Ecole Centrale de Nantes University of Genova

MASTER ERASMUS MUNDUS

EMARO "EUROPEAN MASTER ON ADVANCED ROBOTICS"

2011/2012

Thesis Report

Presented by

Yingcong DENG

on July 16, 2012

KINEMATIC SENSITIVITY ANALYSIS OF TWO DEGREES OF FREEDOM TRANSLATIONAL PARALLEL MANIPULATORS

JURY

President:	Wisama KHALIL	Professor, Ecole Centrale Nantes, IRCCyN, FRANCE
Evaluators:	Christine CHEVALLEREAU	CNRS Research Director, IRCCyN, Nantes
	Philippe WENGER	CNRS Research Director, IRCCyN, Nantes
	Alexandr KLIMCHIK	Postdoctoral fellow, Ecole des Mines de Nantes, IRCCyN, FRANCE
Supervisors:	Stéphane CARO	CNRS Researcher, IRCCyN, Nantes
	Sébastien BRIOT	CNRS Researcher, IRCCyN, Nantes

Laboratory: Institut de Recherche en Communications et Cybernétique de Nantes

Contents

List of Figures

List of Tables

Abstract

This master thesis deals with the sensitivity analysis of two degrees of freedom translational parallel manipulators due to joint clearances and variations in geometric parameters. This aims to find out how the joint clearances and the variations in geometric parameters affect the pose errors of the robots, and then propose some strategies for designing robots.

The IRSBot-2 robot is a novel 2-DOF translational spatial robot, which can perform 2-DOF pick-and-place operations. Its architecture is introduced, the inverse geometry is modelled. Due to its complicated hybrid architecture of the legs, modified Denavit-Hartenberg parameters are used to parameterize this robot. Its singularities are also analyzed.

Local motions due to joint clearances are modelled as error screws in joints. Error screws are mapped to the pose errors by error mapping matrix. An optimization-based method is adopted and improved here to find the maximum positional error and the maximum rotational error due to joint clearances. Sensitivity analysis due to joint clearances for the 5-bar linkage is carried out. Sensitivities to joint clearances for closed-loop and open-loop robots are compared. An actuation redundancy method is proposed to improve the accuracy of the closed-loop/parallel robots. A vector differentiation method for doing sensitivity analysis due to variations in geometric parameters is presented, and is applied for the analysis of the 5-bar linkage.

A new method for doing sensitivity analysis due to joint clearances for robots with hybrid legs is developed here. This method can be applied to robots with hybrid legs and also for robots with simple serial legs. The sensitivity analysis of IRSBot-2 robot is carried out based on this method. Moreover, results of sensitivity analysis by taking into consideraton of the joint clearances in the parallelogram joints are compared with the results obtained without considering the clearances in the parallelogram joints. A tolerance synthesis method is proposed, and is carried out on the 5-bar linkage. A dimension synthesis method is introduced, and is carried out on the IRSBot-2 robot.

Keywords: sensitivity analysis, joint clearance, geometric parameter, 5-bar linkage, IRSBot-2 robot, tolerance synthesis, dimension synthesis.

I

Introduction

I.1 Pick-and-place Robot

I.1.1 Four-DOF Pick-and-place Robot

4-Dof Pick and Place operation is normally of 3-Dof translation and 1-Dof rotation (3T1R), which requires the manipulator to translate independently in the directions of $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$, and rotate around $\boldsymbol{z}$-axis. Usually the manipulator can be assisted with a vision system which provides the position and orientation of the object. So with these sensor data the manipulator will be capable to carry out pick-and-place operations for the desired poses. In food industry, this pick and place operation is quite frequently used, such as Fig. I.1, the manipulator is tracking the food on the conveyor, picking it up and placing it on other conveyor.

Figure I.1 – Mechanism for Tracking with 3 Translation and 1 Rotation

The Pick and Place robot with four degrees of freedom normally can generate 3-DOF independent translational motion and 1-DOF rotational motion (3T1R), namely Schönflies Motion. SCARA robot with 3 revolute joints and 1 prismatic joint organized in a serial way which has been commercialized and widely used in industry is a typical Schonflies Motion Generator. This mechanism is generally employed in various operations like part assembly, conveying systems, food production,e.t.c. Fig. I.2 shows SCARA robot used in food packing, where it is employed to pick up the food in the conveyor and then transmit it to a packing box.

As is stated by many researchers, the serial robots are low in dynamic, stiffness and accuracy. This made the researchers think of designing parallel schonflies-motion generator.

Figure I.2 – SCARA Robot for food packing

Angeles et al.[ACKM06] designed a novel parallel robot that is regarded as McGill Schonflies Motion Generator (SMG), which is shown in Fig. I.3. The McGill SMG is composed of two identical four-degree-of-freedom serial chains in a parallel array, sharing one common base and one common moving platform. The proximal module of each chain is active and has two controlled axes, the motors being installed on the fixed base. The links can thus be made light, thereby allowing for higher operational speeds. The distal module, in turn, is passive and follows the motions of its active counterparts. Therefore the whole mechanism performs four-DOF motion at its moving-platform.

Later, Salgado et al.[SAPH08] presented a new topology of four degrees-of-freedom 3T1R fully-parallel manipulator, which is defined only using lower kinematic pairs, as is shown in Fig. I.4.

It must be added that this topology provides the manipulator some remarkable capabilities. Such as, a high velocity transmission ratio in the inner zones of its workspace and a high stiffness in outer ones. These above mentioned capabilities make the manipulator suitable to perform pick-and-place operations.

I.1.2 Three-DOF Pick-and-place Robot

The pick-and-place operation with 3 DOF can be of two configuration types: 2 DOF of translation and 1 DOF of rotation (2T1R), 3 DOFs of translation (3T). For the 2T1R operation, the manipulator will be able to handle the object to translate along x and z directions. And rotate around z-axis. While for the 3T operation, the manipulator will be able to handle the object independently along x, y and z direction without rotation.

Usually for rapid tracking, it is necessary to have 3 DOF of translations. So that, the robot will be able to handle the object to any position in space within its workspace. Fig. I.5 shows the pick-and-place operation of an object from one conveyor to another. For the 2T1R operation, the robot can handle object in plane composed by $\boldsymbol{x}$ and $\boldsymbol{z}$ axis, and change its orientation around z axis, such as Fig. I.6. Here the center of object and the box are in the

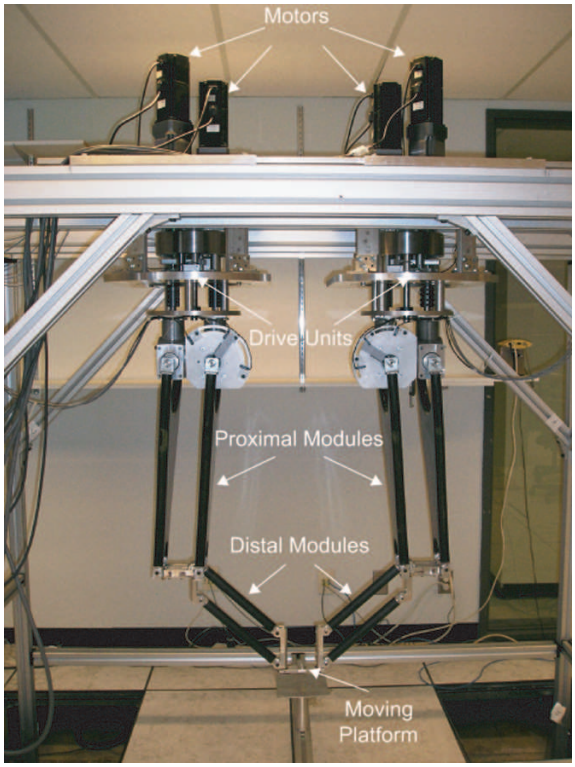


Figure I.3 – The McGill Schönflies-Motion Generator

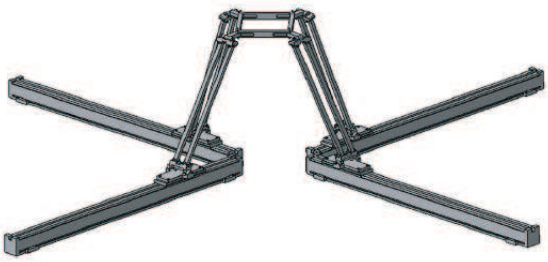


Figure I.4 – CAD Model of the Parallel Manipulator

same plane, and to place the object into the box. The object should be rotated about 90 degrees so as to fit the box.

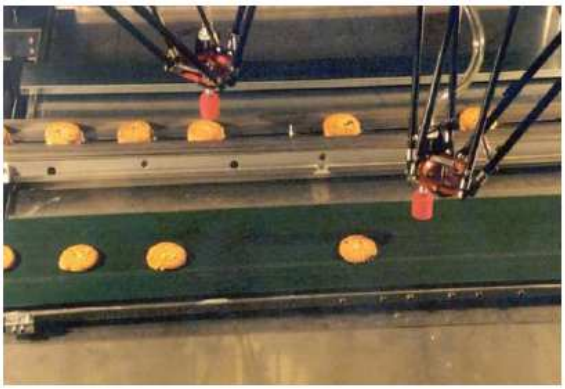


Figure I.5 – Tracking with Translational Motions

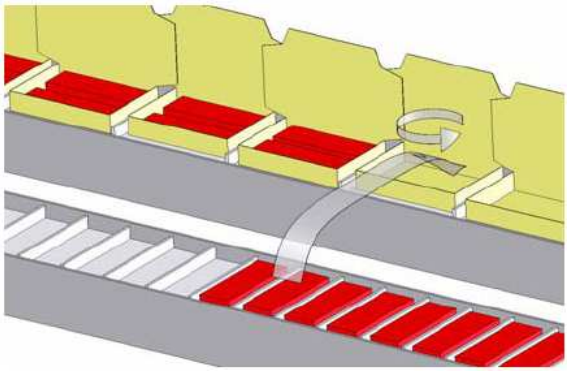


Figure I.6 – 2T1R Type Mechanism for Intermittent Line

The typical robot that can generate 3 independent translations (movement in x, y and z direction) is the Delta robot, which consists of three arms connected to universal joints at the base. The key design feature is the use of parallelograms in the arms, which can maintain the orientation of the end effector. This Fig. I.7 illustrates the Delta robot commercialized by ABB Company. But for the ABB Delta robot, an extra leg is added to make the motion platform rotate around a vertical axis, so as to perform flexible pick-and-place operations.

After the invention of the Delta robot, lots of works have been devoted to the 3-DOF parallel robots. Chablat and Wenger developed a 3-DOF parallel robot (The Orthoglide, Fig. I.8), which can be employed to the 3-DOF translational pick-and-place operations. The input is made of three actuated orthogonal prismatic joints and the output body is connected to prismatic joints through a set of three kinematic chains. Inside each chain, a parallelogram is used so as to restrict the output body to translational motion.

I.1.3 Two-DOF Pick-and-place Robot

Not all the tasks require 3-DOF translational motions, where as some simple tasks use only 2-DOF translation. In very simple production line, 2-DOF operations are sufficient to complete the task. Generally, for the task of 2-DOF pick-and-place operation, the robot

Figure I.7 – ABB Delta Robot

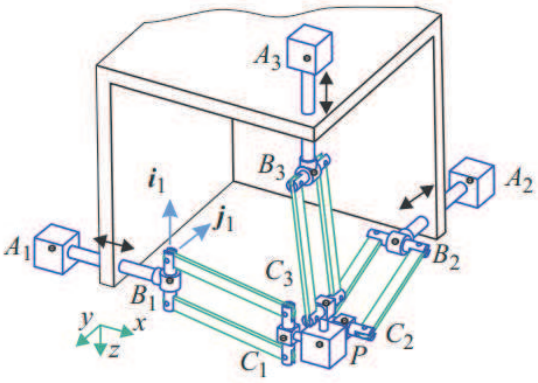


Figure I.8 – Architecture of the Orthoglide

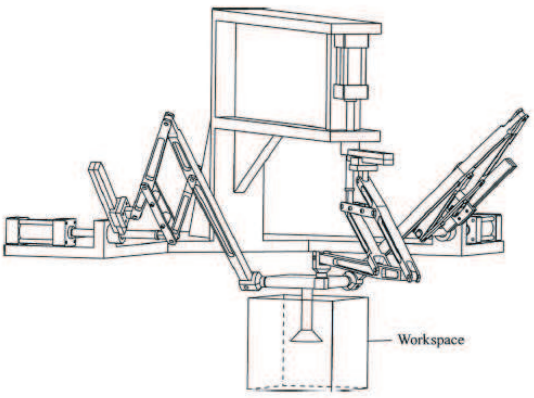


Figure I.9 – CAD View of the Prototype of Pantopteron

handles the object in the plane composed by x and z axis, maintaining the orientation of the object. The object is transformed in one plane. Fig. I.10 shows the manipulation in 2 translational degrees of freedom.

