2))2l_0 + k_3, \tag{2.19a}$$

and the z-component of Eq. (2.15) can be written as

$$z = (k_1 \sin(\theta_2) - k_2 \cos(\theta_2)) \sin(\alpha_0) + k_4, \tag{2.19b}$$

where

$$
\begin{aligned}
k_1 &= f_1 \\
k_2 &= -f_2 \\
k_3 &= f_1^2 + f_2^2 + f_3^2 + a_0^2 + d_2^2 + 2d_2 f_3 \\
k_4 &= f_3 \cos(\alpha_0) + d_2 \cos(\alpha_0)
\end{aligned}
\tag{2.20}
$$

Equation Eq. (2.19a) is very useful since it is independent from θ_1, and can be used to solve for θ_2 and θ_3. So to solve for these joint angles from Eq. (2.19a), we can have 3 interesting cases:

- $a_0 = 0$, then $r = k_3$, and is only a function of θ_3. It can therefore be solved by equating the right-hand-side (RHS) of Eq. (2.19a) with the RHS of Eq. (2.17), and substituting equations (2.11) into $\tan(\frac{\theta_3}{2})$.

- $\sin(\alpha_0) = 0$, then $z = k_4$, where z is z_d. It can therefore also be solved by equating the RHS of (2.19a) with z_d, and applying substitutions via equations (2.11).

- Otherwise, the following equation is used to eliminate $\sin(\theta_2)$ and $\cos(\theta_2)$ from Eq. (2.19a). Note that it can be solved also using the equations of (2.11):

$$\frac{(r-k_3)^2}{4l_0^2}+\frac{(z-k_4)^2}{\sin^2(\alpha_0)}=k_1^2+k_2^2 \tag{2.21}$$

After determining θ_3, Eqs. (2.15) and (2.19a) can easily be manipulated to solve for θ_2 and θ_1.

For a 3-DOF robot, the above procedure is sufficient for the IK solution, but for a 6-DOF, there still remains the wrist angles θ_4, θ_5, and θ_6. Clearly, these only affect the wrist's orientation. These can be solved by applying the Z-Y-Z Euler angle solution (APPENDIX C) for the following rotation matrix (Craig, 2005):

$${}^4_6R|_{\theta_4=0}={}^0_4R^{-1}|_{\theta_4=0}\,{}^0_6R \tag{2.22}$$

2.6.2 Existence of Solution

The IK solution does not always exist for a certain task, while several solutions might exist for another – EE poses can sometimes be reached by different joint angle combinations. So the question of IK solvability is of crucial importance in addressing the IK problem, whose answer lies roughly in the points and orientations a robot wants to reach, i.e. the robot's workspace.

2.7 Types of Workspace

When a robot is manipulated to all possible postures it can achieve, a volume is generated for the set of points that could be reached. This volume is called a manipulator's workspace. In fact, the workspace can be classified into two definitions: The *reachable workspace* is the volume of space that can be reached by the robot, whereas the *dexterous workspace* is the volume of space that can be reached by the robot with all orientations. The dexterous workspace is always a subset of the reachable

workspace (Craig, 2005). Typically, a 3-DOF robot can only satisfy x-y-z positioning tasks, thus only comprises of a reachable workspace, while a 6-DOF robot such as the PUMA 560 and the reconfigurable robot at hand have both a reachable and a dexterous workspace. It is therefore unsurprising to see that the PUMA 560 manipulator is one of the most employed robots in the industry.

2.8 Jacobian Matrix

It is important to note that the indices that will be used to study the performance of the manipulator are based on the Jacobian matrix, so we first recall this matrix and its relation to manipulator singularity.

The Jacobian in robotics is a multidimensional form of the derivative, relating joint velocities to Cartesian velocities of the tip of the arm (Craig, 2005). Hence it is necessary to study the linear and angular velocity propagation from one link to another in order to determine the Jacobian matrix. Craig summarized the equations of velocity propagation for prismatic and revolute joints. However we will only consider revolute joints for the matter of our study. These velocities are denoted by:

$$^{i+1}\omega_{i+1} = {}^{i+1}_{i}R\,{}^{i}\omega + \dot{\theta}_{i+1}\,{}^{i+1}\boldsymbol{Z}_{i+1} \tag{2.23}$$

and

$$^{i+1}v_{i+1} = {}^{i+1}_{i}R({}^{i}v_{i} + {}^{i}\omega_{i} \times {}^{i}P_{i+1}) \tag{2.24}$$

where $^{i+1}v_{i+1}$ and $^{i+1}\omega_{i+1}$ represent the linear and angular velocities of the corresponding link i+1 with respect to frame $\{i+1\}$, respectively. ${}^{i+1}_{i}R$ is the 3 x 3 rotation matrix of frame $\{i\}$ with respect to $\{i+1\}$, ${}^{i}P_{i+1}$is the position of $\{i+1\}$ relative to $\{i\}$, and i ranges from 0 to N-1. $\dot{\theta}_{i+1}$is the joint rate of frame $\{i+1\}$, and

$$\dot{\theta}_{i+1}{}^{i+1}\boldsymbol{Z}_{i+1} = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_{i+1} \end{bmatrix}. \tag{2.25}$$

Equations (2.23) and (2.24) are applied iteratively, resulting in linear and rotational velocities ${}^{N}\omega_{N}$ and ${}^{N}v_{N}$ and a total Cartesian velocity ${}^{N}\mathrm{v} = \begin{bmatrix} {}^{N}v_{N} \\ {}^{N}\omega_{N} \end{bmatrix}$ at EE frame $\{N\}$. To find the velocity in terms of the base frame $\{0\}$, we rotate it with ${}^{0}_{N}R$, such that

$${}^{0}\mathrm{v} = {}^{0}_{N}R\,{}^{N}\mathrm{v}. \tag{2.26}$$

Now for a general *n*-joint manipulator, the Jacobian matrix is given by

$$\mathrm{J}(\theta) = \begin{bmatrix} \mathrm{J_T} \\ \mathrm{J_R} \end{bmatrix} = \begin{bmatrix} \mathrm{J_{T,A}} & \mathrm{J_{T,W}} \\ \mathrm{J_{R,A}} & \mathrm{J_{R,W}} \end{bmatrix}, \tag{2.27}$$

where J is the *m* x *n* Jacobian, the subscripts "T" and "R" relating the joint velocity vector to the translational and rotational velocity of the end effector, respectively. Subscripts "A" and "W" refer to the manipulator's arm and wrist, respectively.

So in summary, the rows of the Jacobian matrix represent the Cartesian degrees of freedom, with a maximum of 6 Cartesian DOF – of which 3 are related to position and the other 3 are related to the rotation. As for the columns, each column represents a joint. So the total number of columns of the Jacobian matrix signifies the total DOF of the robot. For example, a 3 x 3 Jacobian matrix means that the robot at hand is only concerned with positioning tasks, and has 3 joints (DOF). A 6 x 6 Jacobian matrix indicates that the robot is concerned in both positioning and orienting tasks, and has 6 joints (DOF).

Note that the upper sub-matrix of J, which refers to the positioning sub-matrix, has units of length, while the lower sub-matrix, which refers to the rotational sub-matrix,

is dimensionless. This will cause some issues demanding some Jacobian homogenization in some index calculations, as will be seen in Chapter 3.

2.8.1 Jacobian Matrix in the Velocity Domain

The velocity of the EE which was written in Eq. (2.26) is related to the Jacobian by

$$ {}^{0}\mathrm{v} = \begin{bmatrix} {}^{0}v \\ {}^{0}\omega \end{bmatrix} = {}^{0}\mathrm{J}(\theta)\dot{\theta}, \tag{2.28} $$

where ${}^{0}v$ and ${}^{0}\omega$ are the linear and rotational velocities of the EE with respect to the base frame, respectively. For an *n*–DOF manipulator, ${}^{0}\mathrm{v}$ is an *m*-dimensional vector, ${}^{0}\mathrm{J}(\theta)$ is an *m* x *n* matrix relating the EE velocities to the manipulator's base frame, and $\dot{\theta}$ is an *m*-dimensional vector describing the joint rates. The number of actuated joints is *n*, and the number of Cartesian degrees of freedom is *m*. So in order to have a degree of redundancy, $n\text{-}m \geq 0$ should be satisfied.

2.8.2 Jacobian Matrix in the Force Domain

When a manipulator is statically pushing on something in the environment or supporting a load at the EE, it is then actuating joint torques in order to counterbalance these forces. In order to solve for these joint torques, the joints are locked first and the manipulator becomes a rigid structure at that posture. In the force domain, the Jacobian transpose maps the external Cartesian forces acting at the EE into the equivalent joint torques (Craig, 2005).

$$ \tau = {}^{0}\mathrm{J}^{T}\ {}^{0}\mathcal{F}\,, \tag{2.29} $$

where

$$^{0}\mathcal{F} = \begin{bmatrix} ^{0}F \\ ^{0}N \end{bmatrix}. \tag{2.30}$$

^{0}F and ^{0}N are the force and the moment vectors acting on the EE with respect to the manipulator's base frame of reference. Comparing Eq. (2.28) with Eq. (2.29), the Cartesian force and joint torques are in inverse relation with the Cartesian velocity and joint rates, respectively.

2.8.3 Singularities

Now consider that we wish to move the manipulator with a certain velocity in a certain direction, or on a certain trajectory. It is then necessary to solve for the joint rates, requiring the designer to determine the inverse of the Jacobian such that

$$\dot{\theta} = {}^{0}\mathrm{J}^{-1}(\theta)\,{}^{0}\mathrm{v}\,. \tag{2.31}$$

If the inverse of the Jacobian exists, then it can be inverted to calculate the joint rates at each step of the trajectory as in Eq. (2.31). But, if it is not invertible, the Jacobian becomes singular, and the manipulator becomes in a singular configuration. The Jacobian in this case has lost full rank, and the manipulator has lost one or more DOF in Cartesian space. As Craig indicated, singularities can be generally classed in two categories. They can occur on (a) the manipulator's workspace boundaries when the manipulator is fully stretched out or folded back on itself, and in (b) some areas inside the workspace when generally two virtual joint axes intersect.

In manipulators, singularity physically means that no matter which joint rates are selected, the EE cannot move in certain directions. Singularities also manifest themselves in the force domain, where the EE cannot exert static forces in certain directions. So $\mathcal{F}$ could be increased or decreased in these directions without any effect on the calculated τ

(Craig, 2005). Assuming that the Jacobian is a square matrix, det(J)=0 is a necessary and sufficient condition to indicate singularity. In the case that the Jacobian is an *m* x *n* rectangular matrix, singularity can be identified by $\det(JJ^T)=0$, as will be discussed in the next chapter.

2.9 Conclusion

In this chapter, the basics of manipulators and their kinematics were reviewed and extended to reconfigurable manipulators. Additionally, the Jacobian matrices in the velocity and the force domain are presented with their relation to joint rates and joint torques, and the matrix's significance in detecting singularity is exposed. This information will be required to model the robotic arm in the optimization methodology.

CHAPTER 3

PERFORMANCE MEASURES

3.1 Introduction

Different performance indices are presented in this chapter. The *manipulability measure* is used to study the distance from singularity and robot arm maneuverability, the *condition number* is utilized to evaluate the joint rate/torque uniformity, and the parameter of singularity combines both indices. This parameter is then modified, rendering it a global and posture independent index to assess overall robot performance.

3.2 Manipulability Measure

The manipulability measure (manipulability for brevity) is used to assess the manipulator's "closeness" to singularity. It measures the amount of velocity and force that the EE can exert in all directions, and the ability to position and re-orient the EE in different directions, giving greater freedom for the specific configuration.

The manipulability was first used by Paul and Stevenson (Paul & Stevenson, 1983), where the determinant of the Jacobian matrix was employed as a measure to evaluate the kinematic performance of spherical wrists.

3.2.1 The Manipulability Ellipsoid and Force Manipulability Ellipsoid

Yoshikawa further developed Paul and Stevenson's measure by defining the manipulability ellipsoid for non-square Jacobian matrices in (Yoshikawa, 1985a). He also expanded on this concept and presented the dynamic manipulability based on the determinant of the Jacobian and the nonsingular inertia matrix in (Yoshikawa, 1985b). For a given manipulator, the manipulability at a certain state θ was given by

$$W = \sqrt{\det(J.J^T)}, \tag{3.1}$$

where W is the manipulability, representing the volume of a manipulability ellipsoid. For a square Jacobian matrix with *m*=*n*, the manipulability reduces to

$$W = |\det(J)|. \tag{3.2}$$

In order to have a clear view for the concept of manipulability ellipsoids, one can consider the input joint rate space $\dot{\boldsymbol{\theta}} = (\dot{\theta}_1, \dot{\theta}_2, \ldots, \dot{\theta}_n)$, and the corresponding output EE velocity space $\dot{\boldsymbol{x}} = (\dot{x}_1, \dot{x}_2, \ldots, \dot{x}_m)$. We now define an *n*-dimensional unit sphere of

$$|\dot{\boldsymbol{\theta}}| = \sqrt{\dot{\theta}_1^2 + \dot{\theta}_2^2 + \cdots + \dot{\theta}_n^2} \leq 1. \tag{3.3}$$

and its mapping into an m-dimensional ellipsoid of $|\dot{\mathbf{x}}|$ (Figure 3.1) – the velocity or manipulability ellipsoid – by a relation containing the Jacobian (Pham & Chen, 2003), such that

$$\dot{\boldsymbol{x}}^{\boldsymbol{T}}(J\,J^T)^{-1}\dot{\boldsymbol{x}} \leq 1. \tag{3.4}$$

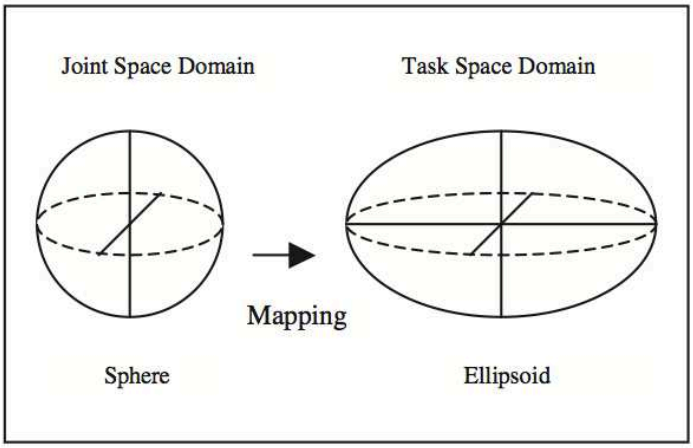


Figure 3.1 Joint space mapping into EE space, A.K.A. the manipulability ellipsoid. (Pham & Chen, 2003)

This relation generates an ellipsoid, whose axes are defined by the eigenvectors of $\mathrm{J}\,\mathrm{J}^{\mathrm{T}}$, and the lengths of the ellipsoid axes have the values of the square roots of the eigenvalues (Park & Kim, 1998). The directions in which the ellipsoid has major axes means a high manipulability or ability for movement at higher speeds, while the minor axes indicate difficulty in manipulation (Pham & Chen, 2003). In fact, when a minor axis disappears, we have singularity, and the EE cannot move in a certain direction no matter which joint rates are applied (Angeles, 2007). As a general rule, analysis of the volume and shape of the ellipsoid can be used to assess the "amplification" between the actuated joint rate and the EE velocity (Pham & Chen, 2003).

Analogous to the manipulability ellipsoid, a *force manipulability ellipsoid* can be derived for illustrating the force transmission and amplification from the joint torques to the EE. We have seen in Eq. (2.29) that $\tau = \mathrm{J}^{T}\ \mathcal{F}$. Then assuming the manipulator is not at a singular configuration one can write $\mathcal{F} = \mathrm{J}^{-T}\tau$. The force manipulability ellipsoid is therefore determined by the eigenvectors and eigenvalues of $(\mathrm{J}\,\mathrm{J}^{\mathrm{T}})^{-1}$; its eigenvectors will still be the same, but the eigenvalues will now have the inverse values of those of $(\mathrm{J}\,\mathrm{J}^{\mathrm{T}})$ (Park & Kim, 1998). This means that the direction that has a maximum velocity transmission ratio is itself the direction of minimum force transmission ratio, and vice versa. In other words, the volume of the manipulating force ellipsoid is in inverse proportion to that of the manipulability ellipsoid (Yoshikawa, 1985a) (Zargarbashi, Khan, & Angeles, 2012). So, the manipulability index should be minimized when it is desired to maximize the force at the EE.

So, in order to properly design a robot, a careful compromise must be made between these two ellipsoids (Pham & Chen, 2003).

3.2.2 Singular Value Decomposition

Performing a Singular Value Decomposition (SVD) to any *m* x *n* Jacobian matrix, the singular values of J can be determined (Elkady, Mohammed, & Sobh, 2009):

$$J_{mxn} = U_{mxm}\Sigma_{mxn}V^{T}_{nxn} \tag{3.5}$$

Where

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 & 0 & \cdots & 0 & 0 \cdots 0 \\ 0 & \sigma_2 & 0 & \cdots & 0 & 0 \cdots 0 \\ 0 & 0 & \ddots & 0 & \cdots & 0 \cdots 0 \\ 0 & 0 & \cdots & \sigma_{m-1} & 0 & 0 \cdots 0 \\ 0 & 0 & \cdots & 0 & \sigma_m & 0 \cdots 0 \end{pmatrix}, \tag{3.6}$$

such that U and V are orthogonal matrices (i.e. the product of each of U or V by its transpose gives the identity matrix), and $\sigma_1, \sigma_2 \ldots, \sigma_m$ are called the singular values of J, sorted in descending order such that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_m$. The singular values can also be determined by the set of eigenvalues of JJ^T.

3.2.3 Minimum Singular Value

When the manipulator is far from singularity, J has a full rank *m*, meaning the smallest singular value is different from zero ($\sigma_m \neq 0$). But when the manipulator is close to singularity, σ_m approaches zero. In that case J does not have full rank and it loses one or more degrees of freedom.

On the other hand, the determinant of J can also be expressed as a product of the matrix's singular values:

$$\det(J) = \sigma_1\ \sigma_2\ \ldots \sigma_m\ . \tag{3.7}$$

Evidently, this product becomes close to zero when σ_m is very small. A remarkable property for σ_m is that it becomes most sensitive near singularities, making it the dominant parameter. For this reason, Klein and Blaho (Klein & Blaho, 1987) deduced

that the *minimum singular value* σ_m alone could be used as a performance index. Physically, using σ_m to evaluate the robot's manipulation ability measures the upper bound of the velocity with which the EE can be moved in all directions (Yoshikawa, 1985a).

3.2.4 Manipulators with Spherical Wrists

Manipulators with spherical wrists (Figure 3.2) are those whose joints are all revolute and their axes intersect at a point. These manipulators are most utilized in the industry because of their singularity decoupling ability (Yoshikawa, 1990) (Tourassis & Ang Jr, 1995). Also, Pieper pioneered the use of decoupling by showing that the IK problem of manipulators with three consecutive intersecting axes can be divided into two sub-problems, whose solutions were obtained through a closed-form (Tourassis & Ang Jr, 1995).

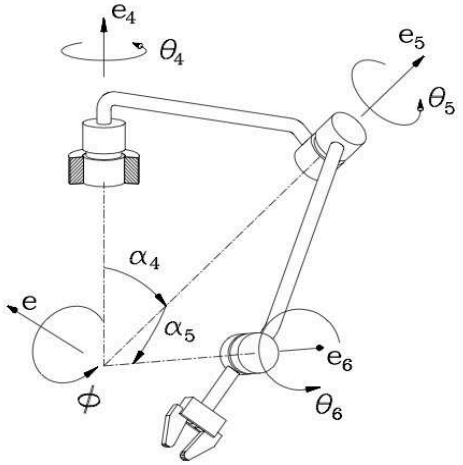


Figure 3.2 General architecture of a spherical wrist.

An interesting property in the manipulability measure for manipulators with spherical wrists is that the Jacobian sub-matrix responsible for the translational part of the wrist becomes zero, so the Jacobian matrix takes the following form (Yoshikawa, 1990):

$$J(\theta) = \begin{bmatrix} J_T \\ J_R \end{bmatrix} = \begin{bmatrix} I_3 & -[p_w\times] \\ 0 & I_3 \end{bmatrix} \begin{bmatrix} J_{T,A} & 0 \\ J_{R,A} & J_{R,W} \end{bmatrix}. \quad (3.8)$$

For a 6-DOF robot, I_3 is a 3x3 identity matrix, and p_w is the vector from the point where the wrist's joint axes intersect to the center point of the end effector. As for the operator $[a\ \times]$, it is defined as the matrix form of the cross-product such that:

$$[a\ \times] = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}. \quad (3.9)$$

The determinant of J in this case can be decoupled (Yoshikawa, 1990), and it becomes equivalent to

$$\det(J) = \det(J_{T,A}) \cdot \det(J_{R,W}) . \quad (3.10)$$

Notice that the determinant of J becomes independent of $J_{R,A}$. For a 6-DOF (positioning and orienting) manipulator, $J_{T,A}$ and $J_{R,W}$ are 3x3 sub-matrices. Notice also that the elements in J have different units. Even though the sub-matrix $J_{T,A}$ possesses units of length, while $J_{R,W}$ is dimensionless, det(J) can be directly computed without any homogenization of the Jacobian, a property that cannot be implemented with the condition number, as will be seen later (Angeles, 2007). It is worthwhile to say that for a 6-DOF robot, decoupling is *guaranteed*, given that any three of its rotational joints intersect at a common point (Tourassis & Ang Jr, 1995).

3.2.5 Manipulability Issues

Although the determinant going to zero means singularity, the real value of the determinant of a matrix is not a practical measure of the degree of the matrix's ill-conditioning, since the manipulability index generates different values for different used units. Also, Angeles and Gosselin (Angeles & Lopez-Cajun, 1992) argued that, "as

pointed out by Forsythe and Moler (1967), the absolute value of the determinant of a matrix is meaningless in assessing the inevitability of the matrix". Angeles then gave an example of a matrix with a large determinant, but inverting it with finite precision led to inadmissibly large round-off errors (Angeles & Lopez-Cajun, 1992) (Gosselin, 1990) (Klein & Blaho, 1987) (Forsythe & Moler, 1967).

Furthermore, the manipulability is strongly dependent on the robot's dimensions. Kim and Khosla (Kim & Khosla, 1993) claimed that the manipulability faces two problems: scale and order dependency, preventing fair comparison of different sizes of robots, as well as comparison of planar and spatial robots. They proposed to divide the manipulability by the square of the total manipulator's link lengths, and then evaluate the m^{th} root (m being the DOF) of the resulting manipulability, thereby eliminating both problems (Kim & Khosla, 1993). On the other hand, Tanev and Stoyanov suggested dividing the manipulability by the maximum manipulability within the workspace of the robot, bounding the resultant index between zero and one, and consequently normalizing the manipulability (Tanev & Stoyanov, 2000). Their resultant index, which is termed as the normalized mobility index, will be used later in the formulation of the objective function.

3.2.6 Other Studies on Manipulability

An optimized form of the manipulability index based on the singular values of the manipulator's Jacobian matrix was proposed in (Elkady, Mohammed, & Sobh, 2009). This method was based on identifying the joints that may lead to singular configurations, varying their values between the lower and upper limits, and calculating the Jacobian J and the singular matrices (Σ) at each step. Each singular value was then normalized,

plotted, and analyzed. The rank of the Jacobian was also checked in each step.

The concept of manipulability has been extensively studied and applied in robotic applications, such as in (Chang, 1988), (Kim & Khosla, 1993), (Lee, 1997), (Pham & Chen, 2003), (Sobh & Toundykov, 2004), (Merlet, 2006a), and (Elkady, Mohammed, & Sobh, 2009), (Zacharias, 2012).

3.3 Condition Number

The condition number is the ratio of the largest to smallest singular values of the Jacobian matrix, measuring the manipulator's *isotropy*, meaning the uniformity of manipulability in all directions. This also means the uniformity of the joint rate or torque distribution over a time history (Zargarbashi, Khan, & Angeles, 2012), and the EE's ability to move or apply forces uniformly in all directions (Yahya, Moghavvemi, & Mohamed, 2012). It basically gives information about the accuracy of a vector $\boldsymbol{x}$ when solving a system of equations $A.\boldsymbol{x}=b$, and a good explication of this significance is provided in APPENDIX A (Keesling). In robotics, when applied to the Jacobian matrix J, the condition number will give a measure of the accuracy of the Cartesian velocity v or general static force $\mathcal{F}$ acting on the EE (Salisbury & Craig, 1982) (Zargarbashi, Khan, & Angeles, 2012).

Other than the accuracy, the condition number describes the robot's dexterity, and the closeness of a pose to singularity (Merlet, 2006a).

The condition number represents the uniformity of the manipulability ellipsoid, as compared to the volume of the ellipsoid in manipulability. In robotics, it was first used by Salisbury and Craig for the design of the Stanford-JPL mechanical fingers (Salisbury & Craig, 1982). The best-conditioned manipulators were termed to be in "*isotropic*"

configurations, while the worst-conditioned were singular. Represented on the manipulability ellipsoid, an isotropy maps unit spheres in the space of joint rates to spheres of arbitrary radius in the space of EE velocity (Park & Kim, 1998).

Given that all entries of the Jacobian matrix bear the same units, the condition number is defined by

$$\kappa(\mathrm{J}) = \frac{\sigma_{Max}}{\sigma_{min}} = \frac{\sigma_1}{\sigma_n} \tag{3.11}$$

where σ_1 and σ_n are the maximum and minimum singular values, respectively. Alternatively as presented in (Gosselin, 1990) and (Angeles & Lopez-Cajun, 1992), the condition number can be denoted by

$$\kappa(\mathrm{J}) = \|\mathrm{J}\|.\|\mathrm{J}^{-1}\|, \tag{3.12}$$

where $\|\,.\,\|$ is any norm of the matrix J. Note that out of all norms, only the Euclidean norm (2-norm), and the Frobenius norm are frame-invariant, meaning that they are independent of the base joint angle θ_1. Also intuitionally, if a robot was rotated about its base joint while all other joints were fixed at the same pose, this makes it the same as looking at the robot from different viewpoints, which should not affect any of the manipulability or isotropy measures. Hence, using the 2-norm or Frobenius norm significantly simplifies calculations (Zargarbashi, Khan, & Angeles, 2012).

Moreover, the 2-norm requires an iterative singular value decomposition since it is defined as the maximum singular value of the matrix at hand. This is bound to cause a heavy computational load on the process. The Frobenius norm on the other hand, is an analytical function of the Jacobian matrix itself, as will be shown below, hence does not require singular value decomposition. For these reasons, the Frobenius norm will be used later in the formulation of the objective function (Zargarbashi, Khan, & Angeles, 2012).

A mathematical definition for the Euclidean and Frobenius norms is presented in APPENDIX B (Horn & Johnson, 1990), (Golub & Van Loan, 2012), (Watrous, 2011).

The Frobenius norm of a square matrix J is utilized by several researchers, such as in (Angeles & Lopez-Cajun, 1992), (Gosselin, 1990), and (Gosselin & Angeles, 1991), and (Zargarbashi, Khan, & Angeles, 2012), and is denoted by

$$\|\mathrm{J}\| = \sqrt{tr(\mathrm{J}^T M \mathrm{J})}\ , \tag{3.13}$$

where tr(.) is the trace of a matrix, and M is a positive definite matrix, used for dimensional homogenization and normalization purposes (Angeles & Lopez-Cajun, 1992). Here M is a scalar, equivalent to the identity matrix divided by the robot's degrees of freedom, as was used by Gosselin in (Gosselin, 1990).

3.3.1 Positioning or orienting tasks vs. Positioning and orienting tasks

As mentioned earlier, calculating the determinant of the Jacobian matrix poses no problem for manipulators with positioning and orienting tasks (6-DOF), although the Jacobian bears different units. The calculation of the Jacobian condition number can be directly applied only if the Jacobian elements bear the same units, i.e. if the robot at hand is restricted to operate only on positioning or on orienting tasks. However, if the robot is meant to perform both tasks, the Jacobian entries possess disparate units, making the calculation of the condition number impossible. Considering a 6-DOF manipulator for instance, in order to determine the manipulator's condition number we need to order the singular values in descending order, from largest to smallest. Three of these singular values have units of length, while those corresponding to orientation are dimensionless. Therefore it does not make sense to divide the largest singular value by the smallest one.

Consequently, the Jacobian matrix has to be homogenized for compatible units before computing its condition number.

3.3.2 Homogenizing the Jacobian Matrix

In order to avoid dimensional inconsistency, Angeles (Angeles, 2007) defined a homogenous Jacobian matrix by introducing a *characteristic length* to divide the rows of the Jacobian with units of length. The *characteristic length* is defined as the normalizing length that generates the minimum condition number of the Jacobian matrix (Angeles, 2007). Gosselin (Gosselin, 1990) used a different approach based on defining the velocity of the EE as the velocity of two or more points on it instead of using the velocity of one point only together with the angular velocity. He then proposed two indices for planar manipulators and derived them for spatial manipulators. One index was based on a redundant formulation of the velocity equations, while the other on the minimum number of parameters of these equations.

The condition number $\kappa(\mathrm{J})$ ranges from 1 to $+\infty$, where 1 implies an isotropic configuration and $+\infty$ singularity. Alternatively, one can calculate the inverse of the condition number, thus bounding it between 0 (singularity) and 1 (*isotropy*).

3.3.3 Performance Indices as Local and Posture-Dependent vs. Global and Posture-Independent

Kinetostatic performance indices are becoming increasingly reliable as design criteria and as figures of merit in robot control. They can be used to either assess a manipulator's performance while performing a task, or in the design process of robots. In the former case, they are used as local posture-dependent criteria to evaluate the invertibility of Jacobian matrix at different postures, whereas in the design stage, they are

implemented as global posture-independent criteria (Angeles, 2007).

In our case, it is desired to define a posture-independent index. Therefore this index has to be defined in a global sense. This can be done in a similar way as calculating the sum of the squares of the vector's components. The global performance index can also be defined by using the Chebyshev norm, i.e. the maximum performance attained over the whole workspace (APPENDIX B) (Horn & Johnson, 1990) (Bock & Krischer, 1998). Alternatively, the Euclidean norm for vectors can be used, i.e. the root-mean square (rms) value of the performance (Angeles, 2007). However, in this thesis we will use the robot's worst-case performance attained over its desired workspace as a global index, as will be clarified in Section 3.4.

3.3.4 Other Indices Based on the Condition Number

A posture-independent Kinematic Conditioning Index (KCI) of a manipulator, based on the inverse of the minimum condition number ($1/\kappa_{\min}$) in the whole workspace, was introduced in (Angeles & Lopez-Cajun, 1992). This makes it a global measure of the condition number, ranging between 0 and 1 (or 0% and 100%, for this matter). The KCI assesses the robot's measure of *isotropy* based on the best-or minimum-conditioned point within its workspace.

A Global Conditioning Index (GCI) that measured the robot's conditioning over the entire workspace was also explained in (Gosselin & Angeles, 1991). It is denoted by

$$\eta = \frac{A}{B}\ , \tag{3.14}$$

where

$$A = \int_W \left(\frac{1}{\kappa}\right) dW, \tag{3.15}$$

and

$$B = \int_W dW\,, \tag{3.16}$$

where κ is the condition number at a particular point of W, the manipulator's workspace, and B is clearly the volume of the workspace. Thus the GCI is a measure of the Jacobian's invertibility over the whole workspace. Note that the concept of a global index to characterize the kinematic or dynamic properties of manipulators could be used with local indices different than the condition number (Gosselin & Angeles, 1991), such as the manipulability index. This fact will be introduced in the *parameter of singularity* in the next subsection, making it a global index.

Finally, spatial robotic isotropy, which is the isotropy for both positioning and orienting the EE, was discussed by (Klein & Miklos, 1991). A more detailed discussion about the condition number can be found in (Angeles, 2007) and (Zargarbashi, Khan, & Angeles, 2012).

3.4 Combining Indices

The manipulability and condition number possess some common features. They both indicate the presence of singularity, and their absolute values appear to represent the distance from singularity (Chang, 1988). On the other hand, as already noticed, the aforementioned indices have different purposes (Klein & Blaho, 1987) (Tanev & Stoyanov, 2000), (Merlet, 2006a) (Zacharias, 2012). Ultimately, the manipulability measures the volume of the manipulability ellipsoid, and the condition number measures the uniformity of its shape. Therefore a performance index representing both the volume

and the shape is required for the manipulability ellipsoid to be better represented. Huo and Baron (Huo & Baron, 2007), (Huo & Baron, 2008), (Huo & Baron, 2011), proposed the *parameter of singularity* for the trajectory optimization of a redundant 6-DOF robot. Yahya, Moghavvemi, and Mohamed (Yahya, Moghavvemi, & Mohamed, 2012), then used this index on a manipulator with extra degrees of freedom, to improve its ability for singularity avoidance, thus making it more maneuverable. The *parameter of singularity* is given by

$$\mathrm{W_{PS}} = \sqrt{\frac{\kappa}{\mathrm{W}}} = \sqrt{\frac{1}{\sigma_2 \sigma_3 \dots \sigma_m{}^2}} \ , \tag{3.17}$$

where κ is the condition number of J, W is the manipulability index, and σ_i is a singular value of J. In order to eliminate the manipulability's scale dependency, we slightly modified the above index by introducing a normalization factor. The manipulability is divided by the maximum manipulability in the workspace (W_{max}), thus bounding the $\mathrm{W}/\mathrm{W}_{max}$ ratio between 0 and 1.

The resultant index to be *minimized* is

$$\mathrm{W_{PS}} = \sqrt{\frac{\kappa}{\mathrm{W}/\mathrm{W}_{max}}} \ , \tag{3.18}$$

and only has a lower bound of 1. As $\mathrm{W_{PS}}$ get smaller, the manipulability ellipsoid gets more spherical and its volume gets larger, resulting in EE velocity maximization. As $\mathrm{W_{PS}}$ gets larger, both the manipulator's conditioning and performance gets worse (Yahya, Moghavvemi, & Mohamed, 2012).

Note that the reciprocal of $\mathrm{W_{PS}}$ can alternatively be used, and is bounded between 0 and 1.

Being dependent on both the manipulability and the condition number, W_{PS} possesses the ability to assess several criteria including our criteria of interest. However, this index in its current state is posture and task dependent, reliant on the joint angles. It is required to define a posture-independent index, so that it can be represented it in the global sense (Angeles, 2007). As mentioned earlier, this index is chosen to be the robot's worst-case performance in the desired workspace. Optimizing the worst case guarantees that the performance criteria for all points on the workspace are met, hence creating a global and posture-independent index. This posture-independent index will be denoted by $\mathcal{W}_{PS}$, termed as the MPIPS.

3.4.1 Maximizing EE Force and Maintaining Actuator Rate/Torque Uniformity

In summary, by finding an optimized set of design parameters that minimizes $\mathcal{W}_{PS}$, this effectively (a) maximizes the robot's manipulability and (b) minimizes the condition number (enhances robot isotropy). Mechanically this means that the velocity at the EE is maximized, and the joint rates and torques are maintained to be as uniform as possible, as explained by (Huo & Baron, 2007) (Huo & Baron, 2008) (Huo & Baron, 2011), and (Yahya, Moghavvemi, & Mohamed, 2012).

Now, consider the case that the force manipulability ellipsoid is desired to be maximized instead of the velocity, while maintaining actuator rate and torque uniformity. In order to fulfill force maximization, the manipulability index in (3.18) should be minimized. As for joint rate and torque uniformity, the condition number of (3.18) should also be minimized. One can consider *maximizing* $\mathcal{W}_{PS}$ of (3.18) to solve the problem. However, this would maximize the condition number. Thus, the objective function has to

be necessarily altered as to just invert the condition number κ in (3.18), and multiply the square root by -1, resulting in the following index, to be *minimized*:

$$\mathrm{W}_{\mathrm{PS}-\mathcal{F}} = -\sqrt{\frac{1/\kappa}{\mathrm{W}/\mathrm{W}_{max}}} \ . \tag{3.19}$$

As $|\mathrm{W}_{\mathrm{PS}-\mathcal{F}}|$ gets larger, the *force manipulability ellipsoid* gets more spherical and its volume gets larger, resulting in EE force maximization. As $|\mathrm{W}_{\mathrm{PS}-\mathcal{F}}|$ gets smaller, both the manipulator's conditioning and performance gets worse (Yahya, Moghavvemi, & Mohamed, 2012).

Analogously to $\mathcal{W}_{\mathrm{PS}}$, if it is required to define a global posture-independent index, the robot's worst-case performance – or the largest value of $\mathrm{W}_{\mathrm{PS}-\mathcal{F}}$ (smallest $|\mathrm{W}_{\mathrm{PS}-\mathcal{F}}|$) in the desired workspace – should be chosen in the optimization approach to be the global index. We will denote this index by $\mathcal{W}_{PS-\mathcal{F}}$.

3.5 Conclusion

In this chapter we presented different performance indices that are used to evaluate the kinematic performance of robots, and a preliminary discussion on force domain of manipulability ellipsoid is also explained. A combination of the manipulability and isotropy measures was presented, then modified and implemented in a global and posture-independent sense by selecting the worst-case performance in the workspace, assessing the overall robot's performance.

CHAPTER 4

OPTIMIZATION METHODOLOGY

4.1 Introduction

Before stating the optimization methodology, it is of vital importance to discuss the assumptions considered in simplifying the robot model, and the numerical methods chosen to perform this optimization. Here, first the assumptions and selected numerical optimization techniques will be discussed and then, the optimization strategy is presented.

4.2 Assumptions

We have proposed some assumptions for determining the unknown DH parameters of the manipulator. These will help to simplify the computations, without majorly affecting the credibility of the model or the results of the optimization. It is assumed that:

- The robot is composed of massless rigid links.
- The workspace in which the robot is working is free of any obstacles, and the robot is assumed to not have any collisions within its own body.
- The 6-DOF manipulator's EE is a spherical wrist with its three joint axes intersecting at one point. As pointed out earlier, this helps in separating the IK position from the IK orientation problem, enabling to solve both problems explicitly using Pieper's method (Pieper, 1968), (Craig, 2005), (Angeles, 2007). Decoupling also permits singularity decoupling (Tourassis & Ang Jr, 1995), allowing the use of equation 3.8.

- The joint offsets at the spherical wrist of the 6-DOF manipulator, and between the wrist and the positioning manipulator, are all zero. This permits the wrist to control only the orientation of the EE, prohibiting it from having any effect on the position. As a result, the upper-right 3x3 sub-matrix of the Jacobian ($J_{T,W}$) will possess only entries of zeros.