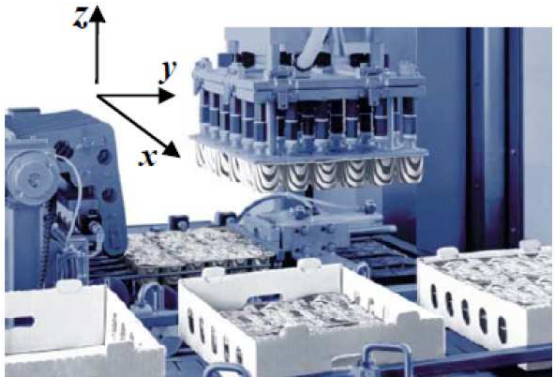


Figure I.10 – Manipulation in 2-DOF Translations

Based on the Delta robot design, a 2D version of the Delta robot was designed by Huang et al. [HLL+04], which is shown in Fig. I.11. The designer used revolute joints, which makes the mechanism cheaper than linear drives. In addition, the actuators can be mounted on the base and the low-mass links can be used, which allows the motion platform to achieve a very high acceleration

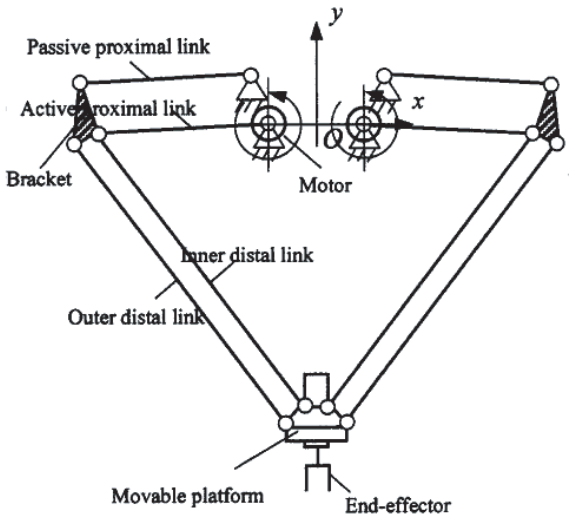


Figure I.11 – Sketch of the 2D Version Delta Robot

But such structure is planar; the stiffness in the direction that is normal to the plane is quite low. In order to overcome this problem, a new Delta-robot like robot named Par2 has been designed, which has been studied in [GBGC11]. As shown in Fig. I.12, in this kind of structure all the elements of the distal parts of the legs are only subject to traction/compression effects, which leads to a lighter structure with better acceleration capacities. The author successfully built a prototype that can reach 53g in acceleration. However, such a complex architecture composed by four identical legs among which two of them are linked by a rigid belt in order to constrain the motion of the moving platform, and in-turn

affects its accuracy. Moreover, the workspace is rather small with four legs.

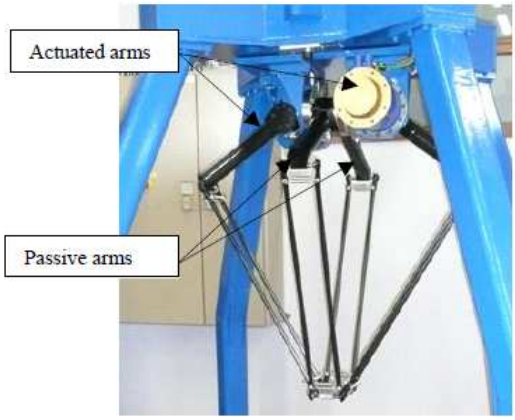


Figure I.12 – Prototype of the Par 2 Robot

Recently, Germain et al.[GBGC11], proposed a novel 2-DOF translational spatial parallel robot, named IRSBot-2 [1]. As shown in Fig. II.1, it has following characteristics:

1. Like the Par2, it has a spatial architecture in which the distal parts of the legs are subject to traction/compression/torsion. As a result, its stiffness is increased and its total mass can be reduced.

2. It is composed of only two legs in order to reduce the mechanism complexity and to increase the size of its Cartesian workspace.

I.2 Sensitivity Analysis Methodology

Sensitivity analysis (SA) is the study of how the uncertainty in the output of a model (numerical or otherwise) can be apportioned to different sources of uncertainty in the model input. IRCCyN [2] wants to develop a 2-DOF translational robot with high accuracy and high acceleration. The objective of the absolute accuracy is about $20\mu m$, and the absolute acceleration is about $20g$. For this, the IRSBot-2 robot has been developed by IRCCyN. Sensitivity analysis should be carried out on IRSBot-2 to reach such a high accuracy of $20\mu m$. Regarding to the error sources that might affect the positional and orientational accuracy, the joint clearances and the variations in geometric parameters are taken into account. Thus, the inputs will be the variations in geometric parameters and joint clearances, and the output will be the positional and orientational errors.

Manufacturing tolerances, installation errors and link offsets cause deviations with respect to the nominal kinematic parameters of the platform. As a result, if the nominal values of these parameters are used within the platform control software, the resulting pose of the platform will be inaccurate [WG01]. Sensitivity analysis has to be carried out before manufacturing the robot, and a tolerance synthesis method should be carried out on the robot, to define the tolerances in joints. This helps the robot to reach required accuracy.

[1] IRCCyN Spatial Robot with 2 DOF
[2] Institut de Recherche en Communications et Cybernétique de Nantes

Parallel Kinematic Manipulators are normally known for high stiffness and high accuracy comparing to their serial counterparts. In fact, PKMs are not necessarily more accurate than their serial counterparts. Even if the dimensional variations can be compensated with PKM, but they can also be amplified contrary to their serial counterparts [WG01] and [WM93].

I.2.1 Sensitivity Analysis Due to Variations in Geometric Parameters

Wang and Masory [WM93] seem to be the first to evaluate the accuracy of Stewart parallel robot. They develop a kinematic model which will accommodate the errors of manufacturing tolerances and installation errors, and present algorithms for forward and inverse kinematics of the platform, and the pose error due to the effects of tolerances is computed with the following algorithm:

1. A random error vector within the specified range was added to the nominal parameters vector.
2. A set of desired end-effector trajectories, $\boldsymbol{x}^n$, within the workspace were specified.
3. The six joint lengths variables were computed by the nominal inverse kinematic solution.
4. The actual end-effector trajectory, $\boldsymbol{x}^a$, was obtained by the general forward kinematic using the parameters of step 1.
5. The pose errors were computed from $\boldsymbol{x}^n$ and $\boldsymbol{x}^a$.

This algorithm is quite simple and easy to implement. But it requires the forward kinematic model. For a parallel robot normally there is no analytic model for doing the forward kinematic model. A numerical method should be used, and it will lead to some numerical errors that might be comparable with the tolerance, which might make the estimated pose errors not so accurate. But if we want to reduce the numerical error for the forward kinematic model, more iteration will be required for doing the computation, which is time consuming. Moreover, this method can not specify the range of the pose error.

Kim and Choi [KC00] presented the development of methods of the forward and inverse error bound analyses of the Stewart platform. The forward error bound analysis is used to find the error bound of the end-effector when the error bounds of the joints are given, while the inverse error bound analysis is used to determine those of the joints for the given error bound of the end-effector. The inverse error bound analysis is important since it provides a designer with a way to determine the tolerance limits of the joints. The forward error is computed with the inverse of the manipulator Jacobian matrix. For this method, the manipulator Jacobian sometimes may not be square matrix or might be singular, so the inverse operation can not be implemented.

Besides, Kim and Tsai [KT03] studied the effect of misalignment of linear actuators of a 3-DOF translational parallel manipulator on the motion of its moving platform. Han et al. [HKKP02] analyze the kinematic sensitivity of the three degree-of-freedom 3-UPU parallel mechanism, and show that the 3-UPU is highly sensitive to certain minute clearances in the

universal joint, and that a careful kinematic sensitivity analysis of the 3-UPU augmented with virtual joints satisfactorily explains the gross motions, which is confirmed by a hardware experimental prototype. Fan et al. [FWZC03] propose and analyze a sensitivity model of the spindle platform subject to the structure parameters by partial differentiation of the structural parameters.

Since 2005, Caro et al. have shown great interests in this field. In 2005, Caro et al.[CBW05] developed a tolerance synthesis method for mechanisms based on a robust design approach, and the general issue is to find a robust mechanism for a given task, and to compute its optimal dimensional tolerances. Later, Caro et al. [CWBC06] introduce two complementary methods: linkage kinematic analysis and differential vector methods, to analyze the sensitivity of a three-degree-of-freedom (3-DOF) translational parallel kinematic machine (PKM) with orthogonal linear joints. Caro et al.[CBW09] also deals with the sensitivity analysis of 3-RPR planar parallel manipulators (PPMs), and the sensitivity coefficients of the pose of the manipulator moving platform to variations in the geometric parameters and in the actuated variables are expressed algebraically. Moreover, 2 aggregate sensitivity indices are determined, one related to the orientation of the manipulator moving platform and another one related to its position. Afterwards, Binaud and Caro [BCW09] analyze the sensitivity of degenerate and non-degenerate planar parallel manipulators, the sensitivity of the pose of their moving platform to variations in the geometric parameters is analyzed.

I.2.2 Sensitivity Analysis Due to Joint Clearances

Joint clearances [BCC10], exhibit low repeatability, generally make their compensation difficult. For this reason, the impact of joint clearances on the pose errors of serial and parallel-mechanism moving platforms should be studied.

Wang and Roth [WR89] essentially proposed a model for clearances in revolute joints embodied as journal bearings. Voglewede and Uphoff [VEU04] shows precisely how much unconstrained end effector motion exists at the end effector for a large class of parallel manipulators, namely those with passive revolute and/or spherical joints, if all the joint clearances are known. Fig. I.13 [VEU04] shows the unconstrained end-effector motions to joint clearances when the actuated joints are locked. Venanzi and Parenti-Castelli [VPC05] develop a technique for assessing the influence of the clearance in the kinematic pairs of a mechanism, and the technique works for both planar and spatial, open-chain and closed-chain mechanisms, but not for over constrained mechanisms.

Recently, Meng et al.[MZL09] modified the formulation proposed in [VPC05], the error prediction model is applicable to planar or spatial parallel manipulators that are either over constrained or non over constrained. By formulating the problem as a standard convex optimization problem, the maximal pose error in a prescribed workspace can be efficiently computed, but in this paper, the author find the maximum error along each direction. actually the total maximum error of the end-effector can be over estimated. Later, Binaud et al. [BCC10] present a method for analyzing the sensitivity for both serial and parallel robot caused by joint clearances, and defined two indices: one for reference-point displacements of the moving platform and another for its rotations. They first presented the error estimation

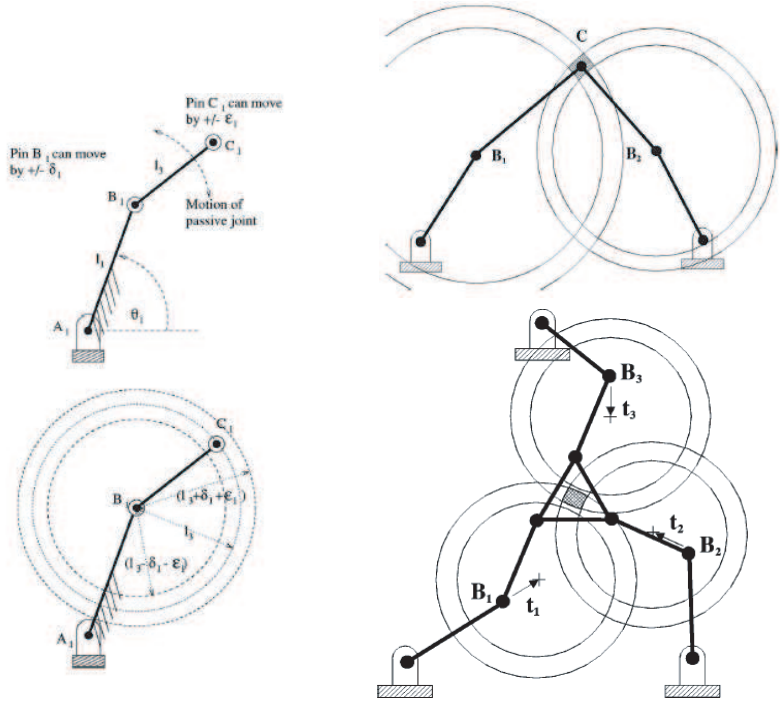


Figure I.13 – Unconstrained End Effector Motion

model, and model the local motion termed by the joint clearance as an error screw, which is constrained by the tolerance in the joint. And then an optimization based method was proposed to find the maximum positional and rotational error of the end-effector. Since the optimization problems included not just inequality constraints, but also equality constraints, they integrated the equality constraints into the inequality constraints based on a QR factorization method for the error mapping matrix of the kinematic chains. Finally, optimization problems with only inequality constraints for finding maximum positional and rotational errors were developped. The overal idea for doing sensitivity analysis due to joint clearances in this paper [BCC10] is nice. And it will be adopted here for doing sensitivity analysis due to joint clearances.

However, in paper [BCC10], there are some errors, wrong comments and drawbacks. The author presented that the translational parts of the error screw in joints did not affect the rotational error of the end-effector. But actually, this is just true for serial robots, for parallel robots, the rotational error will be affected by both of the translational parts and the rotational parts of the error screw. In addition, the author did not consider the idle motions termed by joint clearances for passive joints in parallel robots. When doing the analysis for the closed-loop robot, like for the 5-bar linkage illustrated in the paper, the author just assigned a small values for the rotational motion of the passive joints. But actually, due to the joint clearances, the rotational motions about the joint axis of the passive joints will be larger than the rotational motion abouth the axis in the sectional plane of the joint, larger range of the rotational motion about the joint axis should be assigned. In addition, the method presented in paper [BCC10] can only be used to do the sensitivity analysis due to joint clearances of robots with simple serial kinematic chains in legs, for a more general

robot, that might contain complicated hybrid legs, this method is not sufficient. Therefore, the method needs to be adapted and improved.

I.3 Thesis Constributions and Outline

In this thesis, models of the kinematic sensitivity analysis due to joint clearances and variations in geometric parameters are developed. The method for doing sensitivity analysis due to joint clearances presented in paper [VPC05] is adopted and improved. And the vector differential method presented in paper [CWBC06] is recalled and adopted here.

The sensitivity for the 5-bar linkage due to the joint clearances and variations in geometric parameters are studied. Based on the improved method for sensitivity analysis due to joint clearances, the sensitivity if performed to the two open-loop serial kinematic chains decomposed from the 5-bar linkage separately as well, and the results are compared with the closed-loop 5-bar mechanism. And finally, an actuation redundancy method, which is to actuate the passive joints in the closed-loop/parallel robots, to improve the accuracy of the robot is proposed.