4.3 Optimization Methods

4.3.1 Optimization Formulation

A formal optimization problem is defined as to find a design variable vector **x** that minimizes a certain *cost function* (or objective function) $f(\mathbf{x})$ subject to p equality constraints $c_j(\mathbf{x}) = 0$, $j = 1$ to p, and/or m inequality constraints $d_k(\mathbf{x}) \leq 0$, $k = 1$ to m. The statement can be translated to:

$$\begin{aligned} &\text{minimize} \quad f(\mathbf{x}) \\ &\text{subject to:} \quad \begin{cases} \boldsymbol{c_j}(\mathbf{x}) = 0, & j = 1 \text{ to } p \\ \boldsymbol{d_k}(\mathbf{x}) \leq 0, & k = 1 \text{ to } m \end{cases} \end{aligned} \tag{4.1}$$

Note that the constraints must have a relation with the design variables (**x**), and the formulation of the problem depends on these variables that describe the system (Arora, 2004). In the case that a cost function is to be maximized, it can be inverted and minimized, or better yet multiplied by a negative sign and minimized.

4.3.2 Numerical Optimization Methods

The optimization problem stated in Eq. (4.1) may generally be addressed analytically using the Karush–Kuhn–Tucker (KKT) conditions (APPENDIX C), but except for very simple problems, it is not possible to find optimal solutions. Thus, various numerical techniques have emerged. Among these techniques are stochastic based

algorithms and gradient-based algorithms. The former are prone to generating solutions near the global minimum, while the latter converge at precise but local minima. As a result, one might use a combined stochastic and gradient-based algorithm to generate precise global minima, which is the case of interest.

We discuss two numerical techniques that were implemented in our optimization, a stochastic algorithm (GA), and a gradient-based nonlinear optimization algorithm based on SQP, providing an understanding as to why combined GA and SQP were chosen to perform the optimization.

Genetic Algorithm

Genetic algorithm (GA) is a global optimization method that simulates a process based on natural evolution, natural selection, and natural genetics. It is based on the genetic reproduction process where only the fittest shall survive (Belegundu & Chandrupatla, 2011). As compared to conventional techniques, GA evaluates many individuals by generating random populations (design variables) in parallel and does not need continuity or differentiability of the objective function. Although GA is used in global optimization problems, its convergence is relatively slow, requiring a large amount of calculation for even small-scale problems, and it cannot absolutely guarantee a global convergence. The former can be solved by the exploitation of parallel processing, while running several simulations and choosing the best result can solve the latter.

After GA evaluates random populations at an iteration, it creates a mating pool through the reproduction process where the stronger members (lower fitness function values) replace the weaker ones. Afterwards, a crossover operation exchanges the characteristics between the members of the new population (Arora, 2004).

GA will be necessarily used in our optimization methodology, because global minima are required.

Sequential Quadratic Programming

Sequential Quadratic Programming (SQP) is a set of nonlinear programming (NLP) methods, where the constraints or the objective cost function are nonlinear, which solve an optimization problem by a sequence of optimization sub-problems.

Starting from an initial design parameter guess $\mathbf{x}_0$, SQP approximates $f(\mathbf{x})$ by a Quadratic Programming (QP) sub-problem. The solution of the QP sub-problem is based on a quadratic approximation of the Lagrangian function (APPENDIX C), and is then used to determine a search direction for a line search process and construct a new iterate. This construction is done until the QP converges to a local minimum of the NLP. A major advantage for SQP is that the neither the initial point nor the iterates have to be feasible points, since calculating a feasible point in a complicated constraint function might be very difficult.

Note that, if the SQP problem is unconstrained, it then reduces to Newton's method that finds a point that vanishes the gradient of the objective function.

SQP has been extensively tested and implemented (Arora, 2004), outperforming other methods in terms of *precision* and efficiency. However, it converges to local minima, decreasing the *accuracy* of the algorithm as the number of variables increases (see APPENDIX C for a definition of accuracy vs. precision).

Combined GA and SQP Algorithms

The implementation of GA yields approximate global solutions, while SQP yields precise local solutions, thus an effective way to accurately catch the global optimum

solution is to incorporate both methods sequentially. Hereafter generating a solution from GA, the obtained approximate global optimal solution is substituted in SQP to achieve a precise global minimum. An example will be provided in the case studies to prove the necessity of GA in finding global solutions.

4.4 Optimization Strategy

It is proposed to perform the optimization using the following strategy:

1. Describe a workspace by specifying a set of points, and orientations if necessary, in space. The workspace can be an arbitrary volume taking the form of a sphere, cube, cylinder, or any other shape.

2. Identify the design parameters of interest and define a manipulator using DH parameters in terms of the unknown parameters. Then, generate the kinematic equations for the manipulator.

The design parameters in this study are the link twists α_1, α_2, and link lengths l_1, l_2, as shown in Figure 2.3.

The DH parameters for the 6-DOF manipulator are shown in Table 2.1. If it is desired to consider a 3-DOF robot which performs positioning tasks only, the first three rows of this DH table represent this robot.

The resulting link-transformation matrices are described in equations (2.3)—(2.8), calculated from (2.1) for i ranging from 1 to n. There will be 6 matrices for a 6-DOF reconfigurable manipulator, and 3 for a 3-DOF manipulator. The matrices of latter case are equivalent to the first 3 transformations of the 6-DOF case. The resultant forward kinematic equations are calculated through (2.2), depending on a 6-DOF (n=6) or a 3-DOF (n=3) case.

3. Solve for an explicit closed-form IK solution for the points in space as a function of the design parameters, using Pieper's method. By substituting the symbolic solutions of the joint angles in the objective function, their presence is eliminated from the function, making it only dependent on four variables: α_1, α_2, l_1, and l_2.

4. Formulate the optimization problem based on the MPIPS:

$$\min\{\, \mathcal{W}_{\mathrm{PS}} (\alpha_{i-1}, a_{i-1}, \theta_i, \ldots, \alpha_{n-1}, a_{n-1}, \theta_n)\}$$

$$\text{subject to: } \begin{cases} \boldsymbol{c_j}(x) = 0 \\ \boldsymbol{c_k}(x) \le 0 \\ \boldsymbol{lb} \le \boldsymbol{x} \le \boldsymbol{ub} \end{cases}$$

where α, a, and θ are the DH parameters representing the link twists, link lengths, and joint angles, respectively. The subscript $i = 1$ and $n =$ DOF. $\boldsymbol{c_j}(x)$ and $\boldsymbol{c_k}(x)$ are possible linear or non-linear constraints limiting the joint angles, joint rates and joint torques, as will be explained thereafter. $\boldsymbol{lb}$ and $\boldsymbol{ub}$ are the lower and upper bounds of the design parameters ($\boldsymbol{lb} \le \{l_1, \alpha_1, l_2, \alpha_2\} \le \boldsymbol{ub}$), respectively, and $\mathcal{W}_{\mathrm{PS}}$ is the objective cost function to be minimized over the predefined workspace volume. Note that the workspace volume is not a constraint for the objective function, but rather a means for obtaining the IK solution in terms of the design parameters, which is itself constrained by $\boldsymbol{c_j}(x)$ and $\boldsymbol{c_k}(x)$.

Now, the inequality constraint $\boldsymbol{c_j}(x)$ limits the joint-angle range, which is a property of the motor design, and the joint rates and joint torques, which are properties of the motor power and capacity.

The only equality constraint $\boldsymbol{c_k}(x)$ is the one assuring that the IK solution ($\boldsymbol{\theta_{IK}}$) is real. Note that, although Pieper's method generates an IK solution, it cannot differentiate between real and imaginary solutions, as sometimes only imaginary solutions exist. So an

equality constraint assuring that only the real solution is chosen is necessary. The optimization constraints are tabulated in Table 4.1.

Finally, as for the situation where the force at the EE is needed to be maximized, the only thing that is changed is the objective function, which becomes $\mathcal{W}_{\mathrm{PS}-\mathcal{F}}$ instead of $\mathcal{W}_{\mathrm{PS}}$.

Table 4.1 Optimization bounds and constraints

$c_j(x) = 0$	$isreal(\boldsymbol{\theta_{IK}}) = 1$
$c_k(x) \leq 0$	$\boldsymbol{\theta_{IK}} - \boldsymbol{\theta_{max}} \leq 0$
	$-\boldsymbol{\theta_{IK}} - \boldsymbol{\theta_{min}} \leq 0$
	$\dot{\boldsymbol{\theta}} - \dot{\boldsymbol{\theta}}_{\mathbf{max}} \leq 0$
	$-\dot{\boldsymbol{\theta}} - \dot{\boldsymbol{\theta}}_{\mathbf{min}} \leq 0$
	$\boldsymbol{\tau} - \boldsymbol{\tau}_{\mathbf{max}} \leq 0$
	$-\boldsymbol{\tau} - \boldsymbol{\tau}_{\mathbf{min}} \leq 0$

5. Solve the optimization problem using GA in order to approximate the value of the global solution. The GA algorithm mainly takes the fitness function, number of design parameters, lower and upper bounds on **x**, nonlinear constraint functions, and some pre-specified options as inputs.

6. The optimal solution obtained from GA is then forwarded as the initial solution to the Sequential Quadratic Programming (SQP) to achieve a precise global optimum point. It should be noted that both GA and SQP have been implemented in the MATLAB

environment. The proposed optimization methodology is summarized in the following flowchart:

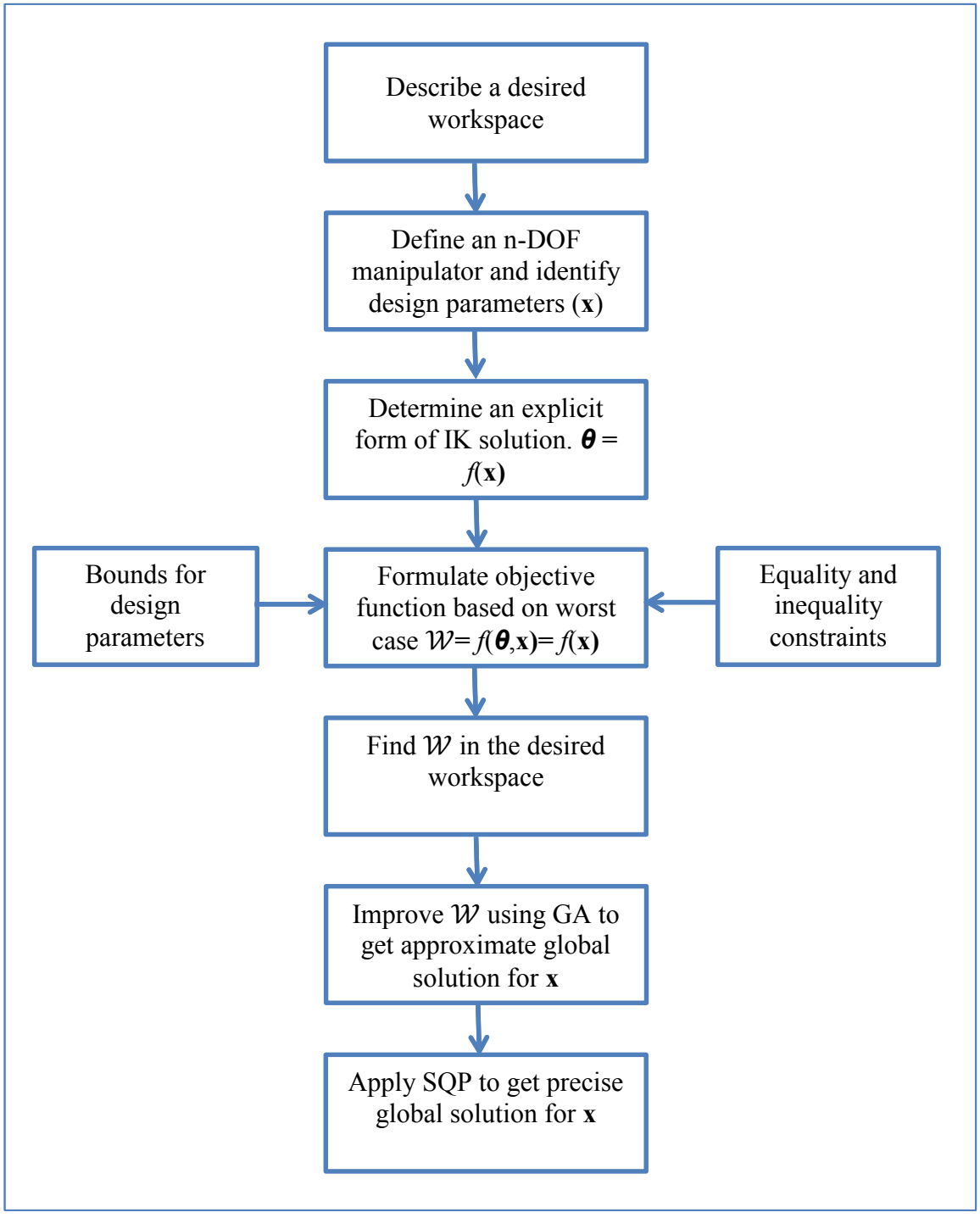


Figure 4.1 Flowchart for general optimization strategy.

4.5 Conclusion

In this chapter we provided the assumptions considered in simplifying the robot model, and presented a generalized methodology in solving the optimization problem. The formulation of the optimization problem was also discussed, and the use of numerical algorithms to solve this problem was justified. A combined stochastic algorithm (GA) and a gradient-based nonlinear optimization algorithm based on SQP were proposed to perform the optimization, and the optimization methodology was finally summarized in a flowchart.

CHAPTER 5

CASE STUDIES

5.1 Introduction

In this chapter, case studies will be presented for both 3-DOF (3R) and 6-DOF (6R) reconfigurable manipulators in different workspaces. First, we evaluate the global nature of optimization strategy using two case studies: (1) the first case studies the existence of multiple local points and the necessity of using combined GA and SQP to find accurate global optimum solution, and (2) the second case study deals with the validation of presented optimization strategy by comparing the optimal solution with that obtained from parametric study.

Afterwards, cases tackling the different performance criteria will be studied, mainly exploiting the use of the manipulability measure and condition number, for force/moment maximization capability at the manipulator's EE, linear/angular velocity maximization capability of the manipulator, and joint rate/torque uniformity. The cases concerning 3-DOF reconfigurable robots deal with the positional performance regardless of the EE's orientation in the workspace, while the 6-DOF cases consider the positional and orientational performance of the robotic manipulator.

For better representation purposes on the figures, the performance index $\mathcal{W}_{\mathrm{PS}}$ will be denoted by its inverse in order to bound it between 0 and 1 instead of 1 and ∞, as well as to represent singularity by a 0 instead of ∞. As for $\mathcal{W}_{\mathrm{PS}-\mathcal{F}}$, which is a negative quantity (See Eq. 3.19), it will be represented by its absolute value on figures. So, the larger the $|\mathcal{W}_{\mathrm{PS}-\mathcal{F}}|$, the better force performance of the robot. In summary, the

performance of all cases which minimize $\mathcal{W}_{\mathrm{PS}}$ is represented graphically as 1 / $\mathcal{W}_{\mathrm{PS}}$, and the performance of all cases which minimize $\mathcal{W}_{\mathrm{PS}-\mathcal{F}}$, is represented graphically as $|\mathcal{W}_{\mathrm{PS}-\mathcal{F}}|$.

5.2 Evaluation of Optimization Strategy

5.2.1 6-DOF Reconfigurable Manipulators

It is required to check if there are several local minima for the objective function $\mathcal{W}$ at hand, resulting in different design parameter solutions with different corresponding performances. If this is true, then SQP alone may not be a reliable method for our model, and GA implementation for catching global minima would be crucial. Optimization is thus performed on a 6-DOF robot with a workspace whose points are shown in Figure 5.1, and their orientations all possess a rotation matrix described by (0.5, -0.5, 0.5) Euler angles with respect to the base frame. The design parameters are bounded such that:

$$\begin{aligned} -90^{\mathrm{o}} \leq \alpha_1 \leq 90^{\mathrm{o}} \\ -90^{\mathrm{o}} \leq \alpha_2 \leq 90^{\mathrm{o}} \\ 5\mathrm{m} \leq l_1 \leq 9\mathrm{m} \\ 5\mathrm{m} \leq l_2 \leq 9\mathrm{m} \end{aligned} .$$

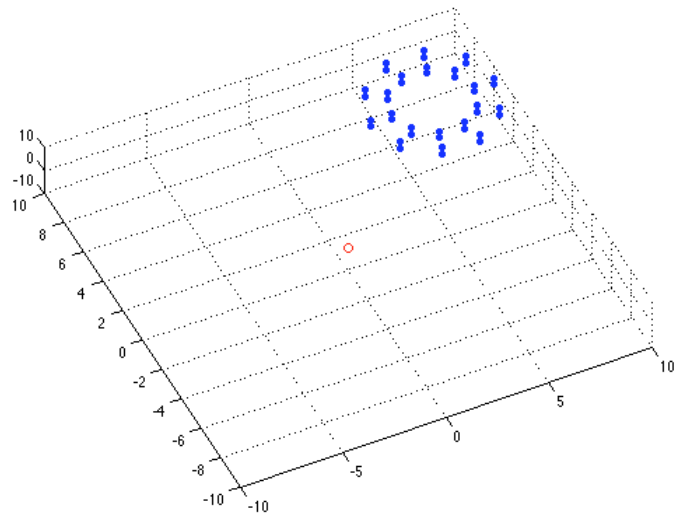


Figure 5.1 Set of points defining a cylindrical workspace. The red circle indicates the base of the robot.

Using different initial points, SQP has converged to different local optimum solutions, while combining GA and SQP has always converged to optimum solutions regardless of starting initial point, confirming that global optimum solution has been found. Results have been summarized in Table 5.1.

Table 5.1 Robot performance corresponding to SQP and combined GA and SQP optimization methods.

Initial Guess (x_0)	**Method**	**Optimized parameters (x)**	$1 / \mathcal{W}_{PS}$
[0.0 0.0 5.0 5.0]	SQP	No feasible Solution	N/A
[1.0 0.0 5.0 9.0]	SQP	[0 0.134 7.587 7.714]	0.424
[0.0 0.0 7.5 7.5]	SQP	[0.149 0.010 9.000 6.258]	0.539
[-1.0 0.0 9.0 5.0]	SQP	[-0.981 -0.777 8.815 5.319]	0.638
N/A	GA + SQP	[0.065 1.405 8.795 6.321]	0.705

5.2.2 3-DOF Reconfigurable Manipulators

In order to further ensure that the proposed optimization strategy generates global values that minimize the objective function, this methodology has to be benchmarked against another technique.

We choose to employ a method that finds the optimal parameters by simply varying the design parameter values and finding the respective robot performances. A 3-DOF reconfigurable robotic arm was chosen for our comparison, because in a 3-DOF case, the EE's orientation isn't of any importance. This makes α_2 independent of the objective function since it is directly attached to the EE, *only* affecting its orientation. Consequently, the design parameters become α_1, l_1 and l_2. A 4D figure can then be plotted, with 3 dimensions being α_1, l_1 and l_2, and the 4th dimension (performance or $1 \,/\, \mathcal{W}_{\mathrm{PS}}$) represented as color. The optimal design parameter combination can finally be identified and compared with the results of the proposed optimization strategy.

Method A: Finding the optimal parameters using the presented optimization strategy.

By following the steps provided in the flowchart of Figure 4.1, the desired workspace is shown in Figure 5.1, and the design parameters are bounded such that

$$\begin{aligned} -90^{\mathrm{o}} \leq \alpha_1 \leq 90^{\mathrm{o}} \\ 5\mathrm{m} \leq l_1 \leq 9\mathrm{m} \\ 5\mathrm{m} \leq l_2 \leq 9\mathrm{m}\,. \end{aligned}$$

Optimized design parameters at the worst case $\mathcal{W}_{\mathrm{PS}} = 1.735$, are found to be:

$$\begin{aligned} \alpha_1 &= -0.01404 \text{ rad} \\ l_1 &= 8.4476 \text{ m} \\ l_2 &= 6.9851 \text{ m} \end{aligned}$$

Post-optimality investigation has also been conducted and results are illustrated in Figure 5.2 and Figure 5.3, which confirm the global nature of the evaluated optimal solution.

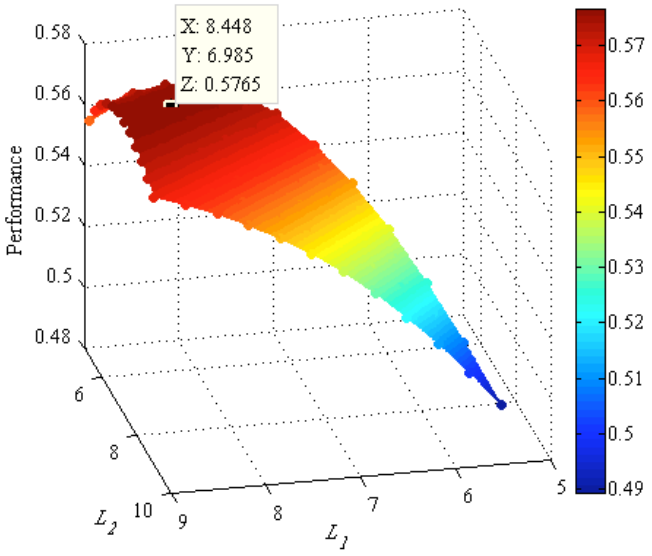


Figure 5.2 Post-optimization analysis by varying l_1 and l_2 at a constant α_1 shows that the optimized link lengths are indeed at the highest performance of the objective function. Note the performance is denoted by the inverse of $\boldsymbol{\mathcal{W}_{PS}}$.

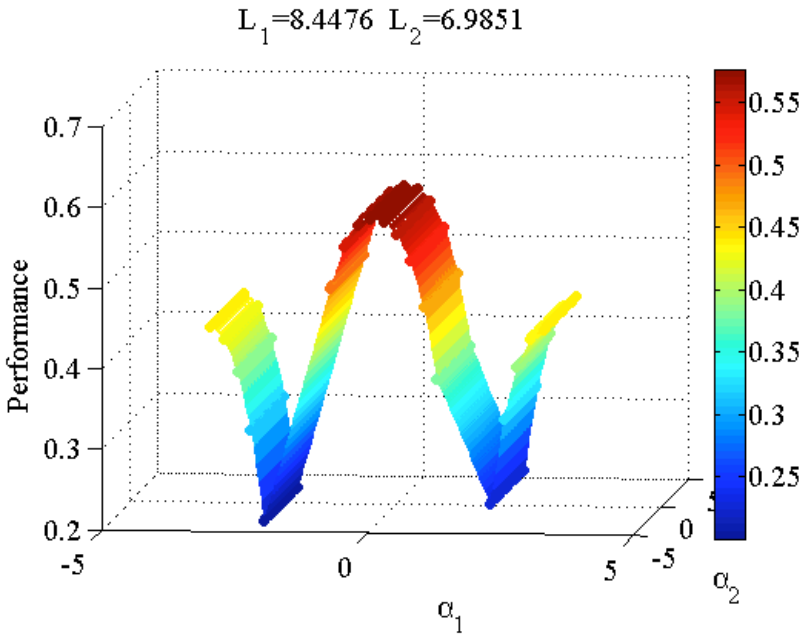


Figure 5.3 Post-optimization analysis by varying α_1 and α_2 with constants link lengths. It shows that the optimized link twist (α_1=-0.01404 rad) is at the highest performance of the objective function. Notice that varying α_2 does not affect the performance. The performance is denoted by the inverse of $\boldsymbol{\mathcal{W}_{PS}}$.

Method B: Finding the optimal solution using parametric study by varying the design parameter values.

The same desired workspace shown in Figure 5.1 is defined. In this method, the design parameters are varied between their lower and upper bounds. So a matrix comprising of all different values of design parameter vector $\mathbf{x} = [\alpha_1, l_1, l_2]$ is generated, where the lower and upper bounds of the design parameters are the same as in Method A, and the worst case $\mathcal{W}_{PS}$ in the workspace is determined for each $\mathbf{x}$. Then, the configuration resulting in the *best* (smallest) $\mathcal{W}_{PS}$ of all worst-case performances is chosen to be the global solution. Figure 5.4 shows the global solution (black dot), and it can be seen that the optimal design parameters are exactly the same as the ones found in Method A.

Both methods generate the same solution, hence validating the proposed optimization strategy of Method A.

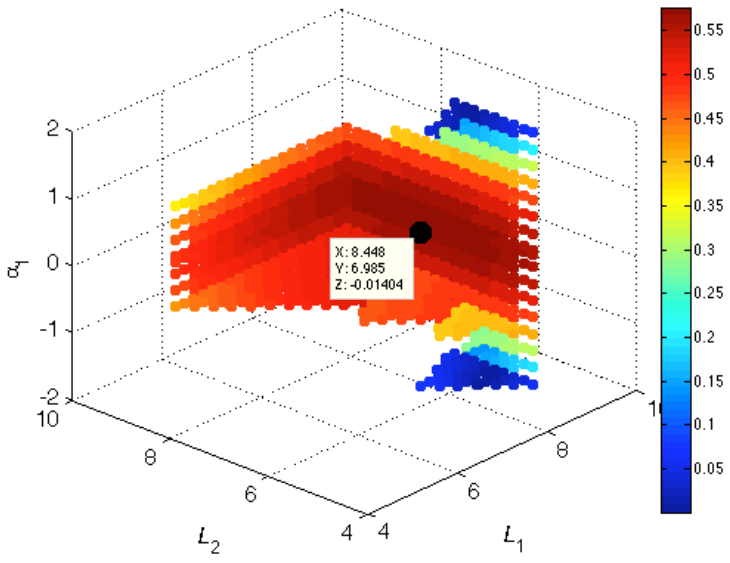


Figure 5.4 Different values of l_1, l_2, α_1, as a function of performance (color). It is seen that the best design parameters (black dot) lie in the high-performance region of the figure.

5.3 3-DOF Reconfigurable Manipulator - Minimize $\mathcal{W}_{PS}$

The first step in the optimization strategy is specifying a set of points describing a workspace, as indicated in Figure 5.5a. The reconfigurable robot is to be optimized for link lengths and link twists, and its base is located at the origin. The first 3 rows of Table 2.1 illustrate the DH parameters. For each point in the workspace, the IK solution (vector $\boldsymbol{\theta_{IK}}$) is explicitly computed in terms of the DH parameters and substituted in the Jacobian matrix. It is noted that the base joint $\boldsymbol{\theta_1}$ does not affect the robot's performance, which is reasonable since the first axis does not affect singularities, neither does it affect isotropy (Angeles, 2007). Also as mentioned earlier, since the last link is directly connected to the EE without any joint offset, α_2 cannot have any control over the robot's performance since it only changes the EE orientation. The objective function can be formulated as follows:

$$\begin{aligned} &\min\{\mathcal{W}_{\mathrm{PS}}\,(\alpha_1, \alpha_2, l_1, l_2)\} \\ &\text{subject to: } \begin{cases} isreal(\boldsymbol{\theta_{IK}}) = 1 \\ \boldsymbol{\theta_{min}} \le \boldsymbol{\theta_{IK}} \le \boldsymbol{\theta_{max}} \\ \dot{\boldsymbol{\theta}}_{min} \le \dot{\boldsymbol{\theta}} \le \dot{\boldsymbol{\theta}}_{max} \\ \boldsymbol{\tau_{min}} \le \boldsymbol{\tau} \le \boldsymbol{\tau_{max}} \\ \boldsymbol{lb} \le \{\alpha_1, \alpha_2, l_1, l_2\} \le \boldsymbol{ub} \end{cases} \end{aligned}$$

Notice $\boldsymbol{\theta}$ is excluded from the objective function since it is now a function of the design parameters α_1, l_1, and l_2. The $isreal(\boldsymbol{\theta_{IK}}) = 1$ condition is necessary to guarantee a real IK solution, and $\boldsymbol{\theta_{IK}}$ is clearly bounded between the lower and upper limits ($\boldsymbol{\theta_{min}}$ and $\boldsymbol{\theta_{max}}$ respectively). The joint rates and joint torques should be bounded to prevent motor failure, in case an EE velocity or a static force is applied. Finally, the set $\{\alpha_1, \alpha_2, l_1, l_2\}$ is bounded by the lower and upper bounds, $\boldsymbol{lb}$ and $\boldsymbol{ub}$ vectors, such that:

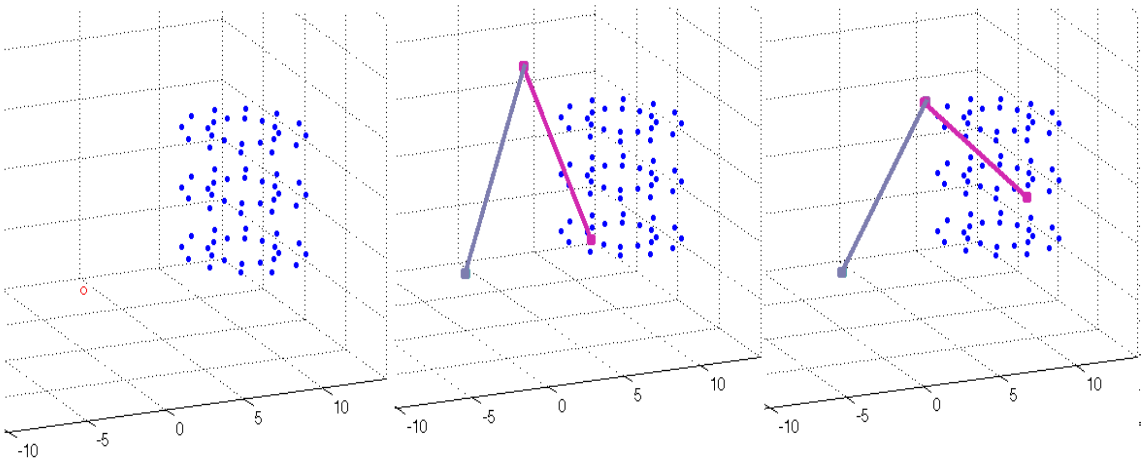
$$-90^{\mathrm{o}} \le \alpha_1 \le 90^{\mathrm{o}}$$

$$-90^{\circ} \leq \alpha_2 \leq 90^{\circ} .$$
$$5\text{m} \leq l_1 \leq 9\text{m}$$
$$5\text{m} \leq l_2 \leq 9\text{m}$$

The successive GA and SQP optimizations converge at a global minimum of $\mathcal{W}_{\text{PS}}$=1.7341, meaning that $\mathcal{W}_{\text{PS}}$ cannot be any worse (higher) than 1.7341 in the workspace, and this value would be even worse if the optimized design vector **x** was varied. Figures 5.5b-c depict the optimized robot in the workspace, and Figure 5.6 shows the performance curve (blue) sorted in ascending order, with the corresponding normalized manipulability and inverse condition number. Recall that the performance in the figures is represented by the inverse of $\mathcal{W}_{\text{PS}}$. So, the optimized worst-case performance is about 58%. The mean is 75.1%, and the standard deviation is calculated to be 6.3%, indicating that the average performance is high, and its distribution is close to the mean. The optimized DH parameters are:

α_1 = -0.0140 rad l_1 = 8.4476 m
α_2 = 0.7576 rad l_2 = 6.9851 m .

Figure 5.7 shows the optimized robot with its corresponding performance in the workspace, and it is apparent that the high-performance (red) region now contains the workspace of interest. On the other hand, Figure 5.8 shows the IK solution distribution in joint space, verifying that the joint angle occurrences all lie in the high performance regions. Furthermore, post-optimization results shown in Figure 5.9 and Figure 5.10 confirm the existence of a global minimum (or maximum for demonstration purposes). Notice that as expected, varying the link twist α_2 in Figure 5.10 has no effect on the performance.



(a) Workspace (b) Optimized robot (c) Optimized robot

Figure 5.5 (a) Set of points defining a cylindrical workspace. The circle is the robot's base. (b) and (c) depict the 3-DOF optimized robot in the workspace.

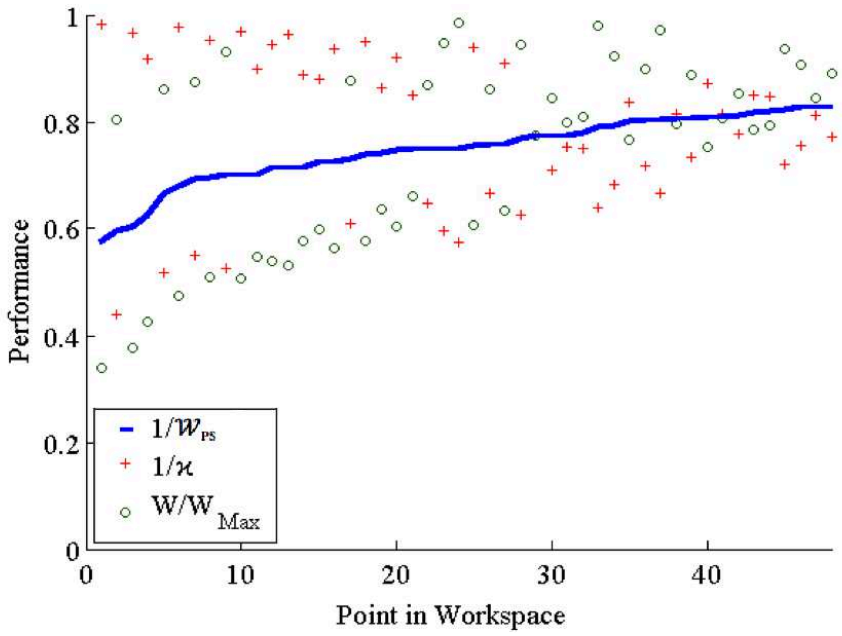


Figure 5.6 The performance curve (blue) of the 3-DOF robot sorted in ascending order with the corresponding normalized manipulability and inverse condition number.

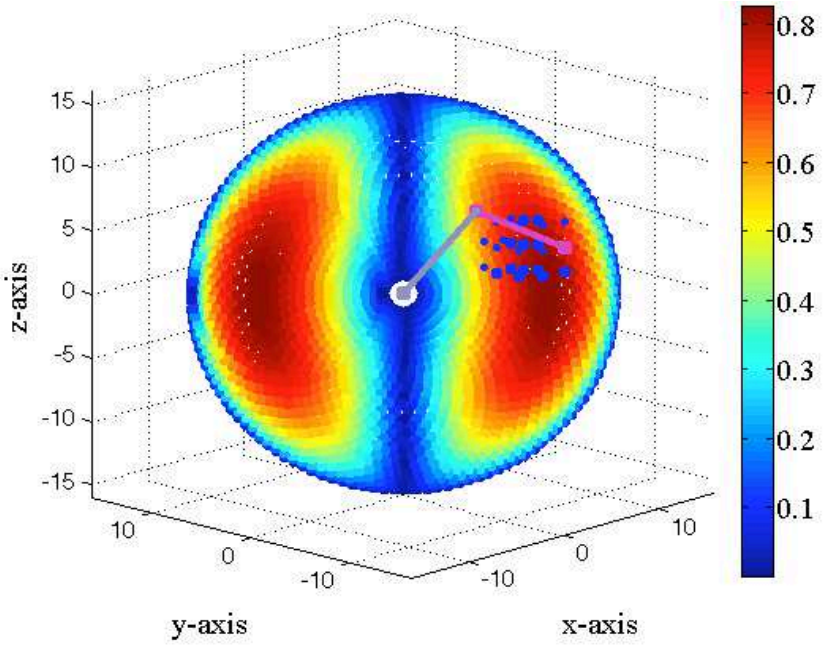


Figure 5.7 Performance of the 3-DOF robot in Cartesian space. The visible dots indicate boundaries of the workspace.

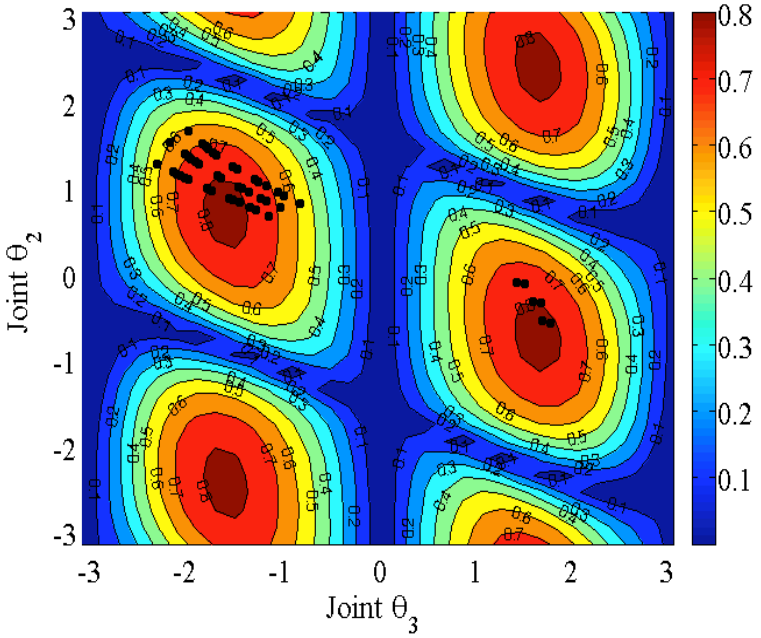


Figure 5.8 The IK distribution of the optimized 3-DOF robot in joint space. The black dots, indicating the IK solutions of θ_2 and θ_3, are located in the high performance regions.

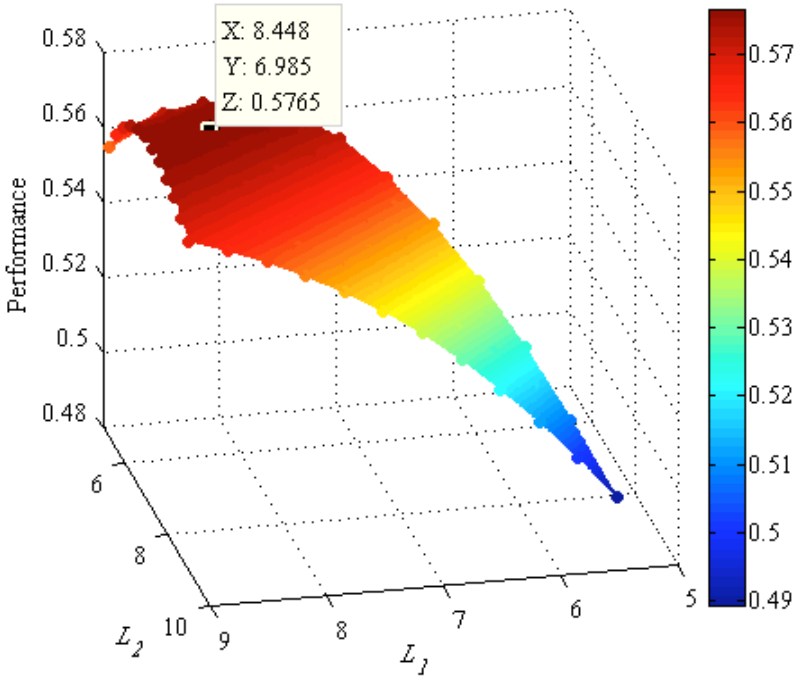


Figure 5.9 Post-optimization analysis by varying l_1 and l_2 at a constant α_1 shows that the optimized link lengths are at the highest performance of the objective function.