Since the IRSBot-2 robot is a very complicated spatial robot, with hybrid kinematic chains in legs, for doing sensitivity analysis due to joint clearances, the method based on paper [BCC10] is not sufficient. Thus, a more general sensitivity analysis method to joint clearances, which can be used for doing the sensitivity analysis for robots with complicated hybrid legs. Based on the newly developped, the sensitivity analysis due to joint clearances for the IRSBot-2 robot is done.

In addition, a tolerance synthesis method is proposed and applied for the 5-bar linkage based on the sensitivity analysis method to joint clearances, so as to know that to design a robot with a certain accuracy, which level of tolerances in the joint should be assigned. And the dimension synthesis is done for the IRSBot-2 robot, so as to know that to study how the dimension affect the accuracy of the IRSBot-2 robot, with knowing the clearances in joints.

The thesis is organized in this way, Chapter II studies the inverse geometric model and direct geometric model, as well as its singularities, such as Type 1 singularity, Type 2 singularity and the constraint singularity. Chapter III presents the sensitivity analysis methods to joint clearances and variations in geometric parameters for robots. Chapter IV studies the sensitivity analysis due to joint clearances and variations in geometric parameters of the 5-bar linkage, the sensitivities for the closed-loop and open-loop mechanisms are compared, and an actuation redundancy method is proposed. Chapter V develops the sensitivity analysis method to joint clearances for robots with complicated hybrid legs. Chapter VI studies the sensitivity analysis due to joint clearances for the IRSBot-2 robot, the sensitivity analysis due to joint clearances in the distal module and to both the joint clearances in the distal module and the parallelograms are studied and compared. Chapter VII develops the tolerance synthesis method and the dimension synthesis method. The tolerance synthesis method is applied to the 5-bar linkage, and the dimension synthesis method is applied to the IRSBot-2 robot.

II

Modelling of the IRSBot-2 Robot

IRSBot-2 robot is a novel 2-DOF translational spatial robot. It can be used in performing 2-DOF translational pick-and-place operations. Structurally, it is composed with two symmetric legs, each of which contains the proximal module and the distal module. The proximal module is a parallelogram, which is able to maintain the orientation of the end-effector. The distal module is of a spatial structure, and can be regarded as one virtual link.

This chapter is based on the work in the thesis [Ger10] and paper [GBGC11],[GBC12] of Germain. Section II.1 presents the architecture and the parameterization of the IRSBot-2 robot. Section II.2 studies the inverse geometric model of the IRSBot-2 robot, with inverse geometric model for the equivalent planar mechanism and inverse geometric model for the spatial mechanism presented. Section II.3 studies the kinematic notations for the closed-loop and tree structures of the IRSBot-2 robot, and presents the modified Denavit-Hartenberg parameters. Section II.4 studies the singularities of the IRSBot-2 robot, the parallel singularity, serial singularity and the constraint singularity are analyzed.

II.1 Architecture of the IRSBot-2 Robot

IRSBot-2 robot is a novel 2-DOF translational spatial robot. As shown in Fig. II.1, it can translate in the plane constructed by the $\boldsymbol{x}_0$ axis and $\boldsymbol{z}_0$ axis with 2 degrees of freedom. The robot has two legs, each of which contains a proximal module and a distal module. The proximal module is constructed by a parallelogram, the so called II joint. The distal module is constructed by 2 links with equal length connected to the moving-platform and the elbow by universal joints. The parallelogram can maintain the orientation of the elbow plane. Together with constraints of the distal module, the orientation of the moving platform will be constantly parallel to the base. But it should be noted that the 2 links in the distal module can not be parallel with each other, otherwise it will become a spatial parallelogram, thus the robot will be in singularity.

The scheme of the IRSBot-2 robot is drawn in Fig. II.2. The leg of the robot is a hybrid leg, with two close loop, parallelogram $A_iB_iC_iD_i$ and distal module $E_{i1}F_{i1}E_{i2}F_{i2}$ connected in a serial way. The elbow plane $\wp_{i1}$ is attached to the end of the parallelogram. Thanks to the parallelogram, the orientation of the elbow plane will keep constantly parallel to the plane $\wp_0$ attached to the base. Distal module is equivalent as a virtual link $H_{bi}H_{hi}$. Thus, the IRSBot-2 robot is equivalent as a plannar robot, as Fig. II.3.

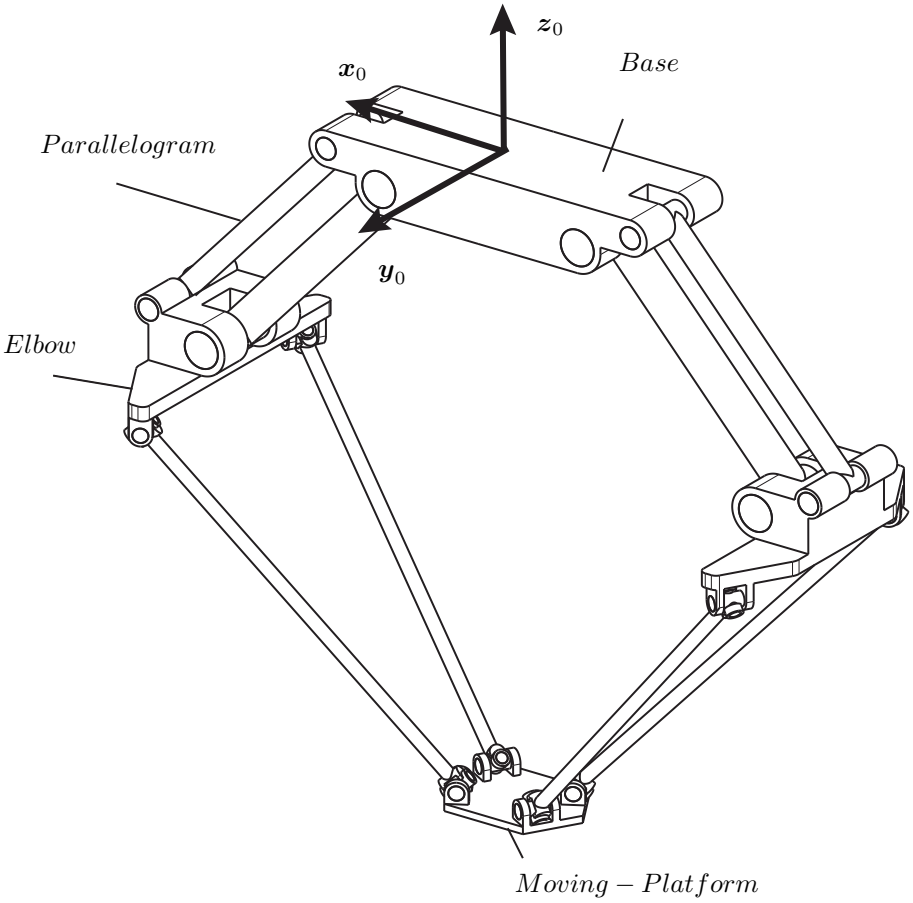


Figure II.1 – CAD Model of the IRSBot-2 Robot

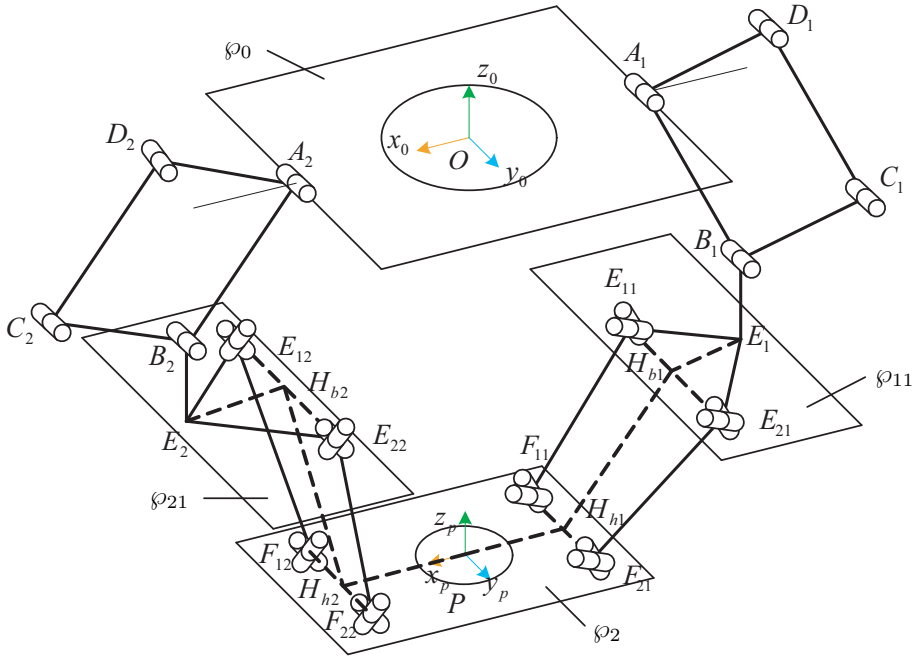


Figure II.2 – Kinematic Model of the IRSBot-2 Robot

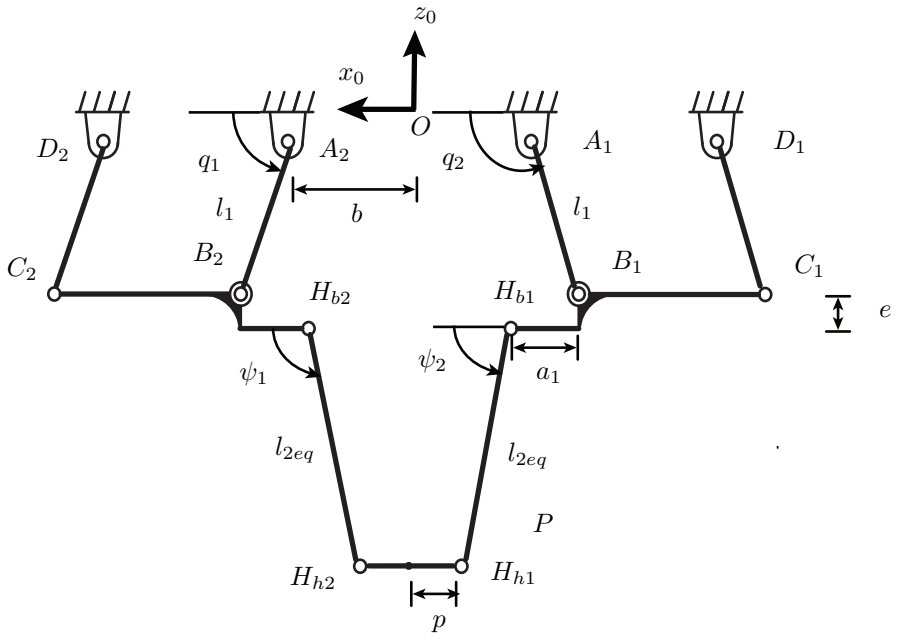


Figure II.3 – Equivalent Planar Mechanism for IRSBot-2

IRSBot-2 robot can be parameterized as is shown in Fig. II.4. Let q_i be the actuated joint coordinate of the $i-$th leg $(i = 1, 2)$, $b = OA_i$ the radius of the base, $l_1 = A_iB_i$ the length of the proximal legs, $l_2 = E_{ij}F_{ij}$ the length of the spatial distal legs, a_1 and a_2 denote the lengths of E_iE_ji respectively. One can notice that the angle between E_iE_{ji} is constant and is equal to β_{ji}. Points H_{bi} and H_{hi} are defined as the midpoints of $E_{1i}E_{2i}$ and $F_{1i}F_{2i}$ respectively. By construction, H_{bi} lies in plane $\wp_1$ and H_{hi} in plane $\wp_2$. ψ_i is the angle between axis x_0 and the line defined by $H_{bi}H_{hi}$. From the Pythagorean theorem, the length between points H_{bi} and H_{hi} is constant and equal to

$$l_{2eq} = \sqrt{l_2^2 - (a_1 - a_2)^2 \cos\beta} \tag{II.1}$$

II.2 Iverse Geometric Model of the IRSBot-2 Robot

II.2.1 Inverse Geometry of the Equivalent Plannar Architecture

As shown in Fig. II.3, the closed-loop equation of the equivalent planar mechanism is

$$\vec{OP} = \vec{OA_i} + \vec{A_iB_i} + \vec{B_iE_i} + \vec{E_iH_{hi}} + \vec{H_{hi}H_{bi}} + \vec{H_{bi}P} \tag{II.2}$$

And based on Eq. (II.2), the closed-loop equation for Leg 1 can be derived

$$\begin{pmatrix} x \\ z \end{pmatrix} = \begin{pmatrix} -b \\ 0 \end{pmatrix} + \begin{pmatrix} l_1 \cos q_1 \\ -l_1 \sin q_1 \end{pmatrix} + \begin{pmatrix} 0 \\ -e \end{pmatrix} + \begin{pmatrix} a_1 \sin\beta \\ 0 \end{pmatrix} + \begin{pmatrix} l_{2eq} \cos\psi_1 \\ -l_{2eq} \sin\psi_1 \end{pmatrix} + \begin{pmatrix} p \\ 0 \end{pmatrix} \tag{II.3}$$