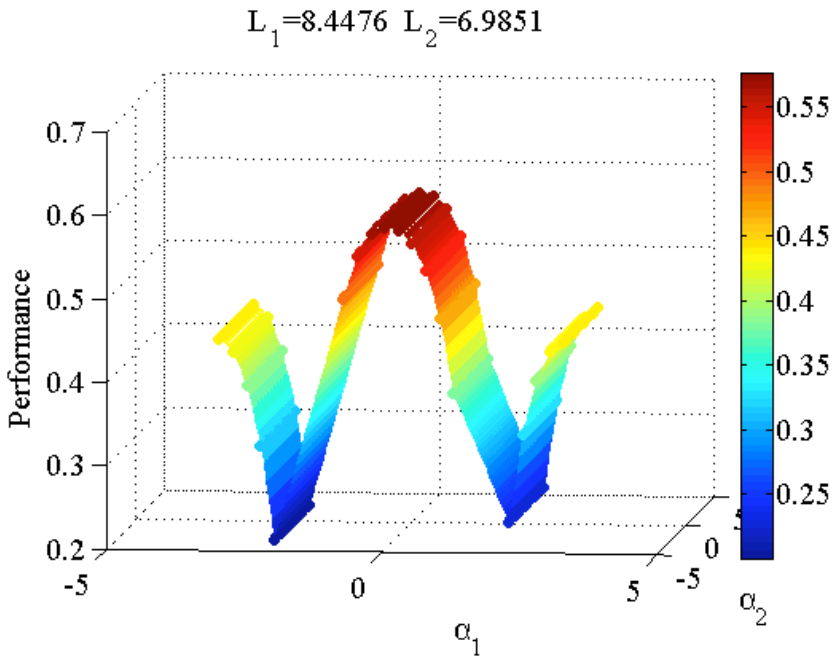


Figure 5.10 Post-optimization analysis by varying α_1 and α_2 with constants link lengths, shows that the optimized link twists are at the highest performance.

5.4 6-DOF Reconfigurable Manipulator - Minimize $\mathcal{W}_{PS}$

The minimization of the $\mathcal{W}_{PS}$ index (MPIPS) is expected to maximize the EE Cartesian velocity making the manipulability ellipsoid as spherical as possible, or minimize the joint rates, making their distribution as uniform as possible. Again, we first need to specify a set of points describing a workspace. For this example, the chosen workspace is described by a set of points and orientations. The positional workspace is a cube as shown in Figure 5.11, and the point orientations possess rotation matrices such that the EE is always directed perpendicular to the surface of the cube and pointing inwards. The robot is to be optimized for link lengths and link twists, and its base is located at the origin. Table 2.1 illustrates the robot's DH parameters.

As for the IK solution (vector $\boldsymbol{\theta_{IK}}$), it is computed symbolically in a closed form using Pieper's method, as was described previously and will be done in later examples.

The objective function can now be formulated as follows, using the $\mathcal{W}_{PS}$ index:

$$\min\{\mathcal{W}_{PS}(\alpha_1, \alpha_2, l_1, l_2)\}$$
$$\text{subject to:} \begin{cases} isreal(\boldsymbol{\theta_{IK}}) = 1 \\ \boldsymbol{\theta_{min}} \le \boldsymbol{\theta_{IK}} \le \boldsymbol{\theta_{max}} \\ -1 \text{ rad/s} \le \dot{\boldsymbol{\theta}} \le 1 \text{ rad/s} \\ -15 \text{ N.m} \le \boldsymbol{\tau} \le 15 \text{ N.m} \\ \boldsymbol{lb} \le \{\alpha_1, \alpha_2, l_1, l_2\} \le \boldsymbol{ub} \end{cases}$$

The joint rates and joint torques are constrained to avoid any motor failure. Finally, the design parameters should be also bounded such that:

$$\begin{aligned} -90^{o} \le \alpha_1 \le 90^{o} \\ -90^{o} \le \alpha_2 \le 90^{o} \\ 5\text{m} \le l_1 \le 9\text{m} \\ 5\text{m} \le l_2 \le 9\text{m} \end{aligned}.$$

The optimization converges at a global minimum of $\mathcal{W}_{PS}$=1.4803 for worst-case performance, and Figure 5.12 illustrates the optimized robot in the workspace. Figure 5.13 shows the performance curve sorted in ascending order, with the corresponding normalized manipulability and inverse condition number. We can extract the following information from the figure: the optimized worst-case performance is about 68% with an RMS value of 72%, and a standard deviation of 3.8%. These values clearly indicate a consistently high manipulability performance for the optimized robot, inside the pre-specified workspace. The optimized DH parameters are:

α_1 = -0.3835 rad l_1 = 8.0557 m
α_2 = -1.0570 rad l_2 = 6.4206 m .

Comparably to the previous example, Figure 5.14 shows that the desired workspace is now in the high-performance (red) region. On the other hand, Figure 5.15 validates that the IK solution lies within its lower and upper boundaries, and Figure 5.16 shows the IK solution distribution in joint space of θ_2 and θ_3, verifying that the joint angle occurrences all lie in the high performance regions. Furthermore, post-optimization results shown in Figure 5.17 and Figure 5.18 confirm the existence of a global minimum (recall that as we portray performance by the inverse of $\mathcal{W}_{PS}$, the inverse of the minimum is a maximum).

Now, in order to show that the joint rate requirement is minimized, the RMS joint torque and joint rate are calculated from equations (2.29) and (2.31) for two different cases applied on the EE, as shown below:

Static force	$\begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix} N$	Eq. (2.29) →	$\tau_{RMS} = \begin{bmatrix}12.271\\8.3497\\8.8382\\0\\0\\0\end{bmatrix} N.m$
Velocity	$\begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix}$ m/s	Eq. (2.31) →	$\dot{\boldsymbol{\theta}}_{\mathrm{RMS}} = \begin{bmatrix}0.1064\\0.0502\\0.2079\\0.1745\\0.0620\\0.1801\end{bmatrix}$ rad/s

Note that the joint torques and rates are still between their lower and upper bounds, but the torque requirement is high. These values are to be retained for future comparison with the results of minimizing the $\mathcal{W}_{\mathrm{PS}-\mathcal{F}}$ index. Also, notice the uniformity of joint rate distribution, for the corresponding optimized design parameters on Figure 5.19.

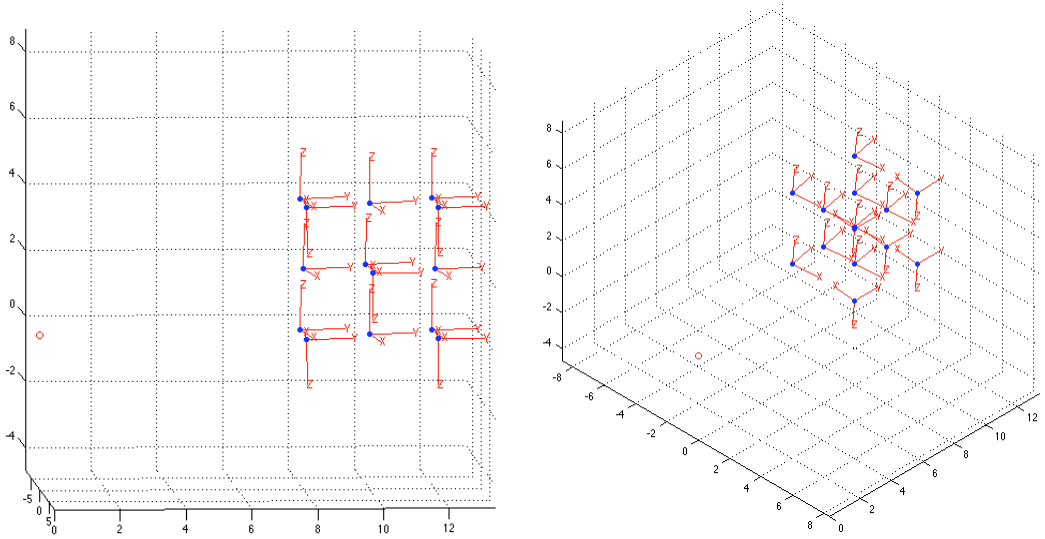


Figure 5.11 Set of points and orientations defining a cubic workspace. The red circle is the robot's base.

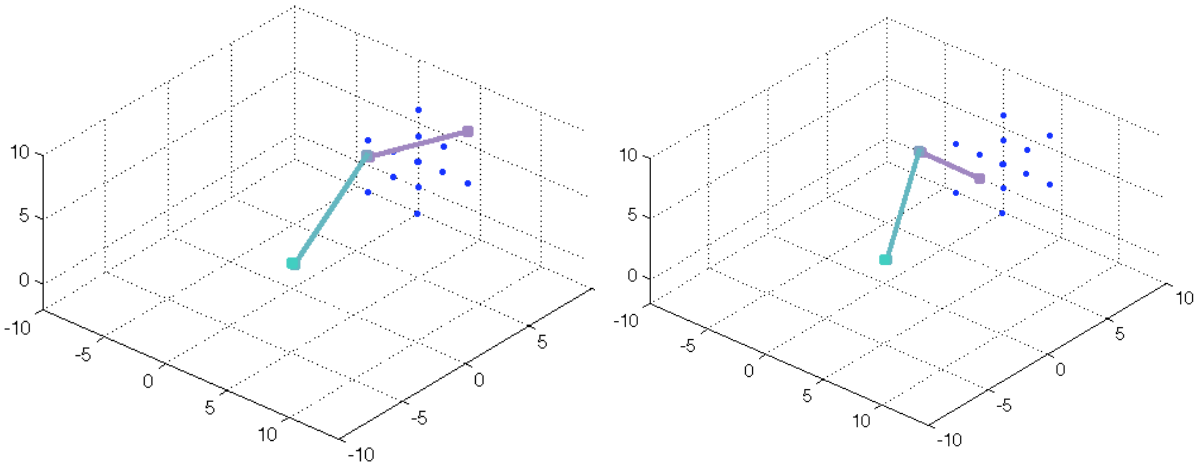


Figure 5.12 The 6-DOF optimized robot in Cartesian workspace.

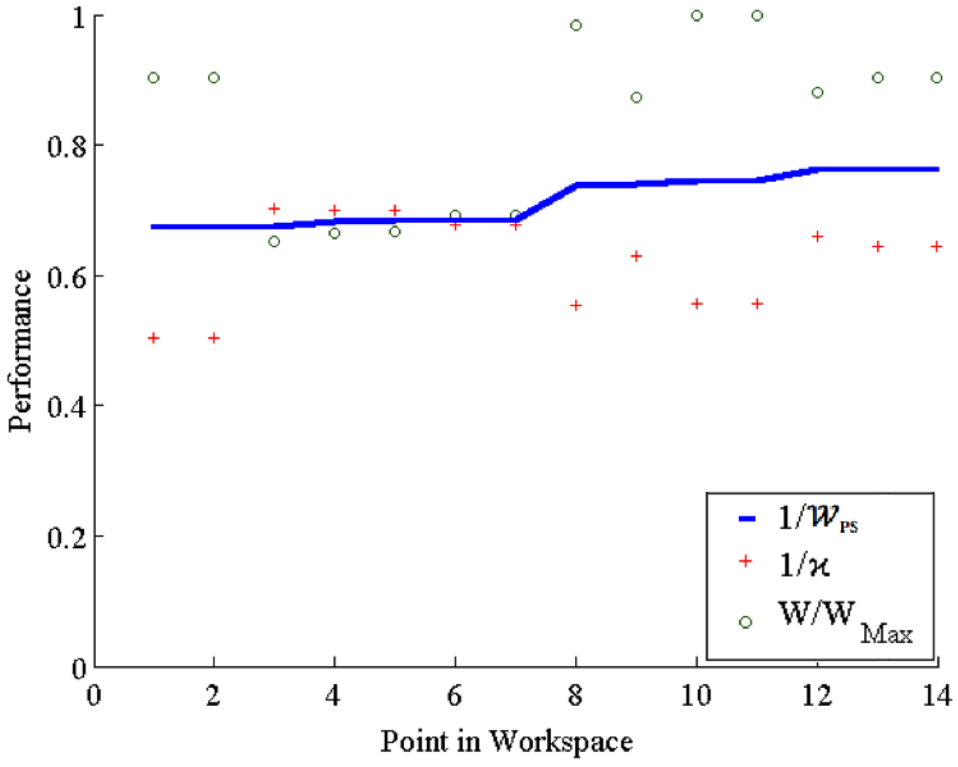


Figure 5.13 The performance curve of the 6-DOF robot sorted in ascending order with the corresponding normalized manipulability and inverse condition number.

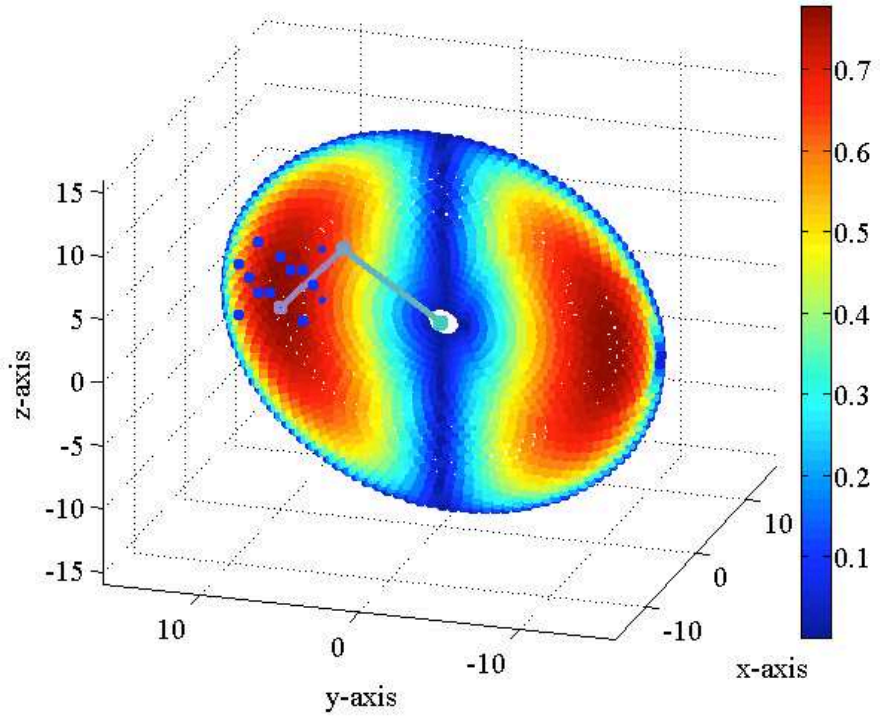


Figure 5.14 Performance of the 6-DOF robot in Cartesian space. The visible dots indicate boundaries of the workspace.

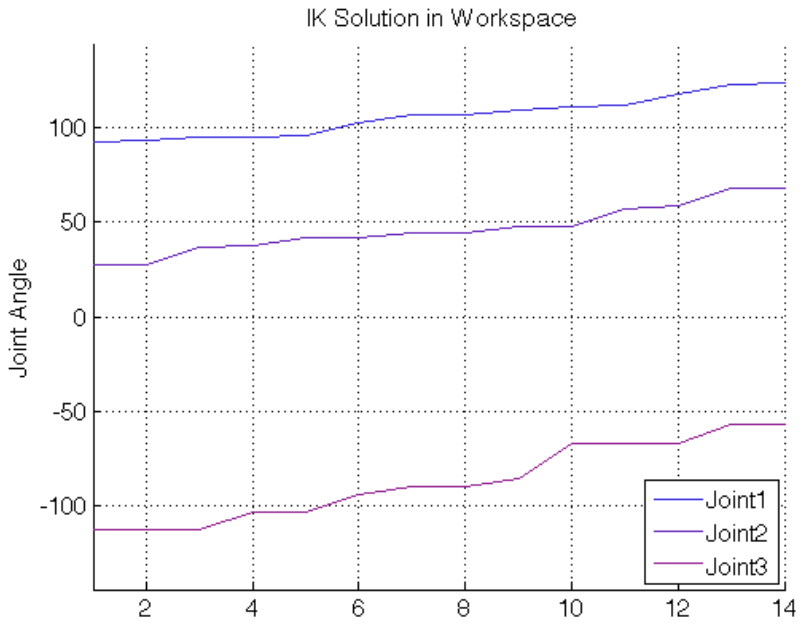


Figure 5.15 IK solution for the first 3 joints, indicating that the IK solution conformed to the joint angle constraints.

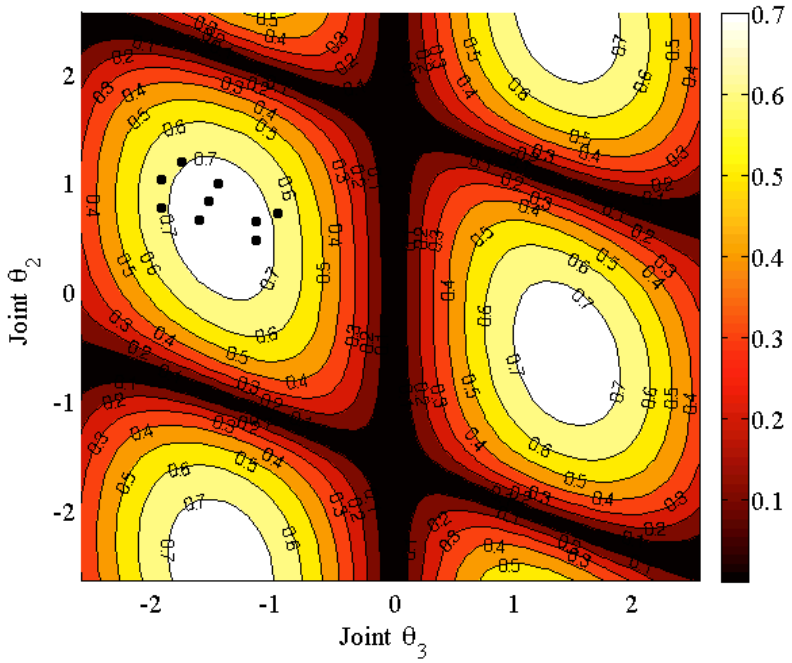


Figure 5.16 The IK distribution of the optimized 6-DOF robot in joint space. The black dots, indicating the IK solutions of θ_2 and θ_3, are located in the high performance regions. Note that the color map is changed here for clearer comparison with the subsequent case.

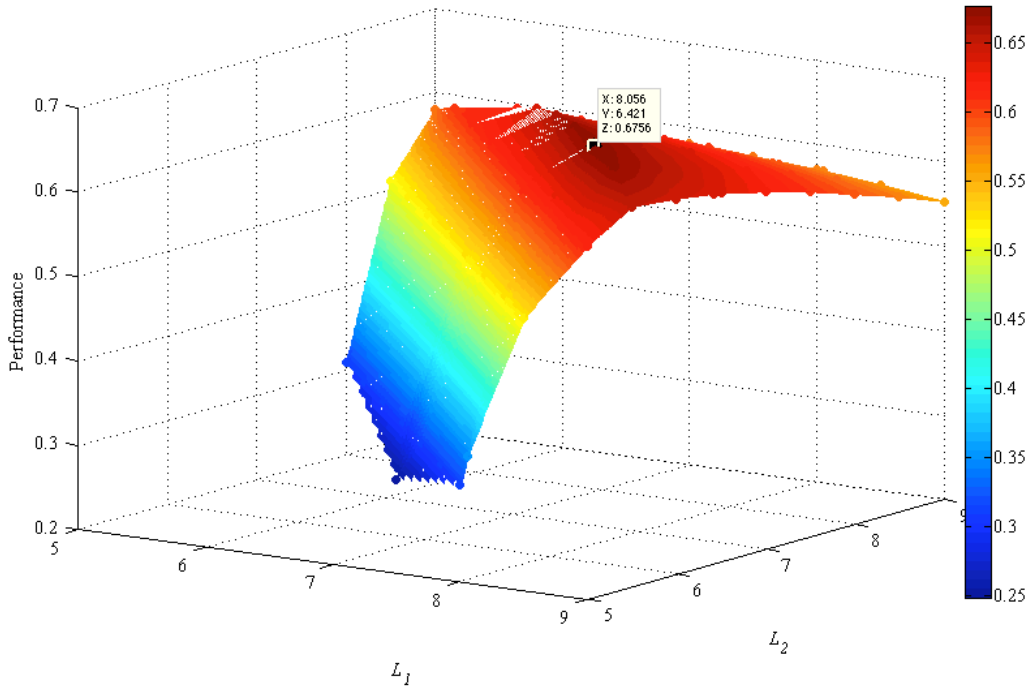


Figure 5.17 Post-optimization analysis by varying l_1 and l_2 at a constant α_1 shows that the optimized link lengths are at the highest performance of the objective function.