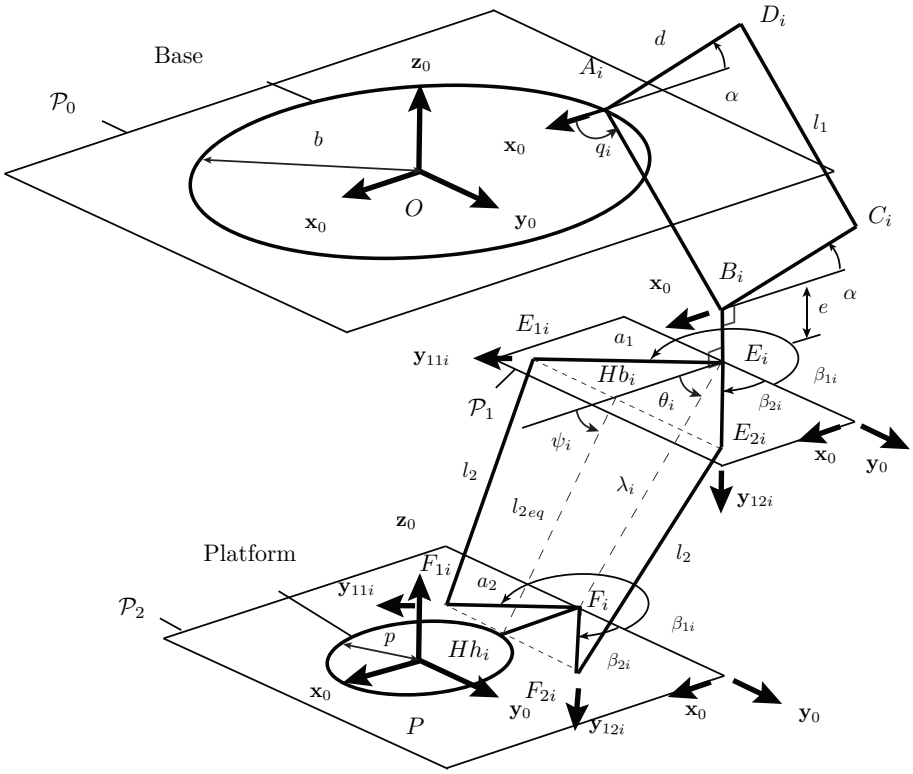


Figure II.4 – Parameterization of the IRSBot-2 robot

In the same way, the closed-loop equation for Leg 2 can be derived

$$\begin{pmatrix} x \\ z \end{pmatrix} = \begin{pmatrix} b \\ 0 \end{pmatrix} + \begin{pmatrix} l_1 \cos q_2 \\ -l_1 \sin q_2 \end{pmatrix} + \begin{pmatrix} 0 \\ -e \end{pmatrix} + \begin{pmatrix} -a_1 \sin \beta \\ 0 \end{pmatrix} + \begin{pmatrix} l_{2eq} \cos \psi_2 \\ -l_{2eq} \sin \psi_2 \end{pmatrix} + \begin{pmatrix} -p \\ 0 \end{pmatrix} \tag{II.4}$$

Germain analyzed inverse geometric model of the IRSBot-2 in her thesis [Ger10]. But just the inverse geometric models for the joint variables of the actuated joints were given. In this section, the inverse geometric models for actuated joints are recalled, and the models for the passive joints will be derived.

$$q_1 = 2 \tan^{-1} \left(\frac{-B_1 \pm \sqrt{B_1^2 + A_1^2 - C_1^2}}{C_1 - A_1} \right) \tag{II.5}$$

with,

$$\begin{aligned} A_1 &= -2l_1(x + b - p - a_1 \sin \beta) \\ B_1 &= 2l_1(z + e) \\ C_1 &= (x + b - p - a_1 \sin \beta)^2 + (z + e)^2 + l_1^2 - l_{2eq}^2 \end{aligned}$$

$$q_2 = 2\tan^{-1}\left(\frac{-B_2 \pm \sqrt{B_2^2 + A_2^2 - C_2^2}}{C_2 - A_2}\right) \tag{II.6}$$

with,

$$\begin{aligned} A_2 &= -2l_1(x - b + p + a_1\sin\beta) \\ B_2 &= 2l_1(z + e) \\ C_2 &= (x - b + p + a_1\sin\beta)^2 + (z + e)^2 + l_1^2 - l_{2eq}^2 \end{aligned}$$

For both Eq. (II.5) and Eq. (II.6), the following constraints should be satisfied

$$\forall x, z, s.t. B_i^2 + A_i^2 \geqslant C_i^2, i = 1, 2$$

Based on Eq. (II.3) and Eq. (II.4), the relations of joint variables ψ_1, ψ_2 and the actuated joint variables q_1,q_2 of the equivalent planar mechanism shown in Fig. II.3 can be derived

$$\tan\psi_1 = \frac{-z - l_1\sin(q_1) - e}{x + b - l_1\cos(q_1) - a_1\sin(\beta) - p} \tag{II.7}$$

$$\tan\psi_2 = \frac{-z - l_1 sin(q_2) - e}{x + b - l_1 cos(q_2) - a_1\sin(\beta) - p} \tag{II.8}$$

Certainly, the values of ψ_1 and ψ_2 can be easily computed by the function arctan. But it is noteworthy that value returned by arctan function is in the range of $[-\frac{\pi}{2}, \frac{\pi}{2}]$, while for the values of ψ_1 and ψ_2, they can be in the range of $[-\pi, \pi]$. It depends on which quadrant the links $H_{b1}H_{h1}$ and $H_{b2}H_{h2}$ lie in. In Matlab, there is a very useful function called $atan2$, which can distinguish in which quadrant it is by detecting from the signs of the numerator and denumerator, and give the exact corresponding values.

II.2.2 Inverse Geometry of the Spatial Distal Module

As we know, the universal joint can be equivalent as two orthogonal revolute joints. Here we choose the architecture that the axis of the first revolute joint in the elbow is parallel to the Plane x_0Oz_0, so is the second revolute joints in the moving platform. Translating the base frame to the center of the universal joint, it is denoted as Frame $x_0'y_0'z_0'$, as shown in Fig. II.5 and Fig. II.6. For the first leg, the joint axis of Joint $\theta_{j1,7}(j = 1, 2)$, is in the plane constructed by axis x_0' and z_0', the angles between the joint axis $z_{j1,7}$ and z_0' is α_d, and the angles between joint axis $z_{j1,10}$ is also α_d. For the second leg, angles between $z_{j2,k}(j = 1, 2, k = 7, 10)$, is $-\alpha_d$.

The D-H frames can be denoted as is shown in Fig. II.6. Transformation matrix are applied to deduce the relative position between Point E_{11} and Point F_{11}. The tranformation can be done in the following steps:

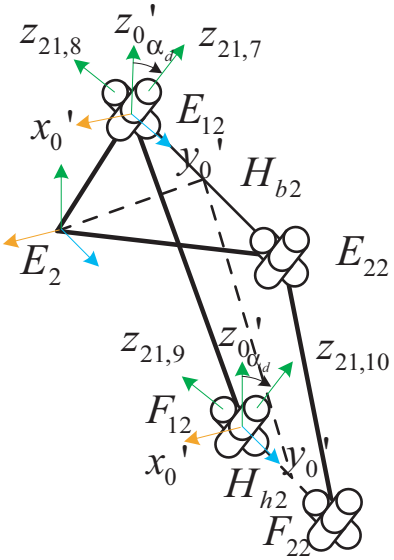


Figure II.5 – Distal Module of Leg 2

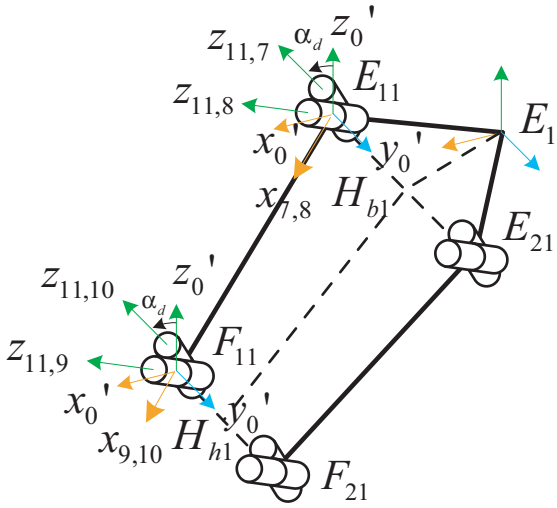


Figure II.6 – Distal Module of Leg 1

First, rotate axis z_0' around axis y_0' with a rotation angle α_d to $z_{11,7}$

$$\mathbf{T}_1 = \begin{bmatrix} \cos\alpha_d & 0 & \sin\alpha_d & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\alpha_d & 0 & \cos\alpha_d & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{II.9}$$

Second, rotate axis $x_{11,7}$ around axis $z_{11,7}$ with a rotation angle $\theta_{11,7}$

$$\mathbf{T}_2 = \begin{bmatrix} \cos\theta_{11,7} & -\sin\theta_{11,7} & 0 & 0 \\ \sin\theta_{11,7} & \cos\theta_{11,7} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{II.10}$$

Thirdly, rotate axis $z_{11,7}$ around $x_{11,7}$ to axis $z_{11,8}$ with a rotation angle of $\frac{1}{2}\pi$

$$\mathbf{T}_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{II.11}$$

Forthly, rotate axis $x_{11,8}$ around axis $z_{11,8}$ with a rotation angle $\theta_{11,8}$

$$\mathbf{T}_4 = \begin{bmatrix} \cos\theta_{11,8} & -\sin\theta_{11,8} & 0 & 0 \\ \sin\theta_{11,8} & \cos\theta_{11,8} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{II.12}$$

Finally, translate along axis $x_{11,8}$ with a displacement l_2, the origin of the frame will be at Point F_{11}

$$\mathbf{T}_5 = \begin{bmatrix} 1 & 0 & 0 & l_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{II.13}$$

Multiplying these transformation matrix, here we obtain

$$\mathbf{T}_{d11} = \mathbf{T}_1\mathbf{T}_2\mathbf{T}_3\mathbf{T}_4\mathbf{T}_5 \tag{II.14}$$

Extracting from the translational part from matrix $\mathbf{T}_{d11}$, we get the position of Point H_{h1} with respect to Point H_{b1}

$$\boldsymbol{t}_{d11} = \begin{bmatrix} l_2 \sin\alpha_d \sin\theta_8 + l_2 \cos\alpha_d \cos\theta_7 \cos\theta_8 \\ l_2 \cos\theta_8 \sin\theta_7 \\ l_2 \cos\alpha_d \sin\theta_8 - l_2 \sin\alpha_d \cos\theta_7 \cos\theta_8 \end{bmatrix} \tag{II.15}$$

Therefore,the position of Point F_{11} can be expressed with respect to Point H_{b1} as Eq. (II.16)

$$\begin{aligned} x_{F11} &= l_2 \sin\alpha_d \sin\theta_8 + l_2 \cos\alpha_d \cos\theta_7 \cos\theta_8 \\ y_{F11} &= -a_1 \cos\beta \\ z_{F11} &= l_2 \cos\alpha_d \sin\theta_8 - l_2 \sin\alpha_d \cos\theta_7 \cos\theta_8 \end{aligned} \tag{II.16}$$

In another way, based on the virtual link $H_{b1}H_{h1}$, the position of Point F_{11} can be deduced with respect to Point H_{b1}, as Eq. (II.17)

$$\begin{aligned} x_{F11} &= l_{2eq} \cos\psi_1 \\ y_{F11} &= -a_2 \cos\beta \\ z_{F11} &= -l_{2eq} \sin\psi_1 \end{aligned} \tag{II.17}$$

Combine the equations of Eq. (II.16) and Eq. (II.17), the angles of $\theta_{11,7}$ and $\theta_{11,8}$ can be solved as the following

$$\theta_{11,8} = \arcsin\left(\frac{\cos\alpha_d z_{F11} + \sin\alpha_d x_{F11}}{l_2}\right) \tag{II.18}$$

$$\theta_{11,7} = \arcsin\left(\frac{a_1 \cos\beta + y_{F11}}{l_2 \cos\theta_{11,8}}\right) \tag{II.19}$$

where, $x_{F11}, y_{F11}, z_{F11}$ are expressed in Eq. (II.17)

Since the moving plane $\wp_2$ is always parallel to the elbow plane $\wp_{11}$, and the axis of the revolute joints have the following relations

$$z_{11,7} \parallel z_{11,10} \tag{II.20}$$

$$z_{11,8} \parallel z_{11,9} \tag{II.21}$$

The rotation angles $\theta_{11,9}$ and $\theta_{11,10}$ can be easily derived

$$\theta_{11,9} = -\theta_{11,8} \tag{II.22}$$

$$\theta_{11,10} = -\theta_{11,7} \tag{II.23}$$

In the same way, the universal joints in the kinematic chain $E_{21}F_{21}$, $E_{12}F_{12}$ and $E_{22}F_{22}$ can be solved

For kinematic chain $E_{21}F_{21}$,

$$\theta_{21,8} = \arcsin\left(\frac{\cos\alpha_d z_{F21} + \sin\alpha_d x_{F21}}{l_2}\right) \tag{II.24}$$

$$\theta_{21,7} = \arcsin\left(\frac{-a_1\cos\beta + y_{F21}}{l_2\cos\theta_{21,8}}\right) \tag{II.25}$$

and therefore,

$$\theta_{21,9} = -\theta_{21,8} \tag{II.26}$$

$$\theta_{21,10} = -\theta_{21,7} \tag{II.27}$$

where,

$$\begin{aligned} x_{F21} &= l_{2eq}\cos\psi_1 \\ y_{F21} &= a_2\cos\beta \\ z_{F21} &= -l_{2eq}\sin\psi_1 \end{aligned}$$

For kinematic chain $E_{12}F_{12}$,

$$\theta_{12,8} = \arcsin\left(-\frac{\cos\alpha_d z_{F12} + \sin\alpha_d x_{F12}}{l_2}\right) \tag{II.28}$$

$$\theta_{12,7} = \arcsin\left(\frac{-a_1\cos\beta - y_{F12}}{l_2\cos\theta_{12,8}}\right) \tag{II.29}$$

and therefore,

$$\theta_{12,9} = -\theta_{12,8} \tag{II.30}$$

$$\theta_{12,10} = -\theta_{12,7} \tag{II.31}$$

where,

$$\begin{aligned} x_{F12} &= -l_{2eq}\cos\psi_2 \\ y_{F12} &= -a_2\cos\beta \\ z_{F12} &= -l_{2eq}\sin\psi_2 \end{aligned}$$

For kinematic chain $E_{22}F_{22}$,

$$\theta_{22,8} = \arcsin\left(-\frac{\cos\alpha_d z_{F22} + \sin\alpha_d x_{F22}}{l_2}\right) \tag{II.32}$$

$$\theta_{22,7} = \arcsin\left(\frac{a_1\cos\beta - y_{F22}}{l_2\cos\theta_{22,8}}\right) \tag{II.33}$$

and therefore,

$$\theta_{22,9} = -\theta_{22,8} \tag{II.34}$$

$$\theta_{22,10} = -\theta_{22,7} \tag{II.35}$$

where,

$$\begin{aligned} x_{F22} &= -l_{2eq}\cos\psi_2 \\ y_{F22} &= a_2\cos\beta \\ z_{F12} &= -l_{2eq}\sin\psi_2 \end{aligned}$$

II.3 Denavit-Hartenberg Parameters of the IRSBot-2 Robot

For sorting out the transformation matrix from frame to frame, the Denavit-Hartenberg method(D-H method) [DH55] is applied here. D-H frames are denoted for IRSBot-2 as shown in Fig. II.7. Plane $\wp_0$, where the reference frame is attached, is fixed. In IRSBot-2, there are two elbow planes attached to the end of the parallograms, Plane $\wp_{11}$ and Plane $\wp_{21}$. Plane $\wp_2$ is the moving plane. It can be seen from the figure that the IRSBot-2 robot is a spatial robot with complex architecture, there are two legs connecting from the base to the moving platform. Each leg contains the proximal module and the distal module, and both them are closed-loops. For the convenience of analyzing the distal module, a frame that is identity with the base frame is attached to the each elbow plane.