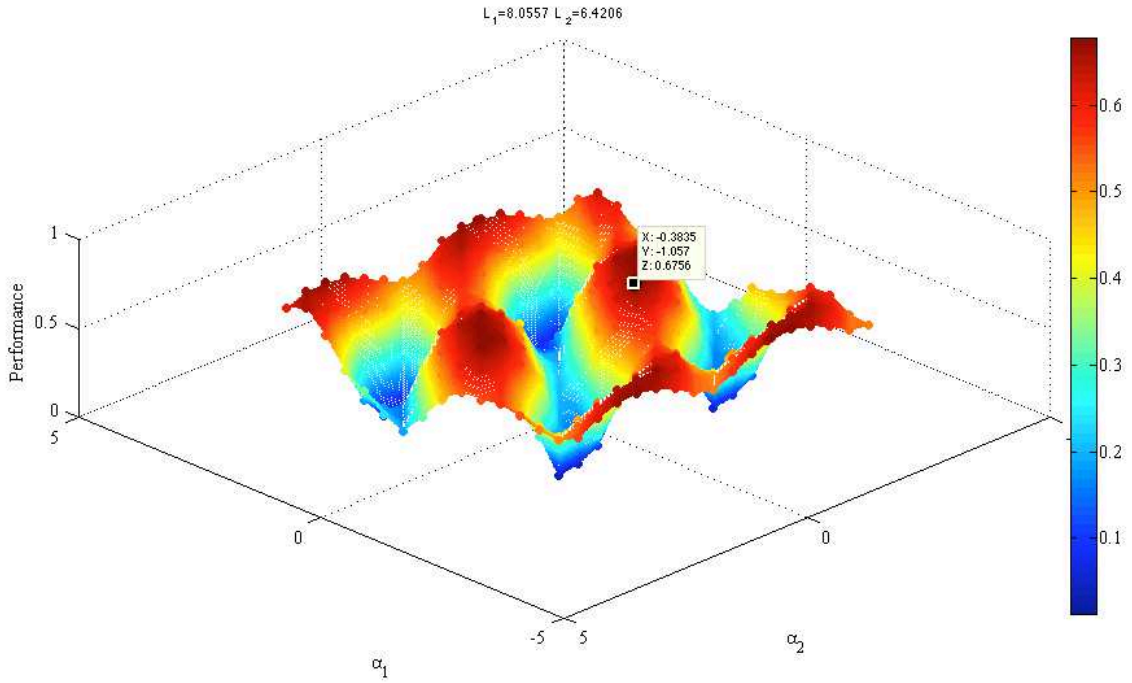


Figure 5.18 Post-optimization analysis by varying α_1 and α_2 with constants link lengths, shows that the optimized link twists are at the highest performance.

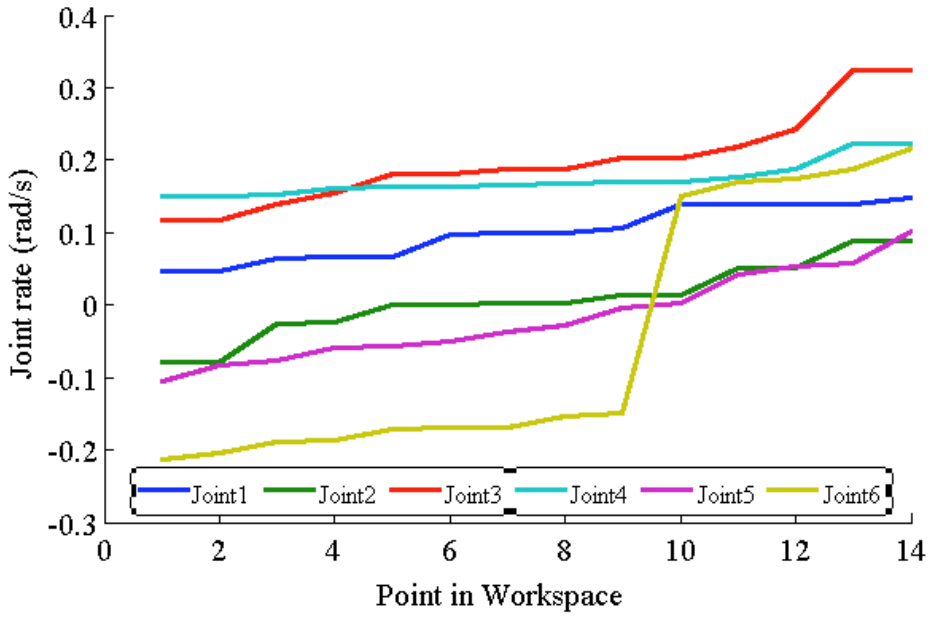


Figure 5.19 Joint rate distribution in the workspace, sorted in ascending order.

5.5 6-DOF Reconfigurable Manipulator – Minimize $\mathcal{W}_{\mathrm{PS}-\mathcal{F}}$

The $\mathcal{W}_{\mathrm{PS}-\mathcal{F}}$ index is expected to minimize the joint torques, making their distribution as uniform as possible. Similarly to the previous example, a set of points are specified to describe a workspace. For this example, the positional and orientational workspace chosen is the same as the previous case, shown in Figure 5.11. The robot is again to be optimized for link lengths and link twists, and its base is located at the origin. Clearly, the DH table does not change, and is represented in Table 2.1.

For each point in the workspace, the IK solution is explicitly computed in terms of the DH parameters and substituted in the Jacobian matrix. The objective function can now be formulated using the $\mathcal{W}_{\mathrm{PS}-\mathcal{F}}$ index subject to the same equality and inequality constraints as the previous example:

$$\begin{aligned} &\min\{\, \mathcal{W}_{\mathrm{PS}-\mathcal{F}}(\alpha_1, \alpha_2, l_1, l_2)\} \\ &\text{subject to: } \begin{cases} isreal(\boldsymbol{\theta_{IK}}) = 1 \\ \boldsymbol{\theta_{min}} \le \boldsymbol{\theta_{IK}} \le \boldsymbol{\theta_{max}} \\ -1 \text{ rad/s} \le \dot{\boldsymbol{\theta}} \le 1 \text{ rad/s} \\ -15 \text{ N.m} \le \boldsymbol{\tau} \le 15 \text{ N.m} \\ \boldsymbol{lb} \le \{\alpha_1, \alpha_2, l_1, l_2\} \le \boldsymbol{ub} \end{cases} \end{aligned}$$

and the set $\{\alpha_1, \alpha_2, l_1, l_2\}$ is bounded such that:

$$\begin{aligned} -90^{\mathrm{o}} \le \alpha_1 \le 90^{\mathrm{o}} \\ -90^{\mathrm{o}} \le \alpha_2 \le 90^{\mathrm{o}} \\ 5\mathrm{m} \le l_1 \le 9\mathrm{m} \\ 5\mathrm{m} \le l_2 \le 9\mathrm{m} \end{aligned} .$$

The combined GA and SQP optimization algorithm converges at a global minimum of $\mathcal{W}_{\mathrm{PS}-\mathcal{F}}$=-1.2384. This value means that the worst $|\mathcal{W}_{\mathrm{PS}-\mathcal{F}}|$ in the pre-specified workspace is 1.2384. As a side note, contrary to minimizing $\mathcal{W}_{\mathrm{PS}}$ to 1, $|\mathcal{W}_{\mathrm{PS}-\mathcal{F}}|$ is to be maximized. Our obtained value is the best value the optimized robot can achieve

at the worst point of the workspace. Changing the entries of the optimized design vector would generate a smaller $|\mathcal{W}_{\mathrm{PS}-\mathcal{F}}|$, hence a lower force performance. Figure 5.20 depicts the optimized robot in the workspace. Note that Figure 5.21 shows the performance curve sorted in ascending order, with the corresponding normalized manipulability and inverse condition number. The RMS performance is 1.55, and the standard deviation is calculated to be 0.27. The optimized DH parameters are found to be:

α_1 = 1.6399 rad $\quad$ l_1 = 9.0000 m
α_2 = -1.2463 rad $\quad$ l_2 = 9.0000 m .

Figure 5.22 shows the optimized robot with its corresponding performance in the workspace, and it is apparent that the high-performance (red) region now contains the workspace of interest. The IK solution is also confirmed to lie within the given joint angle bounds as shown in Figure 5.23, and Figure 5.24 shows its distribution in joint space, verifying that the joint angle occurrences all lie in the high performance regions. Furthermore, post-optimization results shown in Figure 5.25 and Figure 5.26 confirm the existence of a global minimum (or maximum in the case of $|\mathcal{W}_{\mathrm{PS}-\mathcal{F}}|$).

Now, in order to show that the joint torque requirement is minimized, the RMS joint torque and joint rate are calculated from equations (2.29) and (2.31) for two different cases applied on the EE, as shown below:

Static force: $\begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix} N$ $\xrightarrow{\text{Eq. (2.29)}}$ $\tau_{RMS} = \begin{bmatrix}2.9782\\4.4554\\12.680\\0\\0\\0\end{bmatrix} N.m$

Velocity: $\begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix}$ m/s $\xrightarrow{\text{Eq. (2.31)}}$ $\dot{\boldsymbol{\theta}}_{\text{RMS}} = \begin{bmatrix}0.0839\\0.1374\\0.2164\\0.5517\\0.1244\\0.4713\end{bmatrix}$ rad/s

Note that the joint torques and rates are still between their lower and upper bounds. Comparing these two values with the previous case, it can be seen that the RMS value of the joint torques here is lower than when using $\mathcal{W}_{\text{PS}}$. Conversely, the RMS value of the joint rates here is higher than when using $\mathcal{W}_{\text{PS}}$. The difference between the two indices $\mathcal{W}_{\text{PS}}$ and $\mathcal{W}_{\text{PS}-\mathcal{F}}$ is thus now clear. Also, notice the uniformity of joint torque distribution for the corresponding optimized design parameters on Figure 5.27.

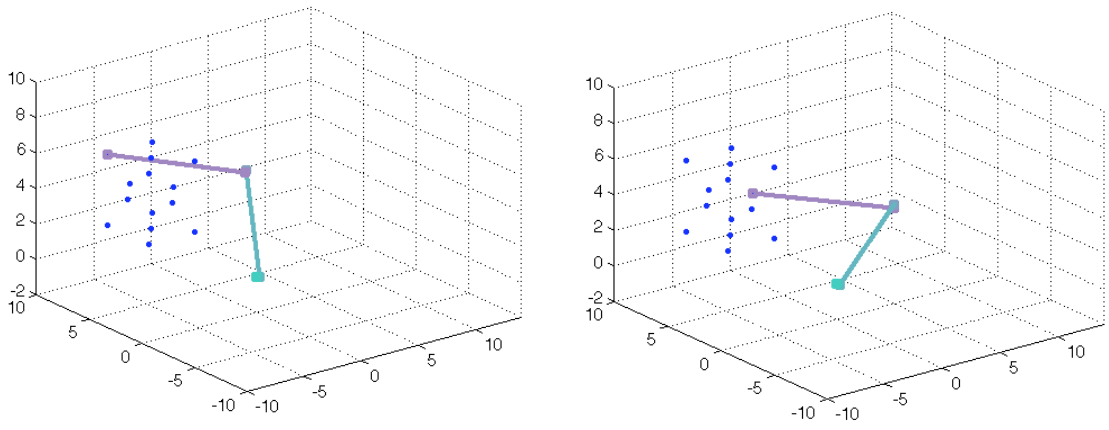


Figure 5.20 The 6-DOF optimized robot in Cartesian workspace.

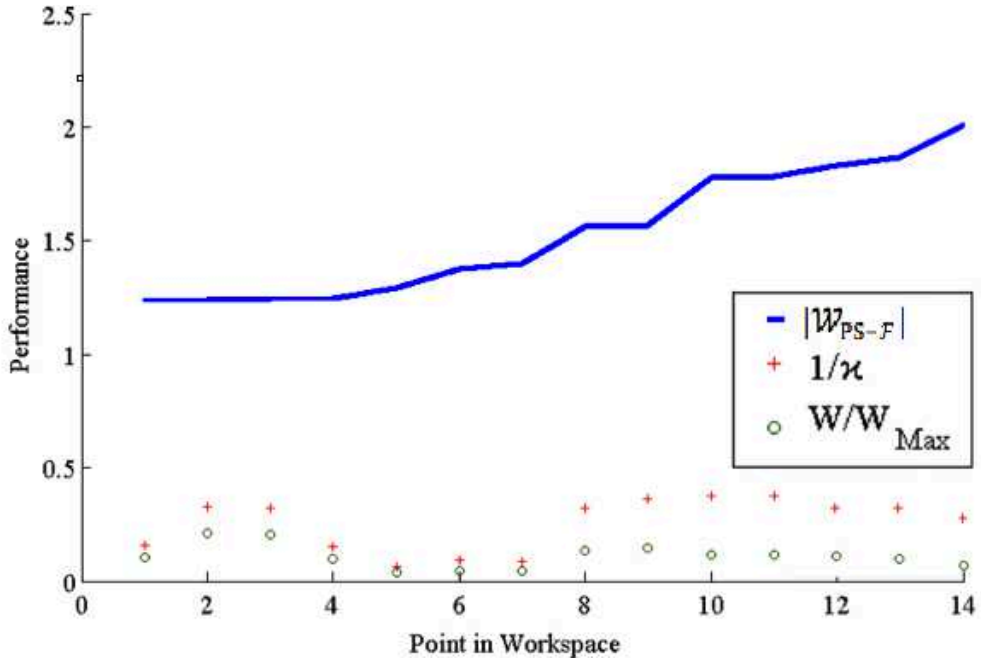


Figure 5.21 The performance curve of the 6-DOF robot sorted in ascending order with the corresponding normalized manipulability and inverse condition number.

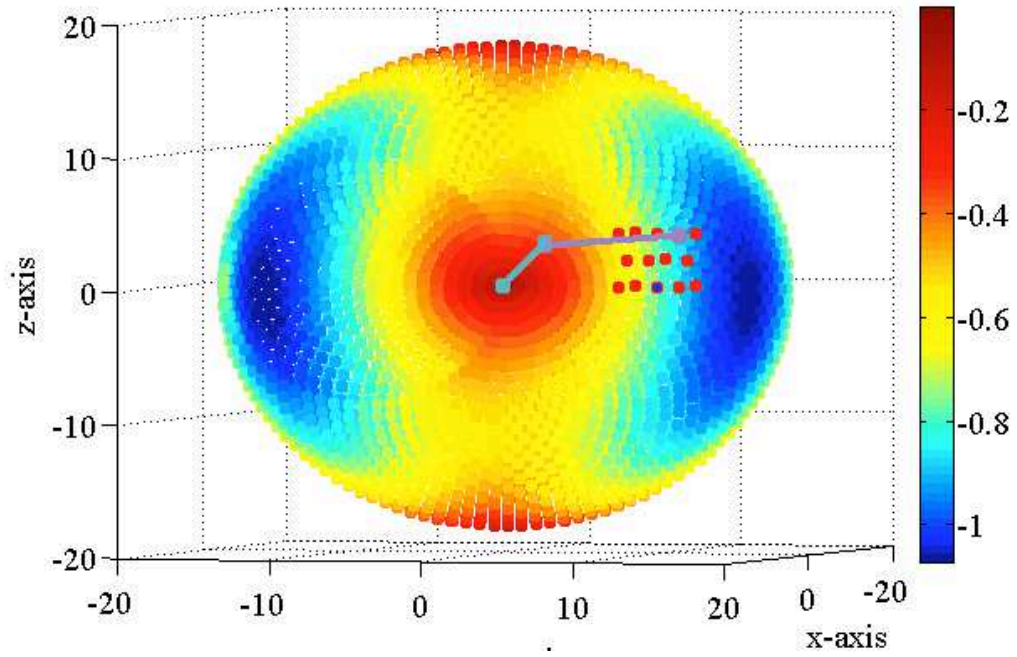


Figure 5.22 Performance of the 6-DOF robot in Cartesian space. The visible red dots indicate boundaries of the cube. Note that the scale has been multiplied by its negative inverse for better illustration, making the red region equivalent to high performance, and blue region low performance.

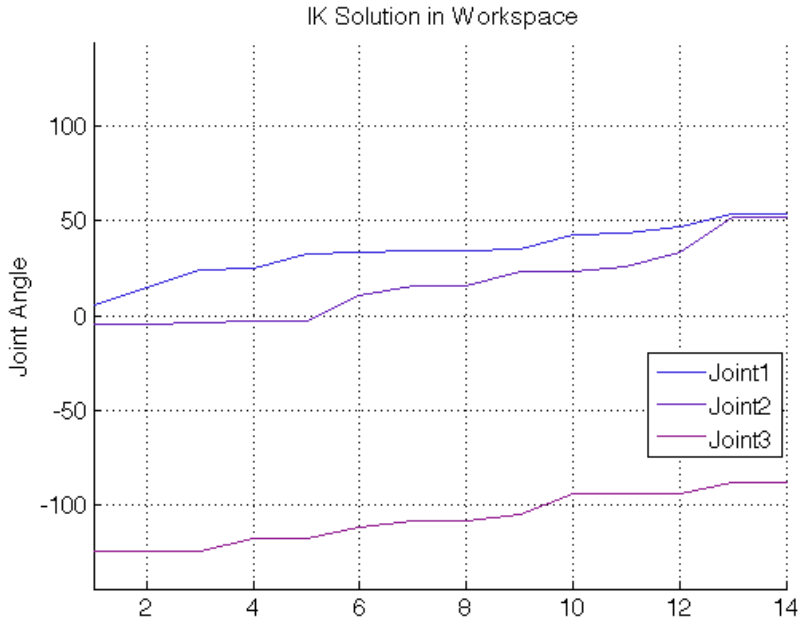


Figure 5.23 IK solution for the first 3 joints, indicating that the IK solution conformed to the joint angle constraints.

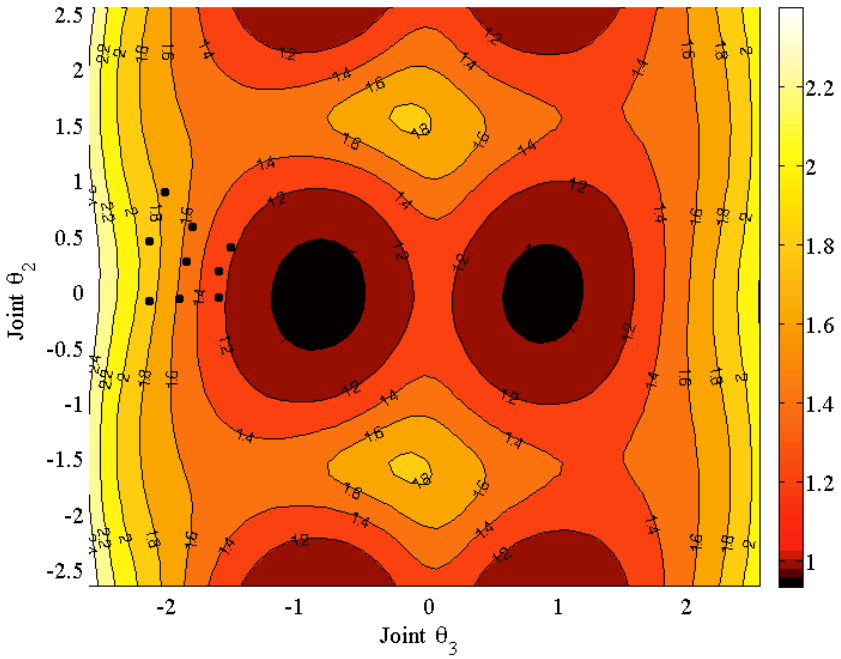


Figure 5.24 The IK distribution of the optimized 6-DOF robot in joint space. The black dots, indicating the IK solutions of θ_2 and θ_3, are located in the high performance regions.