The IRSBot-2 robot is a complicated spatial robot, and it contains closed-loop and tree structures in its leg. The well-known Denavit-Hartenberg notations is powerful for serial robots, but leads to ambiguities in the case of tree and closed-loop structures[Kha86]. Thus, a new kinematic notation for robots was proposed by Khalil[Kha86], let's say the modified D-H parameters.

The modified D-H parameters of Leg 1 is shoen in Tab. II.1 and parameters for Leg 2 is shown in Tab. II.2. All the joint variables in Tab.II.1 and Tab. II.2 are based on the inverse geometric model of the IRSBot-2 robot. The transformation matrix $\mathbf{S}$ can be conveniently constructed by D-H parameters, and the relative pose between frames denoted in Fig. II.7 can be easily deduced by the transformation matrix.

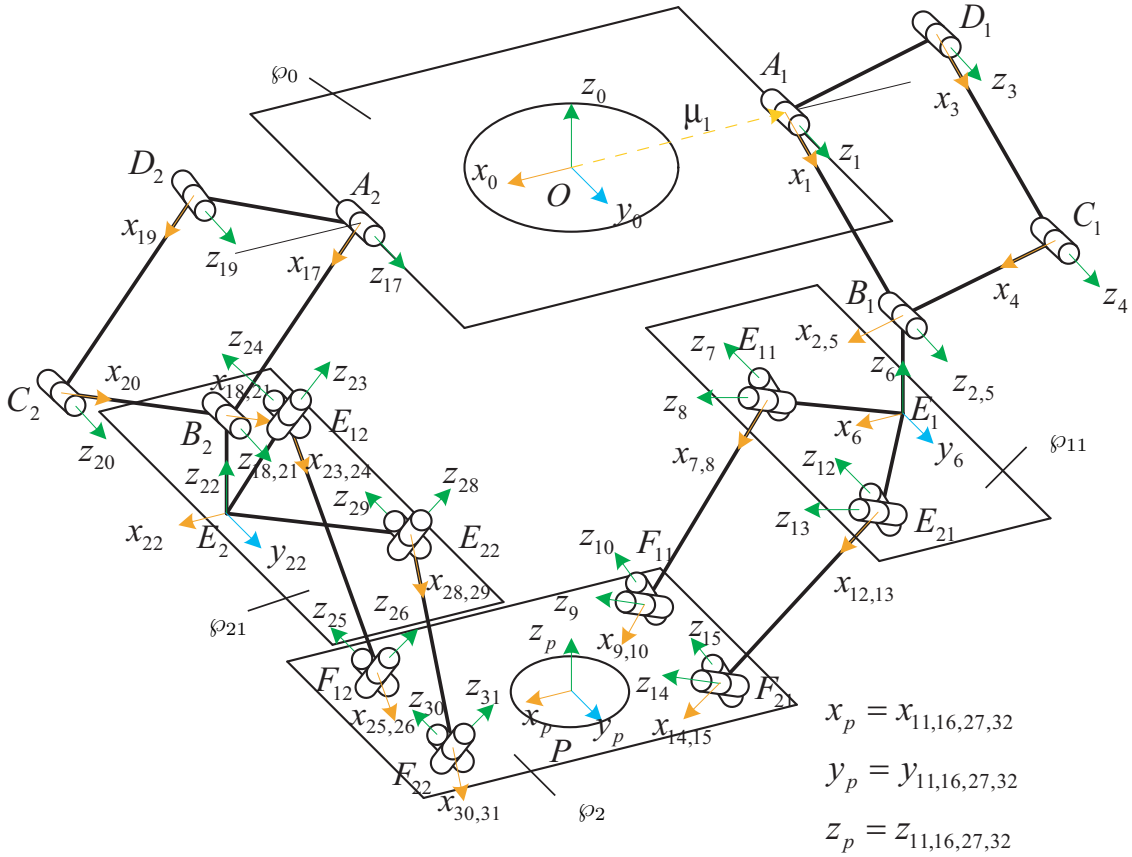


Figure II.7 – D-H Frames of the IRSBot-2 Robot

Table II.1 – Modified D-H Parameters for the IRSBot-2 Robot Leg 1

j	$a(j)$	μ_j	σ_j	γ_j	b_j	α_j	d_j	θ_j	r_j
1	0	1	0	π	0	$\frac{1}{2}\pi$	b	$-(\pi - q_1)$	0
2	1	0	0	0	0	0	l_1	$-q_1$	0
3	1	0	0	$\pi - q_1$	0	0	d	$-(\pi - q_1)$	0
4	3	0	0	0	0	0	l_1	$-q_1$	0
5	4	0	2	0	0	0	d	0	0
6	2	0	2	0	0	$\frac{1}{2}\pi$	0	0	$-e$
7	6	0	0	$-\frac{1}{2}\pi$	$-a_1 \sin\beta \cot\alpha_d$	$-\alpha_d$	$a_1 \cos\beta$	$\theta_{11,7} + \frac{1}{2}\pi$	$\frac{a_1 \sin\beta}{\sin\alpha_d}$
8	7	0	0	0	0	$\frac{1}{2}\pi$	0	$\theta_{11,8}$	0
9	8	0	0	0	0	0	l_2	$\theta_{11,9}$	0
10	9	0	0	0	0	$-\frac{1}{2}\pi$	0	$\theta_{11,10}$	0
11	10	0	2	$\frac{1}{2}\pi$	$\frac{p}{\sin\alpha_d}$	$-\alpha_d$	$a_2 \cos\beta$	$-\frac{1}{2}\pi$	$-p \cot\alpha_d$
12	6	0	0	$\frac{1}{2}\pi$	$-a_1 \sin\beta \cot\alpha_d$	α_d	$a_1 \cos\beta$	$\theta_{21,7} - \frac{1}{2}\pi$	$\frac{a_1 \sin\beta}{\sin\alpha_d}$
13	12	0	0	0	0	$\frac{1}{2}\pi$	0	$\theta_{21,8}$	0
14	13	0	0	0	0	0	l_2	$\theta_{21,9}$	0
15	14	0	0	0	0	$-\frac{1}{2}\pi$	0	$\theta_{21,10}$	0
16	15	0	2	$-\frac{1}{2}\pi$	$\frac{p}{\sin\alpha_d}$	α_d	$a_2 \cos\beta$	$\frac{1}{2}\pi$	$-p \cot\alpha_d$

Table II.2 – Modified D-H Parameters for the IRSBot-2 Robot Leg 2

j	$a(j)$	μ_j	σ_j	γ_j	b_j	α_j	d_j	θ_j	r_j
17	0	1	0	0	0	$-\frac{1}{2}\pi$	b	q_2	0
18	17	0	0	0	0	0	l_1	$\pi - q_2$	0
19	17	0	0	$-q_2$	0	0	d	q_2	0
20	19	0	0	0	0	0	l_1	$\pi - q_2$	0
21	20	0	2	0	0	0	d	0	0
22	18	0	2	0	0	$-\frac{1}{2}\pi$	0	π	$-e$
23	22	0	0	$-\frac{1}{2}\pi$	$-a_1 \sin\beta \cot\alpha_d$	α_d	$a_1 \cos\beta$	$\theta_{12,7} - \frac{1}{2}\pi$	$\frac{a_1 \sin\beta}{\sin\alpha_d}$
24	23	0	0	0	0	$-\frac{1}{2}\pi$	0	$\theta_{12,8}$	0
25	24	0	0	0	0	0	l_2	$\theta_{12,9}$	0
26	25	0	0	0	0	$\frac{1}{2}\pi$	0	$\theta_{12,10}$	0
27	26	0	2	$-\frac{1}{2}\pi$	$\frac{p}{\sin\alpha_d}$	α_d	$a_2 \cos\beta$	$-\frac{1}{2}\pi$	$-p \cot\alpha_d$
28	22	0	0	$\frac{1}{2}\pi$	$-a_1 \sin\beta \cot\alpha_d$	α_d	$a_1 \cos\beta$	$\theta_{22,7} + \frac{1}{2}\pi$	$\frac{a_1 \sin\beta}{\sin\alpha_d}$
29	28	0	0	0	0	$-\frac{1}{2}\pi$	0	$\theta_{22,8}$	0
30	29	0	0	0	0	0	l_2	$\theta_{22,9}$	0
31	30	0	0	0	0	$\frac{1}{2}\pi$	0	$\theta_{22,10}$	0
32	31	0	2	$\frac{1}{2}\pi$	$\frac{p}{\sin\alpha_d}$	$-\alpha_d$	$a_2 \cos\beta$	$\frac{1}{2}\pi$	$-p \cot\alpha_d$

II.4 Singularity Analysis

Based on the close loop equations of Eq. (II.3) and Eq. (II.4) in Section II.2, the velocity model can be obtained by differentiating the geometric model with respect to time:

$$\mathbf{A}\boldsymbol{t} + \mathbf{B}\dot{\boldsymbol{q}} = 0 \tag{II.36}$$

with,

$$\mathbf{B} = \begin{bmatrix} 2l_{2eq}l_1 \sin(q_1 - \psi_1) & 0 \\ 0 & 2l_{2eq}l_1 \sin(q_2 - \psi_2) \end{bmatrix} \tag{II.37}$$

$$\mathbf{A} = \begin{bmatrix} 2l_{2eq}\cos\psi_1 2l_2 eq \sin\psi_1 \\ 2l_{2eq}\cos\psi_2 2l_2 eq \sin\psi_2 \end{bmatrix} \tag{II.38}$$

where, matrix $\mathbf{B}$ and $\mathbf{A}$ are the Type 1 and Type 2 Jacobian matrix respectively, $\dot{\boldsymbol{q}} = \begin{bmatrix} \dot{q}_1 & \dot{q}_2 \end{bmatrix}^T$ is the joint velocity of the actuated joint, vector $\boldsymbol{t} = \begin{bmatrix} \dot{x} & \dot{z} \end{bmatrix}^T$ is the twist of the moving platform, ψ_i, to be recalled here, is the angle between $\boldsymbol{x}_0$ and the virtual link $H_{bi}H_{hi}$.

The main three types of singularities can be determined by the matrix $\mathbf{A}$ and $\mathbf{B}$:

1. if $\mathtt{det}(\mathbf{B}) = 0$, the robot loses one or more DOF and reaches a Type 1 singularity, which is also called serial singularity
2. if $\mathtt{det}(\mathbf{A}) = 0$, the robot gains one or more uncontrolled DOF and reaches a Type 2 singularity, which is also called parallel singularity
3. if $\mathtt{det}(\mathbf{A}) = \mathtt{det}(\mathbf{B}) = 0$, the robot reaches a Type 3 singularity. In the following, only the Type 1 and Type 2 singularity are analyzed, for the Type 3 singularity are obtained from the previous two types.

Similar to a $\underline{R}RRR\underline{R}$ mechanism, it is possible to show that the Type 1 singularities arise in the configurations where segments $[A_iB_i]$ and $[H_{bi}H_{hi}]$ are parallel (Fig. II.8), i.e. $q_i = \psi_i + k\pi$, with $k \in \mathbb{Z}$. Such configurations correspond to the boundaries of the Cartesian workspace.

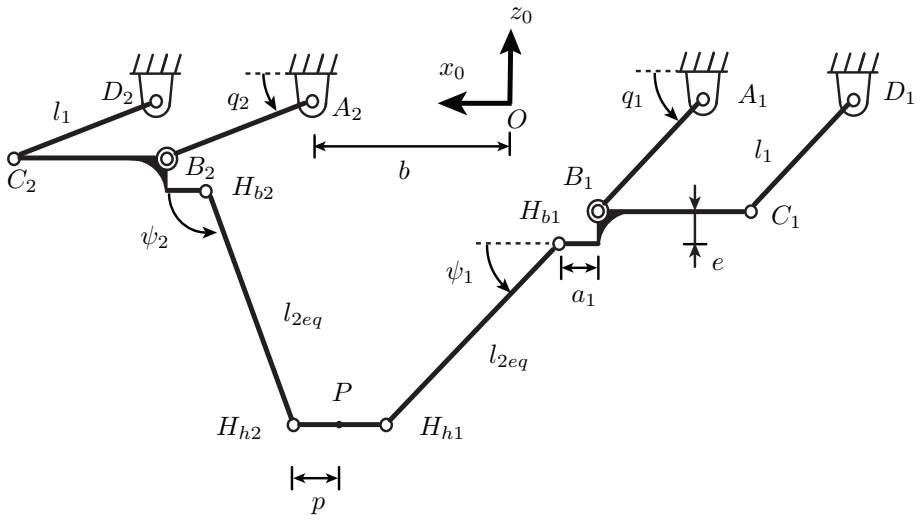


Figure II.8 – Type 1 Singularity of the IRSBot-2 Robot

The Type 2 singularities arise when segments $[H_{b1}H_{h1}]$ and $[H_{b2}H_{h2}]$ are parallel (Fig. II.9), i.e. $\psi_1 = \psi_2 + k\pi$, with $k \in \mathbb{Z}$. In such configurations, the displacement of the end effector along the normal to the distal legs and in the plane $(\boldsymbol{x}_0, O, \boldsymbol{z}_0)$ is no longer controlled.

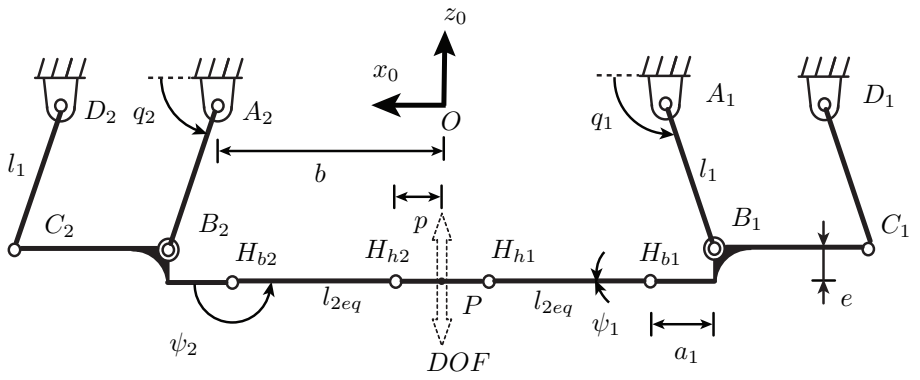


Figure II.9 – Type 2 Singularity of the IRSBot-2 Robot

III

Modeling of the Sensitivity Analysis for Robots

As discussed before, joint clearances and the variations in geometric parameters are the main sources for pose error of robots. For geometric parameters, errors can be reduced or eliminated by the identification process, whereas uncontrollable and unconstrained motions will be termed by joint clearances, and it is not so easy or convenient to eliminate their influences. But still joint clearances can not be avoided, as always some clearances in joints is made for easier assembly for mechanisms. In this chapter, the joint clearances and variations in geometric parameters are studied and modeled. And also the models for doing sensitivity analysis due to joint clearances and variations in geometric parameters are proposed here.

This chapter is divided into two sections. Section III.1 studies the models of the joint clearances, where error screws are proposed to represent the error pose termed by the joint clearances at each joints. Later, how the error pose at each joint transmitted to the pose error of the end-effector is studied. Finally, an optimization based method is proposed to find the maximum positional error and the maximum rotational error. Section III.2 presents the method for sensitivity analysis due to variations in geometric parameters, where a vector differentiate method has been adopted and recalled.

III.1 Sensitivity Analysis Due to Joint Clearances

III.1.1 Modeling the Clearances in an Axisymmetrical Joint

Joint clearance is one of the main sources of the pose error in robots. The joint clearances will allow the joint to have extra uncontrollable and unconstrained motions and can be transmitted to the end-effector by the kinematic chains. This cause the end-effector to move in an extra error space even if the active joints are locked. The extra motions due to the joint clearance can be both rotations and translations. Inituitively, the pose error at the local frame of the joint can be modeled as an error screw $\delta \boldsymbol{s}_{i,j}$

$$\delta \boldsymbol{s}_{i,j} = \begin{bmatrix} \delta \boldsymbol{r}_{i,j} \\ \delta \boldsymbol{t}_{i,j} \end{bmatrix} \tag{III.1}$$

Where, i is the index of the leg, j is the index of the joint in the i^{th} leg of the robot, $\delta \boldsymbol{r}_{i,j} = \begin{bmatrix} \delta r_{i,j,X} & \delta r_{i,j,Y} & \delta r_{i,j,Z} \end{bmatrix}^T$ stands for the orientational error, and $\delta \boldsymbol{t}_{i,j} =$

$\begin{bmatrix} \delta t_{i,j,X} & \delta t_{i,j,Y} & \delta t_{i,j,Z} \end{bmatrix}^T$ stands for the positional error in the local frame of the joint due to the joint clearances. Here the joint axis is considered along the z axis.

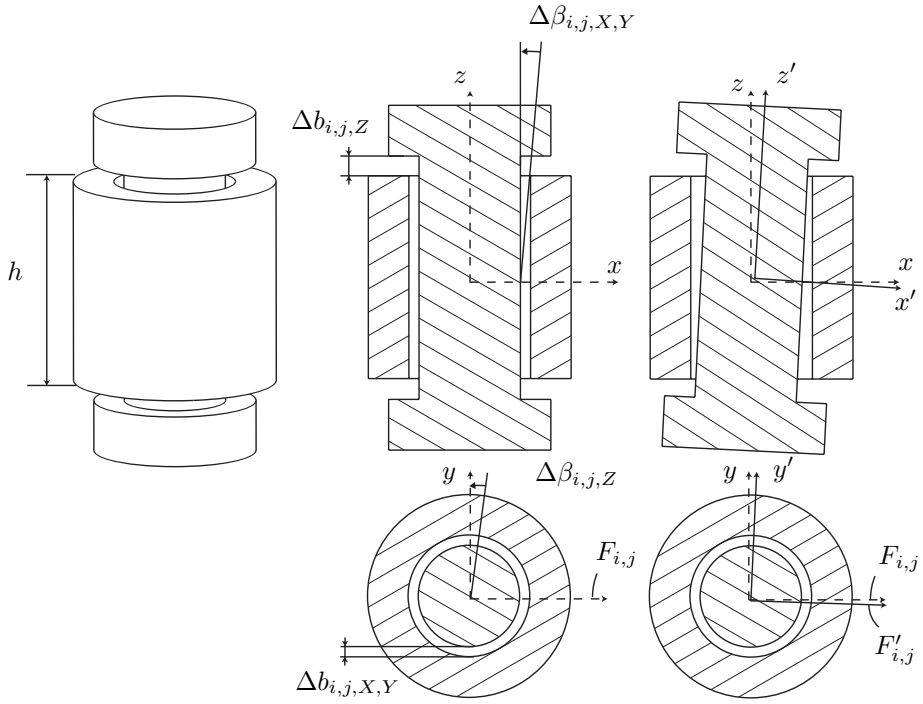


Figure III.1 – Model of the Joint Clearance

Fig. III.1 shows the joint clearance model, here the axisymmetric joint is studied. The translational clearance along the the joint axis is modeled as $\Delta b_{i,j,Z}$, and the translational clearance in the XY plane is modeled as $\Delta b_{i,j,X,Y}$. In addition, the rotational clearance about the joint axis is modeled as $\Delta\beta_{i,j,Z}$, and the rotational clearance about the axis that lies in the XY plane is modeled as $\Delta\beta_{i,j,X,Y}$. Certainly, $\Delta b_{i,j,Z}$ depends on the manufacturing geometric accuracy along the joint axis, and $\Delta b_{i,j,X,Y}$ depends on the manufacturing geometric accuracy along the radial directions. Normally, the value of $\Delta\beta_{i,j,X,Y}$ depends on the value of $\Delta b_{i,j,X,Y}$, by considering the length of the joint h. If the clearance $\Delta\beta_{i,j,X,Y}$ is small, then their relations can be approximately expressed as

$$\Delta\beta_{i,j,X,Y} = \frac{\Delta b_{i,j,X,Y}}{h} \tag{III.2}$$

The value of $\Delta\beta_{i,j,Z}$ depends on if the joint is active or passive. Basically for active joints, they are regarded as locked and the rotations about the joint axis should be assigned zero or a very small value. For passive joints, there will be some uncontrollable idle motions about the joint axis, and therefore $\Delta\beta_{i,j,Z}$ should be assigned a relatively larger value, so as to meet the requirement of idle motion.

Obviously, the error pose expressed in the local frame of the joint should be constrained as the following

$$\delta \boldsymbol{r}_{i,j,X}^2 + \delta \boldsymbol{r}_{i,j,Y}^2 \leqslant \Delta\beta_{i,j,X,Y}^2 \tag{III.3}$$

$$\delta \boldsymbol{r}_{i,j,Z}^2 \leqslant \Delta\beta_{i,j,Z}^2 \tag{III.4}$$

$$\delta \boldsymbol{t}_{i,j,X}^2 + \delta \boldsymbol{t}_{i,j,Y}^2 \leqslant \Delta b_{i,j,X,Y}^2 \tag{III.5}$$

$$\delta \boldsymbol{t}_{i,j,Z}^2 \leqslant \Delta b_{i,j,Z}^2 \tag{III.6}$$

III.1.2 Error Mapped to End-effector Pose

First we consider that there are no clearances in the joints, the pose of the end-effector can be deduced by simply multiplying the transformation matrix one by one in the kinematic chain. Here we recall the transformation matrix from [BCC10], which was $\mathbf{S}_{i,j}$ from Frame $F_{i,j}$ to Frame $F_{i,j+1}$,

$$\mathbf{S}_{i,j} = \mathbf{S}_{i,j,\alpha}\mathbf{S}_{i,j,d}\mathbf{S}_{i,j,\theta}\mathbf{S}_{i,j,r} \tag{III.7}$$

where,

$$\mathbf{S}_{i,j,\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_{i,j} & -\sin\alpha_{i,j} & 0 \\ 0 & \sin\alpha_{i,j} & \cos\alpha_{i,j} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{III.8}$$

$$\mathbf{S}_{i,j,d} = \begin{bmatrix} 1 & 0 & 0 & d_{i,j} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{III.9}$$

$$\mathbf{S}_{i,j,\theta} = \begin{bmatrix} \cos\theta_{i,j} & -\sin\theta_{i,j} & 0 & 0 \\ \sin\theta_{i,j} & \cos\theta_{i,j} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{III.10}$$

$$\mathbf{S}_{i,j,r} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r_{i,j} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{III.11}$$

So the pose of the moving-platform computed from Leg i will be

$$\mathbf{P}_i = \prod_{j=1}^{n_{i,f}} \mathbf{S}_{i,j} \tag{III.12}$$

where, $i = 1, 2, \ldots, m$. m is the total number of legs of the robot, $n_{i,f}$ represents the total number of frames of Leg i.

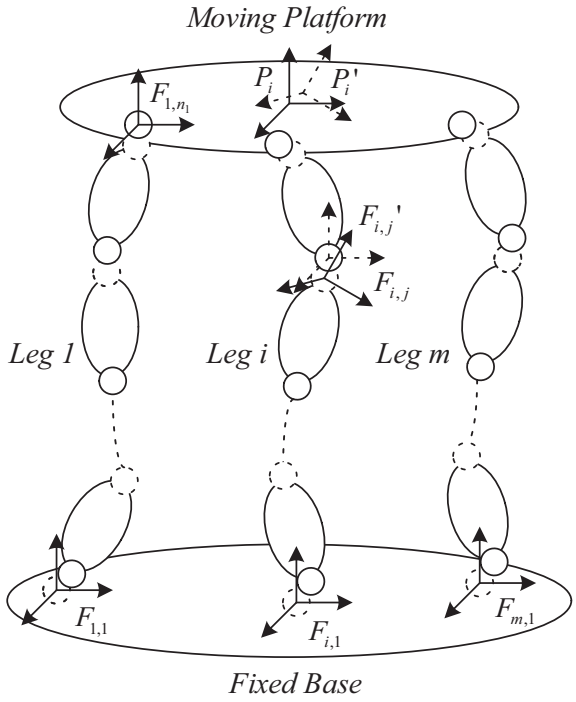


Figure III.2 – Model of the Robot with Joint Clearances

The moving-platform is connected to the base by the m legs, the pose computed from any leg should be equal to each other, as Eq. (III.13)

$$\mathbf{P} = \mathbf{P}_1 = \mathbf{P}_2 = \ldots = \mathbf{P}_m \tag{III.13}$$

However, when considering joint clearances in the kinematic chains, the real pose of the moving-platform $\mathbf{P}'$ won't be exactly equal to the theoretical pose $\mathbf{P}$. Because the error screws of the joint clearances modeled in Section III.1.1 will be transmitted to the end-effector pose through the kinematic chains. To find out how the error screw be mapped to the end-effector pose, let us recall the adjoin map of the transformation matrix $\mathbf{S}_{i,j}$

$$\mathtt{adj}(\mathbf{S}_{i,j}) = \begin{bmatrix} \mathbf{R}_{i,j} & \mathbf{O}_{3\times3} \\ \mathbf{T}_{i,j}\mathbf{R}_{i,j} & \mathbf{R}_{i,j} \end{bmatrix} \tag{III.14}$$

The translational vector $\boldsymbol{t}_{i,j}$ and the rotational matrix $\mathbf{R}_{i,j}$ can be extracted from the transformation matrix $\mathbf{S}_{i,j}$.$\mathbf{T}_{i,j}$ is the skew matrix of vector $\boldsymbol{t}_{i,j}$.

The adjoin map of screw expressed in Eq. (III.14) is to transfer screws at the local Frame $F_{i,j+1}$ to Frame $F_{i,j}$. The adjoin map of the inverse of the transformation matrix $\mathtt{adj}(\mathbf{S}_{i,j}^{-1})$ is to express screws at Frame $F_{i,j}$ to Frame $F_{i,j+1}$. As a result, the error screw $\delta\boldsymbol{s}_{i,j}$ expressed in the local frame of Joint j in Leg i can be expressed to the end-effector frame $F_{i,n_{i,f}}$ by multiplying all the inverse of the adjoin map from $n_{i,f}$ to $j+1$, as Eq. (III.15)

$$\left(\prod_{k=n_{i,f}}^{j+1} \mathtt{adj}(\mathbf{S}_{i,k}^{-1})\right)\delta\boldsymbol{s}_{i,j} \tag{III.15}$$

where, $n_{i,f}$ is the number of frames in Leg i.