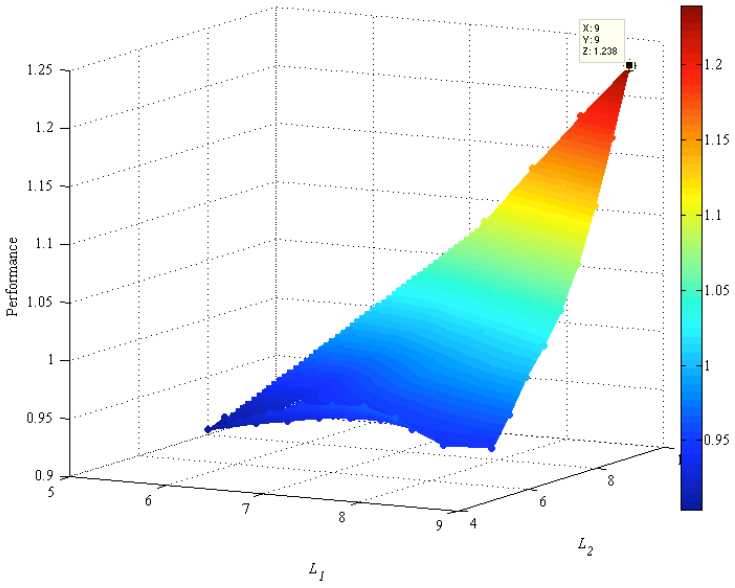


Figure 5.25 Post-optimization analysis by varying l_1 and l_2 at a constant α_1 shows that the optimized link lengths are at the highest performance of the objective function.

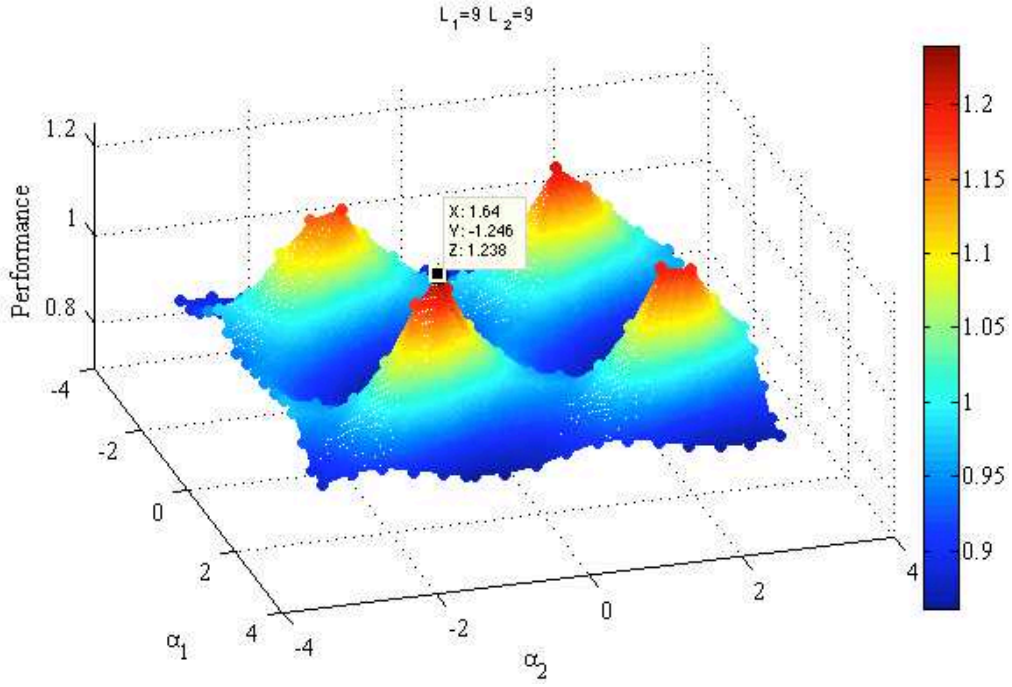


Figure 5.26 Post-optimization analysis by varying α_1 and α_2 with constants link lengths, shows that the optimized link twists are at the highest performance.

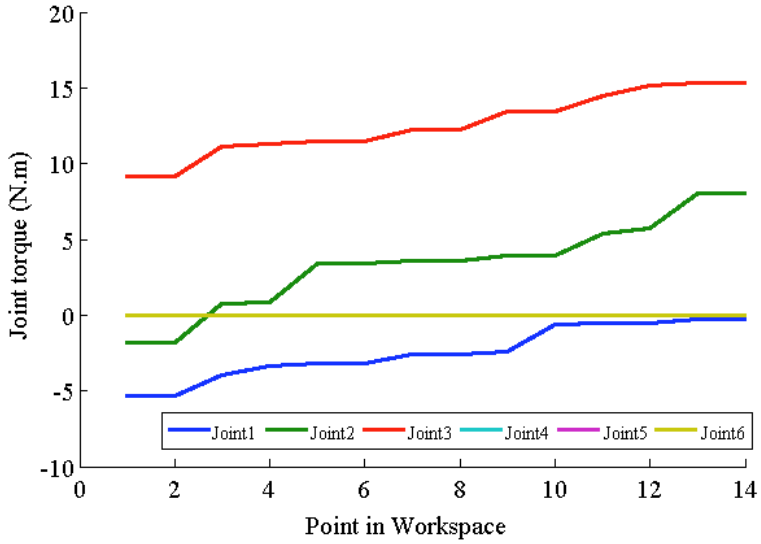


Figure 5.27 Joint torque distribution in the workspace, sorted in ascending order.

5.6 6-DOF Reconfigurable Manipulator – Minimize W/W_{max}

In this case, the force manipulability ellipsoid is to be maximized by minimizing the Yoshikawa index only. This will minimize the robot's velocity manipulability, but will maximize its force manipulability. After specifying a set of points and orientations in the desired workspace as shown in Figure 5.11 and described previously, the robot is to be optimized for link lengths and link twists, with its base located at the origin.

The same procedure shown in Table 4.1 is followed, with the exception of the objective function, which is now formulated as follows, using the W/W_{max} index:

$$\min\{ W/W_{max}\,(\alpha_1, \alpha_2, l_1, l_2)\}$$

$$\text{subject to: } \begin{cases} isreal(\boldsymbol{\theta_{IK}}) = 1 \\ \boldsymbol{\theta_{min}} \le \boldsymbol{\theta_{IK}} \le \boldsymbol{\theta_{max}} \\ -1 \text{ rad/s} \le \dot{\boldsymbol{\theta}} \le 1 \text{ rad/s} \\ -15 \text{ N.m} \le \boldsymbol{\tau} \le 15 \text{ N.m} \\ \boldsymbol{lb} \le \{l_1, l_2\} \le \boldsymbol{ub} \end{cases}$$

and the set $\{l_1, l_2\}$ is bounded by the lower and upper bounds, ***lb*** and ***ub*** vectors, such

that:

$$5\text{m} \leq l_1 \leq 9\text{m}$$
$$5\text{m} \leq l_2 \leq 9\text{m}$$

The optimization algorithm converges at a global minimum of W/W_{max}= 0.2319 for worst-case performance. We are interested in determining the force manipulability ellipsoid of the robot, as well as the velocity manipulability ellipsoid. Afterwards we desire to compare it with an opposing case that maximizes the velocity only.

The optimized worst-case performance here is 23.19%. This is the highest manipulability in the desired workspace, i.e. the worst force manipulability. So the volumes of the normalized velocity and force ellipsoid are both illustrated in Figure 5.28. The following information can be extracted from the figure:

RMS volume of velocity manipulability = 0.18131
RMS volume of force manipulability = 7.4298
Worst volume of velocity manipulability = 0.077794
Worst volume of force manipulability = 4.3216

The optimized DH parameters are found to be:

α_1 = 1.6559 rad l_1 = 5.9740 m
α_2 = -1.2895 rad l_2 = 6.9790 m .

Now, the normalized manipulability index (W/W_{max}) is expected to minimize the joint torques maximizing the force manipulability ellipsoid volume. In order to show that the joint torque requirement is minimized, the average joint torque and joint rate are calculated from equations (2.29) and (2.31) for two different cases applied on the EE, as shown below:

Static force	$\begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix} N$	Eq. (2.29) →	$\tau_{RMS} = \begin{bmatrix}5.3315\\6.5493\\6.3298\\0\\0\\0\end{bmatrix} N.m$
Velocity	$\begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix}$ m/s	Eq. (2.31) →	$\dot{\boldsymbol{\theta}}_{\mathrm{RMS}} = \begin{bmatrix}0.4163\\0.1246\\0.6777\\0.5492\\0.1707\\0.4718\end{bmatrix}$ rad/s

It is noted that the joint torques and rates are still between their lower and upper bounds. These values are to be retained for future comparison with the results of maximizing the $\mathrm{W}/\mathrm{W}_{max}$ index (or minimizing $\mathrm{W}/\mathrm{W}_{max}$).

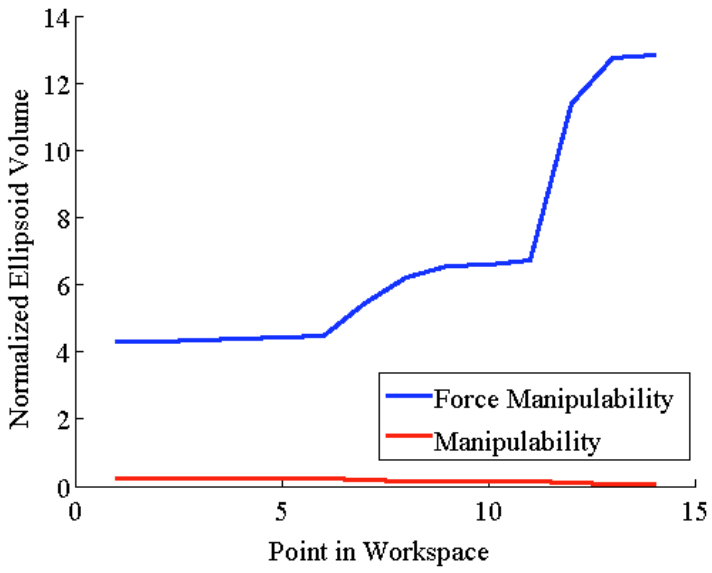


Figure 5.28 Volumes of the normalized velocity and force ellipsoid.

5.7 6-DOF Reconfigurable Manipulator – Maximize W/W_{max}

In this case, the velocity (manipulability) is to be maximized by maximizing the Yoshikawa index only. This will minimize its force manipulability. After specifying a set of points and orientations in the desired workspace as described previously, the robot is to be optimized for link lengths and link twists, with its base located at the origin. The exact procedure as the latter case is followed, with only the difference in the objective function, which can now be formulated as follows, using the W/W_{max} index:

$$\min\{-W/W_{max}\,(\alpha_1, \alpha_2, l_1, l_2)\}$$
$$\text{subject to:} \begin{cases} isreal(\boldsymbol{\theta_{IK}}) = 1 \\ \boldsymbol{\theta_{min}} \le \boldsymbol{\theta_{IK}} \le \boldsymbol{\theta_{max}} \\ -1 \text{ rad/s} \le \dot{\boldsymbol{\theta}} \le 1 \text{ rad/s} \\ -15 \text{ N.m} \le \boldsymbol{\tau} \le 15 \text{ N.m} \\ \boldsymbol{lb} \le \{l_1, l_2\} \le \boldsymbol{ub} \end{cases}$$

and the set $\{l_1, l_2\}$ is bounded by the same values as the latter case.

The successive GA and SQP optimizations converge at a global minimum of W/W_{max}= 0.7299 for worst-case performance. Again, we are interested in determining both the velocity and force manipulability ellipsoids. Afterwards we desire to compare them with the previous case that maximized the force only.

The optimized worst-case performance here is about 73%. This is the worst manipulability in the desired workspace. So the volumes of the normalized velocity and force ellipsoid are both illustrated in Figure 5.29. The following information can be extracted form the figure:

RMS volume of velocity manipulability = 0.8365
RMS volume of force manipulability = 1.2223
Worst volume of velocity manipulability = 0.73076
Worst volume of force manipulability = 1.0536

The optimized DH parameters are:

$\alpha_1 = 2.9139$ rad $\quad l_1 = 6.9744$ m
$\alpha_2 = 2.2645$ rad $\quad l_2 = 6.6600$ m .

Now, contrary to the latter case, the normalized manipulability index (W/W_{max}) is expected to minimize the joint rates maximizing the manipulability ellipsoid volume. In order to show that the joint rate requirement is minimized, the RMS joint torque and joint rate are calculated from equations (2.29) and (2.31) for two different cases applied on the EE, as shown below:

Static force	$\begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix} N$	Eq. (2.29) $\longrightarrow$	$\tau_{RMS} = \begin{bmatrix}9.3227\\8.3591\\2.9345\\0\\0\\0\end{bmatrix} N.m$
Velocity	$\begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix}$ m/s	Eq. (2.31) $\longrightarrow$	$\dot{\boldsymbol{\theta}}_{\mathrm{RMS}} = \begin{bmatrix}0.1150\\0.2337\\0.3464\\0.1136\\0.0487\\0.0500\end{bmatrix}$ rad/s

Note that the joint torques and rates are still between their lower and upper bounds. Comparing these two values with the previous case, it can be seen that the RMS value of the joint torques here is higher than when minimizing W/W_{max}. Conversely, the RMS value of the joint rates here is lower than when minimizing W/W_{max}.

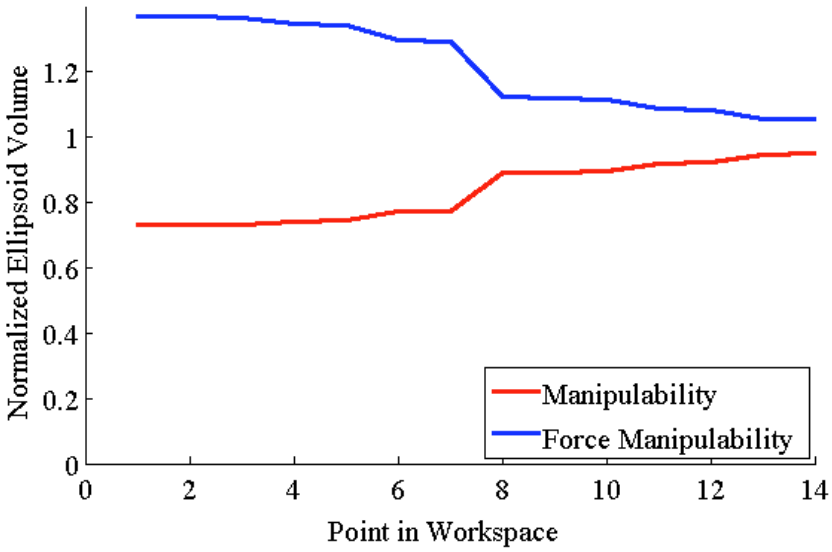


Figure 5.29 Volumes of the normalized velocity and force ellipsoid.

5.8 Conclusion

As a conclusion to the case studies, one can say that that a compromise between velocity maximization and force maximization should be taken, according to the task that the robot is desired to perform. Hence, a weighted objective function is suggested to be formulated as follows:

$$\min\{u_1 \mathcal{W}_{PS} + u_2 \mathcal{W}_{PS-\mathcal{F}}\}$$
$$\text{subject to: } \begin{cases} isreal(\boldsymbol{\theta_{IK}}) = 1 \\ \boldsymbol{\theta_{min}} \le \boldsymbol{\theta_{IK}} \le \boldsymbol{\theta_{max}} \\ \dot{\boldsymbol{\theta}}_{\min} \le \dot{\boldsymbol{\theta}} \le \dot{\boldsymbol{\theta}}_{\max} \\ \boldsymbol{\tau}_{\min} \le \boldsymbol{\tau} \le \boldsymbol{\tau}_{\max} \\ \boldsymbol{lb} \le \{\alpha_1, \alpha_2, l_1, l_2\} \le \boldsymbol{ub} \end{cases}$$

where u_1 and u_2 are the weight coefficients of $\mathcal{W}_{PS}$ and $\mathcal{W}_{PS-\mathcal{F}}$, respectively. If a task demands high velocity and agility, more weight is emphasized on $\mathcal{W}_{PS}$. As for tasks demanding large forces, more weight is stressed on $\mathcal{W}_{PS-\mathcal{F}}$.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

In this study, a generic algorithm for optimizing reconfigurable manipulators, based on a novel modification and global posture-independent implementation of the *parameter of singularity*, was presented. This new index was termed the modified posture-independent parameter of singularity (MPIPS). Optimization was essentially carried out using combined GA and SQP to catch the global optimal solution accurately. The strategy is designed to optimize the DH parameters of a serial robotic manipulator over a pre-specified workspace volume, given several inequality constraints on the design parameters. Several criteria have been tackled, based on the measures of manipulability and isotropy.

Afterwards, a preliminary study was discussed involving force maximization at the EE. It was seen that maximizing the manipulability index could either maximize the velocity ellipsoid or the force ellipsoid. For the case that required uniformity of joint rate/torque distribution, $\mathcal{W}_{PS}$ has been necessarily altered to $\mathcal{W}_{PS-\mathcal{F}}$ in order to maintain this uniformity.

Different optimization case studies using $\mathcal{W}_{PS}$ and $\mathcal{W}_{PS-\mathcal{F}}$ are provided for 3-DOF and 6-DOF reconfigurable manipulators. Results were illustrated to demonstrate the performance of the generated manipulators, and were validated. Post-optimality analysis was also conducted to investigate the sensitivity of the index to the variation in optimal parameters.

Finally, a weighted objective function that balanced between the opposing actions of $\mathcal{W}_{PS}$ and $\mathcal{W}_{PS-\mathcal{F}}$ was concluded. The designer was given the freedom to place more weight on the primary index of interest accordingly.

6.2 Future Work

Since the indices used in performance evaluation are *kinetostatic* indices (dealing with kinematics and statics), there will be a need for studying the dynamics of the robot at hand.

As for the robot's physical properties (length, cross-section diameter, material etc.), these could be taken into account for dynamic and stress optimization too.

The model can also incorporate a more thorough torque and joint rate analysis, especially with trajectory planning.

Obstacle avoidance inside the desired workspace could be taken into account for future work.

Finally, for practical but more complex applications, the assumptions that were considered in the study can be integrated in the calculations, resulting in more accurate and precise calculations.

REFERENCES

Zacharias, F. (2012). *Knowledge Representations for Planning Manipulation Tasks.* Springer-Verlag Berlin Heidelberg.

Zargarbashi, S., Khan, W., & Angeles, J. (2012). The Jacobian condition number as a dexterity index in 6R machining robots. *Robotics and Computer-Integrated Manufacturing , 28* (6), 694-699.

Zhang, G., Liu, P., & Ding, H. (2012). Dynamic Optimization with a New Performance Index for a 2-DoF Translational Parallel Manipulator. *5th International Conference, ICIRA. 7507*, pp. 103-115. Montreal: Springer Berlin Heidelberg.