Therefore, all error screws termed by joint clearances in one kinematic chain can be mapped to the end-efector pose, thus the pose error of the end-effector termed by all the joint clearances from all the joints in the kinematic chain is estimated as Eq. (III.16)

$$\delta \boldsymbol{p_i}_{|F_{i,P}} = \sum_{j=1}^{n_i} \prod_{k=n_{i,f}}^{j+1} \mathtt{adj}(\mathbf{S}_{i,k}^{-1}) \delta \boldsymbol{s}_{i,j} \tag{III.16}$$

Please note that n_i is the number of joints in Leg i, and $n_{i,f}$ is the number of frames in Leg i. A frame is attached to each link at the joint. Definitely, $n_{i,f} \geqslant n_i$. In the following expressions that contain n_i and $n_{i,f}$, they have the same meaning.

Since $\mathtt{adj}(\mathbf{S}_{i,k}^{-1})$ is equal to $(\mathtt{adj}(\mathbf{S}_{i,k}))^{-1}$, Eq. (III.16) can be rewritten in the following way

$$\delta \boldsymbol{p_i}_{|F_{i,P}} = \sum_{j=1}^{n_i} \prod_{k=n_{i,f}}^{j+1} (\mathtt{adj}(\mathbf{S}_{i,k}))^{-1} \delta \boldsymbol{s}_{i,j} \tag{III.17}$$

It is noteworthy that $\delta \boldsymbol{p}_i$ is expressed in the frame attached to the moving platform, that is Frame $F_{i,P}$. For the evaluation of the pose errors on the moving-platform, the small-displacement screw taking Frame $F_{i,P}$ to Frame $F'_{i,P}$ has to be expressed in the reference frame attached to the fixed base, that is Frame $F_{i,1}$

$$\delta \boldsymbol{p_i}_{|F_{i,1}} = \prod_{j=1}^{n_{i,f}} (\boldsymbol{N}_{i,j}) \delta \boldsymbol{p}_{i|P_{i,1}} \tag{III.18}$$

Where,

$$\mathbf{N}_{i,j} = \begin{bmatrix} \mathbf{R}_{i,j} & \mathbf{O}_{3\times 3} \\ \mathbf{O}_{3\times 3} & \mathbf{R}_{i,j} \end{bmatrix} \tag{III.19}$$

As a result,

$$\delta \boldsymbol{p}_{i|F_{i,1}} = \prod_{j=1}^{n_{i,f}} (\mathbf{N}_{i,j}) \sum_{j=1}^{n_i} \prod_{k=n_{i,f}}^{j+1} (\mathtt{adj}(\mathbf{S}_{i,k}))^{-1} \delta \boldsymbol{s}_{i,j} \tag{III.20}$$

$$= \sum_{j=1}^{n_i} \prod_{l=1}^{n_{i,f}} \mathbf{N}_{i,l} \prod_{k=n_{i,f}}^{j+1} (\mathtt{adj}(\mathbf{S}_{i,k}))^{-1} \delta \mathbf{s}_{i,j} \tag{III.21}$$

In a more compact way, the equation is rewritten as

$$\delta \boldsymbol{p} = \mathbf{M}_i \delta \boldsymbol{s}_i \tag{III.22}$$

Where,

$$\mathbf{M}_i \equiv \begin{bmatrix} \mathbf{M}_{i,1} & \mathbf{M}_{i,2} & \ldots & \mathbf{M}_{i,n_i} \end{bmatrix} \tag{III.23}$$

$$\mathbf{M}_{i,j} \equiv \prod_{l=1}^{n_{i,f}} (\mathbf{N}_{i,l}) \prod_{k=n_{i,f}}^{j+1} (\mathtt{adj}(\mathbf{S}_{i,k}))^{-1} \tag{III.24}$$

$$\delta \boldsymbol{s}_i \equiv \begin{bmatrix} \delta \boldsymbol{s}_{i,1}^T & \delta \boldsymbol{s}_{i,2}^T & \dots & \delta \boldsymbol{s}_{i,n_i}^T \end{bmatrix} \tag{III.25}$$

III.1.3 Maximum Pose Error for the Moving-Platform

In Section III.1.1, the joint clearances are modeled as error screws, and the screw is constrained by inequiaties (III.3)~(III.6). Based on the constraints, an optimization based method can be modeled to find the maximum positional error and the rotational error for the end-effector. Intuitively, due to the joint cleareaces, the end-efffector can reach some error space even when the actuated joints are locked. Inside the error space, it is possible to find the maximum positional and rotational errors for the end-effector by comparing all the reachable points.

Formally, the maximum positional error is denoted as $p_{\max}$, the maximum rotational error is denoted as $r_{\max}$. $p_{\max}$ can be obtained by solving the optimization problem

$$\begin{aligned} -p_{max}^2 \quad = \quad & \mathtt{minimize} - \sum_{k=4,5,6} (\boldsymbol{e}_{6,k}^T \delta \boldsymbol{p})^2, \\ \mathtt{over} \quad & \delta \boldsymbol{p}, \delta \boldsymbol{s}_{i,j}, j = 1, ..., n_i, i = 1, ..., m, \\ \mathtt{subject\ to} \quad & (\boldsymbol{e}_{6,1}^T \delta \boldsymbol{s}_{i,j})^2 + (\boldsymbol{e}_{6,2}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta\beta_{i,j,X,Y}^2 \leqslant 0, \\ & (\boldsymbol{e}_{6,3}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta\beta_{i,j,Z}^2 \leqslant 0 \\ & (\boldsymbol{e}_{6,4}^T \delta \boldsymbol{s}_{i,j})^2 + (\boldsymbol{e}_{6,5}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta b_{i,j,X,Y}^2 \leqslant 0, \\ & (\boldsymbol{e}_{6,6}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta b_{i,j,Z}^2 \leqslant 0, \\ & \delta \boldsymbol{p} = \mathbf{M}_i \delta \boldsymbol{s}_i, \\ & j = 1, ..., n_i, i = 1, ..., m \end{aligned} \tag{III.26}$$

where, $\boldsymbol{e}_{j,k} \in \mathbb{R}^6$ is defined such that $\mathbf{1} = \begin{bmatrix} \boldsymbol{e}_{j,1} & \boldsymbol{e}_{j,2} & \dots & \boldsymbol{e}_{j,j} \end{bmatrix}$

In the same way, r_{max} can be obtained by solving the optimization problem

$$\begin{aligned} -r_{max}^2 \quad = \quad & \mathtt{minimize} - \sum_{k=1,2,3} (\boldsymbol{e}_{6,k}^T \delta \boldsymbol{p})^2, \\ \mathtt{over} \quad & \delta \boldsymbol{p}, \delta \boldsymbol{s}_{i,j}, j = 1, ..., n_i, i = 1, ..., m, \\ \mathtt{subject\ to} \quad & (\boldsymbol{e}_{6,1}^T \delta \boldsymbol{s}_{i,j})^2 + (\boldsymbol{e}_{6,2}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta\beta_{i,j,X,Y}^2 \leqslant 0, \\ & (\boldsymbol{e}_{6,3}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta\beta_{i,j,Z}^2 \leqslant 0 \\ & (\boldsymbol{e}_{6,4}^T \delta \boldsymbol{s}_{i,j})^2 + (\boldsymbol{e}_{6,5}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta b_{i,j,X,Y}^2 \leqslant 0, \\ & (\boldsymbol{e}_{6,6}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta b_{i,j,Z}^2 \leqslant 0, \\ & \delta \boldsymbol{p} = \mathbf{M}_i \delta \boldsymbol{s}_i, \\ & j = 1, ..., n_i, i = 1, ..., m \end{aligned} \tag{III.27}$$

when the robot is a parallel robot,that is $m > 1$, the position and orientation of the moving platform are constrained by all the legs. The maximum pose error can be solved by the optimization problems (III.26) and (III.27). But it is noteworthy that when $m = 1$, the robot becomes serial robot, and in serial robots, the translational parts in error screw model for the joint clearance do not affect the rotational error of the end-effector. And there can be less constraints in the optimization problem, as follows

$$\begin{aligned} -r_{max}^2 \quad = \quad & \texttt{minimize} - \sum_{k=1,2,3} (\boldsymbol{e}_{6,k}^T \delta \boldsymbol{p})^2, \\ & \texttt{over} \quad \delta \boldsymbol{p}, \delta \boldsymbol{s}_{i,j}, j = 1, ..., n_i, i = 1 \\ & \texttt{subject to} \quad (\boldsymbol{e}_{6,1}^T \delta \boldsymbol{s}_{i,j})^2 + (\boldsymbol{e}_{6,2}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta\beta_{i,j,X,Y}^2 \leqslant 0, \\ & \qquad (\boldsymbol{e}_{6,3}^T \delta \boldsymbol{s}_{i,j})^2 - \Delta\beta_{i,j,Z}^2 \leqslant 0 \\ & \qquad \delta \boldsymbol{p} = \mathbf{M}_i \delta \boldsymbol{s}_i, \\ & \qquad j = 1, ..., n_i, i = 1, ..., m \end{aligned} \tag{III.28}$$

III.1.4 Computing Maximum Pose Errors with Inequality Constraints

However, In the optimization problems Eq. (III.26) , Eq. (III.27) and Eq. (III.28), there exist both inequality constraints and equality constraints. Usually, the equality constraints are more difficult to be satisfied when solving the optimization by computer programs. It is preferable to eliminate the equality constraints or adapt them into inequality constraints. Here, by doing QR factorization for the transpose of the error mapping matrix $\mathbf{M}_i^T$, it is possible to integrate the equality constraints into the inequality constraints.

$$\begin{aligned} \mathbf{M}_i^T &= \mathbf{V}_i \mathbf{U}_i \\ &= \begin{bmatrix} \mathbf{V}_{i,1} & \mathbf{V}_{i,2} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{i,1} \\ \mathbf{0}_{(6n_i-6)\times 6} \end{bmatrix} \\ &= \mathbf{V}_{i,1} \mathbf{U}_{i,1} \end{aligned} \tag{III.29}$$

(III.30)

where, $i = 1, ..., m$. $\mathbf{V}_i \in \mathbb{R}^{6n_i \times 6n_i}$ is orthogonal,with $\mathbf{V}_{i,1} \in \mathbb{R}^{6n_i \times 6}$, $\mathbf{V}_{i,2} \in \mathbb{R}^{6n_i \times (6n_i-6)}$. $\mathbf{U}_i \in \mathbb{R}^{6n_i \times 6}$, and $\mathbf{U}_{i,1} \in \mathbb{R}^{6\times 6}$ is an upper-triangular matrix. This QR factorization may be computed using the Householder method. Notice that, from its structure $\mathbf{M}_i$ is bound to bear its full row rank of six. As a result, $\mathbf{U}_{i,1}$ is also full rank, thus it is invertible. Moreover, we define

$$\delta \boldsymbol{q}_i \equiv \begin{bmatrix} \boldsymbol{q}_{i,1} \\ \boldsymbol{q}_{i,2} \end{bmatrix} = \mathbf{V}_i^T \delta \boldsymbol{s}_i \tag{III.31}$$

where $\boldsymbol{q}_{i,1} \in \mathbb{R}^6$ and $\boldsymbol{q}_{i,2} \in \mathbb{R}^{6n_i-6}$, so that

$$\delta \boldsymbol{s}_i = \mathbf{V}_{i,1} \delta \boldsymbol{q}_{i,1} + \mathbf{V}_{i,2} \delta \boldsymbol{q}_{i,2} \tag{III.32}$$

This allows us to rewrite the equality constraints as

$$\delta \boldsymbol{p} = \mathbf{M}_i \delta \boldsymbol{s}_i = \mathbf{U}_{i,1}^T \mathbf{V}_{i,1}^T \delta \boldsymbol{s}_i = \mathbf{U}_{i,1}^T \delta \boldsymbol{q}_{i,1} \tag{III.33}$$

or,

$$\delta \boldsymbol{q}_{i,1} = \mathbf{U}_{i,1}^{-T} \delta \boldsymbol{p} \tag{III.34}$$

where $\mathbf{U}_{i,1}^{-T}$ is the inverse of the transpose of $\mathbf{U}_{i,1}$. Upon substituting Eq. (III.32), we obtain

$$\delta \boldsymbol{s}_i = \mathbf{V}_{i,1} \mathbf{U}_{i,1}^{-T} \delta \boldsymbol{p} + \mathbf{V}_{i,2} \delta \boldsymbol{q}_{i,2} \tag{III.35}$$