Wang, L., & Chen, C. (1991). A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. *Robotics and Automation, IEEE Transactions on , 7* (4), 489-499.

Watrous, J. (2011). *Theory of Quantum Information.* Lecture notes, University of Waterloo, Waterloo.

Wenz, M., & Worn, H. (2007). Solving the inverse kinematics problem symbolically by means of knowledge-based and linear algebra-based methods. *Emerging Technologies and Factory Automation* (pp. 1346-1353). IEEE.

Xu, J., Wang, W., & Sun, Y. (2010). Two optimization algorithms for solving robotics inverse kinematics with redundancy. *Journal of Control Theory and Applications , 8* (2), 166-175.

Yahya, S., Moghavvemi, M., & Mohamed, H. (2012). Improvement of Singularity Avoidance for Three Dimensional Planar Manipulators by Increasing their Degrees of Freedom. *International Symposium on Computer, Consumer and Control (IS3C).* IEEE.

Yoshikawa, T. (1985a). Manipulability of robotic mechanisms. *Int. J. Rob. Res. , 4* (2), 3-9.

Yoshikawa, T. (1985b). Dynamic manipulability of robot manipulators. *IEEE International Conference on Robotics and Automation.*, *2*, pp. 1033-1038.

Yoshikawa, T. (1990). Translational and rotational manipulability of robotic manipulators. *American Control Conference* (pp. 228-233). IEEE.

Vasilyev, I., & Lyashin, A. (2010). Analytical solution to inverse kinematic problem for 6-DOF robot-manipulator. *Automation and Remote Control , 71* (10), 2195-2199.

Vinogradov, I., Kobrinski, A., & Stepanenko, Y. (1971). Details of kinematics of manipulators with the method of volumes. . *Mekhanika Mashin* , 5-16.

Abdi, H., & Nahavandi, S. (2012). Well-conditioned configurations of fault-tolerant manipulators. *Robotics and autonomous systems , 60* (2), 242-251.

Aghili, F., & Parsa, K. (2009). A reconfigurable robot with lockable cylindrical joints. *IEEE Trans. on Robotics , 25* (4), 785--797.

Al-Dois, H., Jha, A., & Mishra, R. (2013). Task-based design optimization of serial robot manipulators. *Engineering Optimization , 45* (6), 647-658.

Angeles, J. (2007). *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms* (3rd ed.). New York: Springer-Verlag.

Angeles, J., & Lopez-Cajun, C. (1992). Kinematic isotropy and the conditioning index of serial robotic manipulators. *The International Journal of Robotics Research , 11* (6), 560-571.

Arora, J. (2004). *Introduction to optimum design.* Academic Press.

Buss, S. (2004). Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation , 17*.

Bagchi, S. (2007). *Design Optimization And Synthesis Of Manipulators Based On Various Manipulation Indices.* Masters thesis, The University of Texas Arlington.

Barissi, S., & Taghirad, H. (2008). Task Based Optimal Geometric Design and Positioning of Serial Robotic Manipulators. *Mechatronic and Embedded Systems and Applications, IEEE/ASME International Conference on* , (pp. 158-163).

Belegundu, A., & Chandrupatla, T. (2011). *Optimization Concepts and Applications in Engineering.* Pearson Education Publication .

Bock, R., & Krischer, W. (1998). *The data analysis briefbook.* Springer Berlin Heidelberg.

Castano, A., Behar, A., & Will, P. (2002). The Conro modules for reconfigurable robots. *IEEE/ASME Trans. Mechatron. , 7* (4), 403-409.

Carbone, G., Ottaviano, E., & Ceccarelli, M. (2008). Optimality criteria for the design of manipulators. *Robotics, Automation and Mechatronics, 2008 IEEE Conference on*, (pp. 768-773).

Chang, P. (1988). *A dexterity measure for the kinematic control of robot manipulators with redundancy.* Massachussets Institute of Technology, Massachussets Institute of Technology A.I. Laboratory. Boston: Massachussets Institute of Technology.

Chapelle, F., & Bidaud, P. (2004). Closed form solutions for inverse kinematics approximation of general 6R manipulators. ,. *Mechanism and machine theory , 39* (3), 323-338.

Craig, J. (2005). *Introduction to Robotics* (2nd ed.). Pearson Education Publication.

Elkady, A., Mohammed, M., & Sobh, T. (2009). A new algorithm for measuring and optimizing the manipulability index. *Journal of Intelligent and Robotic systems , 59* (1), 75-86.

Du, Q., Zhang, X., & Zou, L. (2007). Design Optimization of a Minimally Invasive Surgical Robot. *International Conference on Integration Technology* (pp. 8265-8270). Shenzhen, China: IEEE.

Farritor, S., Dubowsky, S., Rutman, N., & Cole, J. (1996). A systems level modular design approach to field robotics. *Proc. IEEE Int. Conf. Robot. Autom.*, (pp. 2890-2895).

Forsythe, G., & Moler, C. (1967). *Computer solution of linear algebraic systems.* New Jersey: Prentice-Hall.

Gupta, K., & Roth, B. (1982). Design consideration for manipulator workspace. *ASME J. Mech. Design , 104*, 704-712.

Golub, G., & Van Loan, C. (2012). *Matrix computations* (Vol. 3). John Hopkins University Press.

Gosselin, C. (1990). Dexterity indices for planar and spherical robotic manipulators. *Proc. 1990 Int. IEEE Conf. Robotics and Automation*, (pp. 650-655).

Gosselin, C., & Angeles, J. (1991). A global performance index for the kinematic optimization of robotic manipulators. *J. Mech. Design , 113* (3), 220-226.

Huo, L., & Baron, L. (2007). Inverse kinematics of functionally-redundant serial manipulators under joint limits and singularity avoidance. *Conference on systems and control.* Morocco.

Huo, L., & Baron, L. (2008). The joint-limits and singularity avoidance in robotic welding. *Industrial Robot , 35* (5), 456-464.

Huo, L., & Baron, L. (2011). The self-adaptation of weights for joint-limits and singularity avoidances of functionally redundant robotic-task. *Robotics and Computer-Integrated Manufacturing , 27*, 367-376.

Hartenberg, R., & Denavit, J. (1964). *Kinematic synthesis of linkages.* New York: McGraw-Hill.

Hebert, P., Tatossian, C., Cairns, M., Aghili, F., & Parsa, K. (2007). Toward the design and simulation of a new generation of reconfigurable space manipulators using telescoping passive joints. *Transactions of the Canadian Society for Mechanical Engineering , 31* (4), 535-545.

Horn, R., & Johnson, C. (1990). *Matrix Analysis.* Cambridge, England: Cambridge University Press.

Jing, Z., & Yi, J. (2007). Dimensional synthesis based on fault tolerant performance for redundant robots. *Proc. 13th Int. Conf. Adv. Robot.*, (pp. 1023-1028).

Kumar, A., & Waldron, K. (1981). The workspace of a mechanical manipulator. *ASME J. Mech. Design , 103*, 665-672.

Keesling, J. *The condition number for a matrix.* University of Florida, Department of Mathematics.

Kim, J., & Khosla, P. (1993). A formulation for task based design of robot manipulators,. *Intelligent Robots and Systems, Proceedings of the 1993 IEEE/RSJ International Conference on* , *3*, pp. 2310-2317.

Khatami, S., & Sassani, F. (2002). Isotropic Design Optimization of Robotic Manipulators Using a Genetic Algorithm Method. *Proceedings of IEEE, Intl. Symposium on Intelligent Control* (pp. 562 - 567). IEEE.

Klein, C., & Blaho, B. (1987). Dexterity measures for the design and control of kinematically redundant manipulators. *The International Journal of Robotics Research , 6* (2), 72-83.

Klein, C., & Miklos, T. (1991). Spatial robotic isotropy. *The International journal of robotics research , 10* (4), 426-437.

Konietschke, R., Ortmaier, T., Weiss, H., Engelke, R., & Hirzinger, G. (2003). Optimal Design of a Medical Robot for Minimally Invasive Surgery. *Jahrestagung der Deutschen Gesellschaft fuer Computer-und Roboterassistierte Chirurgie (CURAC)* .

Krefft, M., & Hesselbach, J. (2006). The dynamic optimization of PKM. *Advances in Robot Kinematics* , 339-348.

Lee, J. (1997). A Study on the Manipulability Measures for Robot Manipulators . *International Conference on Intelligent Robots and Systems* (pp. 1458 - 1465). Grenoble: IEEE.

Leger, C. (1999). *Automated Synthesis and Optimization of Robot Configurations: An Evolutionary Approach.* Carnegie Mellon Univ.

Liegeois, A. (1977). Automatic supervisory control for the configuration and behavior of multibody mechanisms. *IEEE Systems, Man, and Cybernetics Society , 7* (12), 868-871.

Mavroidis, C., Ouezdou, F., & Bidaud, P. (1994). Inverse kinematics of six-degree of freedom" general" and" special" manipulators using symbolic computation. *Robotica , 12* (5), 421-430.

Manocha,, D., & Canny, J. (1994). Efficient inverse kinematics for general 6R manipulators. *Robotics and Automation, IEEE Transactions , 10* (5), 648-657.

Merlet, J. (2006a). Jacobian, manipulability, condition number, and accuracy of parallel robots. *Journal of Mechanical Design , 128* (1), 199-206.

Merlet, J. (2006b). *Parallel Robots* (2nd ed.). New York: Springer-Verlag.

Oetomo, D., Daney, D., & Merlet, J. (2009). Design Strategy of Serial Manipulators With Certified Constraint Satisfaction. *Robotics, IEEE Transactions on , 25* (1), 1-11.

Paul, R., & Stevenson, C. (1983). Kinematics of robot wrists. *The Int. J. Robotics Res. , 2* (1), 31-38.

Paredis, C., & Khosla, P. (1995). Design of modular fault tolerant manipulators. *Proc. 1st Workshop Algorithmic Found. Robot.*, (pp. 371-383).

Paredis, C., & Khosla, P. (1993). Kinematic design of serial link manipulators from task specifications. *The International Journal of Robotics Research , 12* (3), 274-287.

Park, F., & Kim, J. (1998). Manipulability of closed kinematic chains. *Journal of Mechanical Design , 120*, 542-548.

Pieper, D. (1968). *The kinematics of manipulators under computer control.* PhD thesis, Stanford University.

Pham, H., & Chen, I.-M. (2003). Optimal synthesis for workspace and manipulability of parallel flexure mechanism. *11th World Congress in Mechanism and Machine Science.* Tianjin, China.

Salisbury, J., & Craig, J. (1982). Articulated Hands: Force Control and Kinematic Issues. *The International Journal of Robotics Research , 1* (1), 4-17.

Shibata, T., & Ohkami, Y. (2002). Development of brachiating control system for reconfigurable brachiating space robot. *Proc. 3rd Int. Workshop Robot Motion Control*, (pp. 255-259). Poznan, Poland.

Shiller, Z., & Sundar, S. (1991). Design of Robotic Manipulators for Optimal Dynamic Performance. *EEE Conf. of Robotics and Automation*, (pp. 344-349). Sacramento CA.

Snyman, J., & Van Tonder, F. (1999). Optimum design of a three-dimensional serial robot manipulator. *Structural Optimization , 17*, 172-185.

Sobh, T., & Toundykov, D. (2004). Optimizing the tasks at hand [robotic manipulators]. *Robotics & Automation Magazine, IEEE , 11* (2), 78-85.

Stewart, D. (1965). A platform with six degrees of freedom . *Proceedings of the institution of mechanical engineers*, *180*, pp. 371-386.

Raghavan, M., & Roth, B. (1993). Inverse kinematics of the general 6R manipulator and related linkages. *Journal of Mechanical Design , 115* (3), 502-508.

Tanev, T., & Stoyanov, B. (2000). On the performance indexes for robot manipulators. *Problems of engineering cybernetics and robotics , 49*, 64-71.

Tourassis, V., & Ang Jr, M. (1995). Task decoupling in robot manipulators. *Journal of Intelligent and Robotic Systems , 14* (3), 283-302.

APPENDIX A

SIGNIFICANCE OF THE CONDITION NUMBER

The condition number of an $n \times n$ matrix A is

$$\text{cond}(A) = |||A||| \cdot |||A^{-1}|||.$$

This number tells us how accurate we can expect the vector x when solving a system of equations $A \cdot x = b$. We assume that there is an error in representing the vector b, call it ε and otherwise the solution is given to absolute accuracy. That is we solve $A \cdot x = b + \varepsilon$ and get a solution $x + \delta$ where x is the solution of $Ax = b$.

How does the condition number help estimate the number δ? We note that

$$x + \delta = A^{-1}(b + \varepsilon) = A^{-1}b + A^{-1}\varepsilon .$$

Since $A^{-1}b = x$, this gives us the following equation for δ.

$$\delta = A^{-1} \cdot \varepsilon$$

$$|| \delta || \leq |||A^{-1}||| \cdot || \varepsilon ||$$

So, the condition number for the magnitude of the absolute error δ for such a calculation is just the operator norm, $|||A^{-1}|||$.

On the other hand, the relative error is given by $|| \delta || / ||x||$. For the relative error we simply divide the above inequality by the norm of x to get the following inequality.

$$\frac{|| \delta ||}{||x||} \leq \frac{|||A^{-1}||| \cdot || \varepsilon ||}{||x||}$$

However, from the definition of the norm of A, $\frac{||A.x||}{||x||} \leq |||A|||$ and $Ax = b$. So,

$||b|| \leq |||A||| \cdot ||x||$. Thus, combining these inequalities we get the following.

$$\frac{||\,\delta\,||}{||x||} \leq \frac{|||\mathrm{A}^{-1}|||\,\cdot\,||\,\varepsilon\,||}{||x||} \leq \frac{|||\mathrm{A}\,|||\,\cdot\,|||\mathrm{A}^{-1}|||\,\cdot\,||\,\varepsilon\,||}{|||\mathrm{A}\,|||\cdot||x||} = \mathrm{cond(A)}\,\cdot\,\frac{||\,\varepsilon\,||}{||\,\mathrm{b}\,||}$$

So, in solving the equation Ax = b, the relative error in the solution divided by the relative error in the right-hand-side vector is given by the condition number of A. The following rule of thumb is a useful way to express the above estimate. It states that if m = $\log_{10}$ (cond(A)), then m is the number of digits accuracy lost in solving the system of equations Ax = b. There is typically additional error due to the many calculations needed in solving the equations. The estimate for additional losses is given by $\log_{10}(n)$ if the matrix A is $n \times n$.

APPENDIX B

VECTOR AND MATRIX NORMS

B.1 Euclidean Norm

The Euclidean norm |x| (also called the l^2 – norm) is a vector norm defined for a complex vector

$$\mathbf{x} = \begin{vmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{vmatrix}$$

By

$$|\mathrm{x}| = \sqrt{\sum_{\mathrm{k}=1}^{\mathrm{n}} |\mathrm{x_k}|^2}$$

where $|\mathrm{x_k}|$ on the right denotes the complex modulus. The Euclidean norm is the vector norm that is commonly encountered in vector algebra and vector operations (such as the dot product), where it is commonly denoted |x|. However, if desired, a more explicit (but more cumbersome) notation $|\mathrm{x}|_2$ can be used to emphasize the distinction between the vector norm |x| and complex modulus |z| together with the fact that the Euclidean norm is just one of several possible types of norms.

For real vectors, the absolute value sign indicating that a complex modulus is being taken on the right of equation (2) may be dropped. So, for example, the Euclidean norm of the vector

$$\mathbf{x} = \begin{vmatrix} x_1 \\ x_2 \\ x_3 \end{vmatrix}$$

is given by

$$\mathrm{x} = \sqrt{(\mathrm{x_1}^2 + \mathrm{x_2}^2 + \mathrm{x_3}^2)} \tag{3}$$

As for the special case of square matrices, the induced matrix norm is called the spectral norm. The spectral norm of a matrix A is the largest singular value of A i.e. the square root of the largest eigenvalue of the positive-semidefinite matrix A^TA:

$$||\mathrm{A}||_2 = \sqrt{\lambda_{\max}\,(\mathrm{A^TA})} = \sigma_{\max}\,(\mathrm{A})$$

where A^T denotes the conjugate transpose of A.

B.2 Frobenius Norm

The Frobenius norm, sometimes also called the Euclidean norm (which may cause confusion with the above vector l^2-norm which also known as the Euclidean norm), is matrix norm of an *m* x *n* matrix A defined as the square root of the sum of the absolute squares of its elements,

$$||A||_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}$$

(Golub & Van Loan, 2012)

The Frobenius norm can also be considered as a vector norm. It is also equal to the square root of the matrix trace of A^TA, where A^T is the conjugate transpose, i.e.,

$$||A||_F = \sqrt{tr(AA^T)}$$

B.3 Chebyshev Norm

Also known as the *uniform norm*, *supremum norm*, or *infinity norm* (l^{∞}-norm)

In particular, for the case of a vector

$$\mathbf{x} = \begin{vmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{vmatrix}$$

in finite dimensional coordinate space, the Chebyshev norm takes the form:

$$||x||_{\infty} = \max\{|x_1|, |x_2|, \dots, |x_n|\}$$

In the Chebyshev norm, the distance between two sets of points or two lines is just the largest distance between any pair of points or the separation between two lines at the point where they are the farthest apart. A Chebyshev approximation minimizes the maximum distance between the data and the approximating function, hence the occasional name *minimax approximation* (Bock & Krischer, 1998)

APPENDIX C

DEFINITIONS

Kinetostatic performance: the kinematic (related to velocity) and static (related to forces) performance of a manipulator.

Transcendental equations: are equations that contain transcendental functions, i.e. functions that are not algebraic. Examples include trigonometric, exponential, and logarithmic functions.

Z-Y-Z Euler angles: is a method to describe a frame {B} with respect to frame {A}, using three rotations about Z then Y then Z axis, all relative to frame {B}.

Given that a rotation matrix

$$ {}^{A}_{B}R_{Z'Y'Z'}(\alpha,\beta,\gamma) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} $$

already exists, the Z-Y-Z Euler angles can be extracted from that matrix using the following formulas:

If $\sin(\beta) \neq 0$,

$$ \beta = \text{Atan2}(\sqrt{r_{31}^2 + r_{32}^2}, r_{33}) $$

$$ \alpha = \text{Atan2}(r_{23}/\sin(\beta), r_{13}/\sin(\beta)) $$

$$ \gamma = \text{Atan2}(r_{32}/\sin(\beta), -r_{31}/\sin(\beta)) $$

If $\beta = 0$,

$$ \beta = 0 $$

$$ \alpha = 0 $$

$$ \gamma = \text{Atan2}(-r_{12}, r_{11}) $$

If $\beta = 180^0$,

$$\beta = 180°$$

$$\alpha = 0$$

$$\gamma = \text{Atan2}(r_{12}, -r_{11})$$

Newton's method: a method that uses the second order derivative of a function at each step starting from an initial design parameter guess $\mathbf{x_0}$, to find the direction that takes it a step closer towards the minimum (or maximum for - $f(\mathbf{x})$ or $1/f(\mathbf{x})$) of that function, until local convergence is attained after n iterations, at a stationary point $\mathbf{x_n}$.

Lagrangian function: a method for finding the local minima/maxima of a function subject to equality constraints.

KKT conditions: are first order necessary conditions for finding the local minima/maxima of a function subject to equality and inequality constraints. These conditions generalize the Lagrangian function, which can also take into account inequality constraints.

Quadratic Programming: is the optimization of a quadratic function subject to linear constraints.

Accuracy: is the nearness of an actual value to the true value.

Precision: is the closeness of repeated measurements, given that the same conditions are maintained.