Let us regroup all remaining optimization variables into the array

$$\delta \boldsymbol{u} \equiv \left[\begin{array}{cccc} \delta \boldsymbol{p}^T & \delta \boldsymbol{q}_{1,2}{}^T & \delta \boldsymbol{q}_{2,2}{}^T & \ldots \quad \delta \boldsymbol{q}_{m,2}{}^T \end{array} \right] \in \mathbb{R}^v \tag{III.36}$$

where $v = 6 + 6 \sum_{k=1}^{m} (n_k - 1)$, so that $\delta \boldsymbol{s}_i$ may be expressed as

$$\delta \boldsymbol{s}_i \equiv \boldsymbol{\Upsilon}_i \delta \boldsymbol{u} \tag{III.37}$$

where,
for $i = 1$

$$\boldsymbol{\Upsilon}_i \equiv \left[\begin{array}{ccc} \mathbf{V}_{i,1} \mathbf{U}_{i,1}^{-T} & \mathbf{V}_{i,2} & \mathbf{0}_{6n_i \times 6 \sum_{k=i+1}^{m} (n_k - 1)} \end{array} \right] \tag{III.38}$$

for $i = 2, 3, \ldots, m-1$

$$\boldsymbol{\Upsilon}_i \equiv \left[\begin{array}{cccc} \mathbf{V}_{i,1} \mathbf{U}_{i,1}^{-T} & \mathbf{0}_{6n_i \times 6 \sum_{k=1}^{i-1} (n_k - 1)} & \mathbf{V}_{i,2} & \mathbf{0}_{6n_i \times 6 \sum_{k=i+1}^{m} (n_k - 1)} \end{array} \right] \tag{III.39}$$

for $i = m$

$$\boldsymbol{\Upsilon}_i \equiv \left[\begin{array}{ccc} \mathbf{V}_{i,1} \mathbf{U}_{i,1}^{-T} & \mathbf{0}_{6n_i \times 6 \sum_{k=1}^{i-1} (n_k - 1)} & \mathbf{V}_{i,2} \end{array} \right] \tag{III.40}$$

Finally, substitute Eq. (III.35) into Eq. (III.26), we will be able to obtain an optimization problem that contains only inequality constraints, namely,

For maximum positional error,

$$\begin{array}{rcll} -p_{max}^2 & = & \texttt{minimize} & -\delta \boldsymbol{u}^T \mathbf{F}_0 \delta \boldsymbol{u}, \\ & & \texttt{over} & \delta \boldsymbol{u} \\ & & \texttt{subject to} & f_{i,j,k}(\delta \boldsymbol{u}) \equiv \delta \boldsymbol{u}^T \mathbf{F}_{i,j,k} \delta \boldsymbol{u} - 1 \leqslant 0 \\ & & & k = 1, \ldots, 4, j = 1, ..., n_i, \\ & & & i = 1, ..., m \end{array} \tag{III.41}$$

where,

$$\mathbf{F}_0 = \sum_{l=4,5,6} \boldsymbol{e}_{v,l}\boldsymbol{e}_{v,l}^T \tag{III.42}$$

For maximum rotational error,

$$\begin{aligned} -r_{max}^2 \quad &= \quad \texttt{minimize} - \delta\boldsymbol{u}^T\mathbf{F}_0\delta\boldsymbol{u}, \\ \texttt{over} \quad &\quad \delta\boldsymbol{u} \\ \texttt{subject to} \quad &\quad f_{i,j,k}(\delta\boldsymbol{u}) \equiv \delta\boldsymbol{u}^T\mathbf{F}_{i,j,k}\delta\boldsymbol{u} - 1 \leqslant 0 \\ &\quad k = 1,\ldots,4, j = 1,...,n_i, \\ &\quad i = 1,...,m \end{aligned} \tag{III.43}$$

where,

$$\mathbf{F}_0 = \sum_{l=1,2,3} \boldsymbol{e}_{v,l}\boldsymbol{e}_{v,l}^T \tag{III.44}$$

And for both of the two optimization problems,

$$\mathbf{F}_{i,j,1} = (1/\Delta\beta_{i,j,X,Y}^2)\boldsymbol{\Upsilon}_i^T(\boldsymbol{e}_{6n_i,6j-5}\boldsymbol{e}_{6n_i,6j-5}^T + \boldsymbol{e}_{6n_i,6j-4}\boldsymbol{e}_{6n_i,6j-4}^T)\boldsymbol{\Upsilon}_i \tag{III.45}$$

$$\mathbf{F}_{i,j,2} = (1/\Delta\beta_{i,j,Z}^2)\boldsymbol{\Upsilon}_i^T(\boldsymbol{e}_{6n_i,6j-3}\boldsymbol{e}_{6n_i,6j-3}^T)\boldsymbol{\Upsilon}_i \tag{III.46}$$

$$\mathbf{F}_{i,j,3} = (1/\Delta b_{i,j,X,Y}^2)\boldsymbol{\Upsilon}_i^T(\boldsymbol{e}_{6n_i,6j-2}\boldsymbol{e}_{6n_i,6j-2}^T + \boldsymbol{e}_{6n_i,6j-1}\boldsymbol{e}_{6n_i,6j-1}^T)\boldsymbol{\Upsilon}_i \tag{III.47}$$

$$\mathbf{F}_{i,j,4} = (1/\Delta b_{i,j,Z}^2)\boldsymbol{\Upsilon}_i^T\boldsymbol{e}_{6n_i,6j}\boldsymbol{e}_{6n_i,6j}^T\boldsymbol{\Upsilon}_i \tag{III.48}$$

(III.49)

III.2 Sensitivity Analysis Due to Variations in Geometric Parameters

The position and orientation of the end-effector of a robot are computed by the geometric model. The geometric parameters should be assigned to the geometric model, so as to do the computation, such as Eq. (III.7) . However, due to the manufacturing quality, there might be some errors/variations in the geometric parameters, the parameters assigned to the geometric model is not exactly the real ones. Thus, pose error of the end-effector can be termed.

Caro et al. [CWBC06] studied the sensitivity of a three-DOF translational parallel kinematic machine, and presented two sensitivity analysis method to geometric parameters: The linkage kinematic analysis method and the differential vector method. The linkage kinematic analysis method is proposed to have a rough idea of the length variations of the manipulator on the location of its end-effector. And the parallelogram in the leg is just regarded as a virtual link, where the variations of the parallelogram can not be taken into consideration. Besides, this method just take into consideration of the variations in link length and positional errors, where the variations in angular parameters can affect the pose error as well. The differential vector method, not only just take into consideration of the variations in the link lengths, but also the variations of the angular parameters.

Basically, the vector differential method is based on the theory of differentiating of a vector. Take the vector A_iB_i from paper [CWBC06] as an example

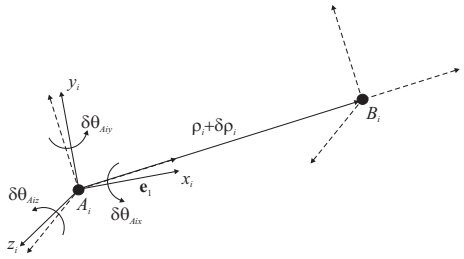


Figure III.3 – Variation in $A_i - B_i$ Chain

As is shown in Fig. III.3, the length of vector A_iB_i is ρ_i, and its unit vector is $\boldsymbol{e}_1$ expressed in the local Frame R_i. Assume $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ are the Cartesian coordinates of points A_i and B_i respectively, expressed in the base Frame R_b. If there is no variation in the vector A_iB_i, the vector can be expressed as

$$\boldsymbol{b}_i - \boldsymbol{a}_i = \mathbf{R}_i \rho_i \boldsymbol{e}_1 \tag{III.50}$$

where, $\mathbf{R}_i$ is the rotation matrix of Frame R_i to Frame R_b.

For the variations of vector A_iB_i, the local frame attached to point A_i might have some orientation errors, expressed as $\delta\theta_{Ai} = [\delta\theta_{Aix}, \delta\theta_{Aiy}, \delta\theta_{Aiz}]^T$, and the lenght might have a variation $\delta\rho_i$. When considering the variations, the vector will be expressed as

$$\boldsymbol{b}_i - \boldsymbol{a}_i = \mathbf{R}_i(\rho_i + \delta\rho_i)\boldsymbol{e}_1 + \mathbf{R}_i \delta\theta_{Ai} \times (\rho_i + \delta\rho_i)\boldsymbol{e}_1 \tag{III.51}$$

So the differentation of vector A_iB_i can be calculated by substracting Eq. (III.51) with Eq. (III.50), just leave the first order variations, here we obtain:

$$\delta(\boldsymbol{b}_i - \boldsymbol{a}_i) = \mathbf{R}_i \delta\rho_i \boldsymbol{e}_1 + \mathbf{R}_i \delta\theta_{Ai} \times \rho_i \boldsymbol{e}_1 \tag{III.52}$$

From Eq. (III.52), it can be seen that the differentiation of a vector, not just the variation in the length has to be taken into consideration, but also the change of orientation.

Obviously, a link A_iB_i can be modeled as vector A_iB_i. When considering the variations of the link, it should be in the same way as the variations of orientation and the link length. Thus, Eq. (III.52) can be adopted to model the variation of a link. And in the kinematic chain, all the links can be modeled in the same way. Therefore the pose error of the end-effector can be expressed by the variations of the geometric parameters in the kinematic chains. Section IV.5 in Chapter IV will show in details about how to apply the vector differentiation method to do sensitivity analysis due to variations in geometric parameters for close loop 5-bar linkage.

IV

Sensitivity Analysis of the 5-bar Linkage

Methods of sensitivity analysis due to joint clearances and geometric parameters have been discussed in Chapter III, and the methods are adopted in this chapter to analyze the sensitivity of the 5-bar linkages to its geometric parameters like link length and angular variations of the actuated joints, as well as its 5 joint clearances. In addition, it will be nice to verify the sensitivity analysis methods by studying the 5-bar linkage.

In this chapter, Section IV.1 studies the inverse geometric model of the 5-bar linkage. Section IV.2 studies the notations of the 5-bar linkage, and the modified Denavit-Hartenberg parameters are presented. And based on the geometric models, the sensitivity to joint clearances is analyzed in Section IV.3 studies the sensitivity analysis due to joing clearances of the 5-bar linkage based on the method presented in Section III.1. Section IV.4 compares of the closed-loop and open-loop robot, and introduces an actuation redundancy method to improve the accuracy of the closed-loop/parallel robot. Section IV.5 studies the sensitivity analysis due to variations in geometric parameters of the 5-bar linkage, and the sensitivity matrix is deduced.

IV.1 Inverse Geometric Model

5-bar linkage is a 2-DOF planar mechanism, which can perform 2-DOF positioning task in a plane. As is shown in Fig. IV.1, the 5-bar linkage is composed of 5 links: Link L_0, Link L_1, Link L_2 (attached with Link L_5), Link L_3, Link L_4. It can be seen from the figure that the close-loop mechanism is formed by points $O-A-C-E-D-B-O$. Point P is attached to the end-effector. The mechanism can be decomposed into two kinematic chains with a closure point at E, composing of kinematic chain $O-A-C-E-P$ and kinematic chain $O-B-D-E-P$.

The inverse geometry is to solve the joint variables based on knowing the position and orientation of the end-effector. Asume the position of Point P in Cartesian space is (X,Y), with Point A fixed in the base at (X_A,Y_A), the length of dash line AP can be computed

$$L_{AP}=\sqrt{(X-X_A)^2+(Y-Y_A)^2} \tag{IV.1}$$

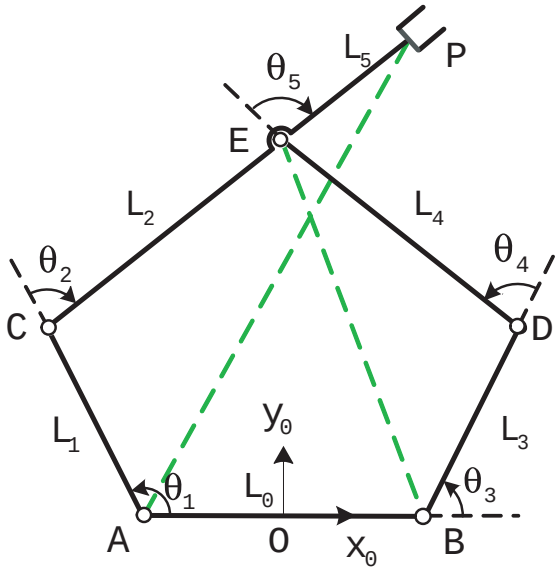


Figure IV.1 – Scheme of 5-bar Linkage for the Inverse Geometry

Based on the coordinates of the two ends of line AP, the orientation angle can be derived

$$\theta_{AP} = \arctan 2(Y - Y_A, X - X_A) \tag{IV.2}$$

Here, the function $\arctan 2$ is used in Eq. (IV.2), it can generate the angular value in terms of the pose of the line. Normally, the function arctan can only give the angular value between $-\frac{\pi}{2}$ and $\frac{\pi}{2}$. But the line can be in 4 different quadrants, in matlab, there is a function 'atan2', which can give the angle values corresponding to different quadrants.

As is shown in Fig. IV.1, line AP can be regarded as one edge of $\triangle ACP$, the length of AC is L_1 and the length of CF is $L_2 + L_5$, based on the Law of Cosine, the angle between line AP and line AC, denoted as θ_A can be formulated

$$\theta_A = \arccos \frac{L_1^2 + L_{AP}^2 - (L_2 + L_5)^2}{2L_1 L_{AP}} \tag{IV.3}$$

In the same way, the angle between line CP and line AC, denoted as θ_C can be formulated

$$\theta_C = \arccos \frac{L_1^2 + (L_2 + L_5)^2 - L_{AP}^2}{2L_1(L_2 + L_5)} \tag{IV.4}$$

Since the kinematic chain $A - C - P$ is a $2R$ SCARA robot, to reach the same position for the end-effector, there are two configurations: elbow up and elbow down.

Solution for elbow-up configuration

$$\theta_1 = \theta_{AP} + \theta_A, \tag{IV.5}$$

$$\theta_2 = -(\pi - \theta_C) \tag{IV.6}$$

Solution for elbow-down configuration

$$\theta_1 = \theta_{AP} - \theta_A, \tag{IV.7}$$
$$\theta_2 = \pi - \theta_C \tag{IV.8}$$

For the kinematic chain $O - B - D - E - P$, the position of Point E can be calculated

$$X_E = X - L_5 cos(\theta_1 + \theta_2), \tag{IV.9}$$
$$Y_E = Y - L_5 sin(\theta_1 + \theta_2) \tag{IV.10}$$

Asume the position of Point B is (X_B, Y_B), the length of line BE can be formulated

$$L_{BE} = \sqrt{(X_E - X_B)^2 + (Y_E - Y_B)^2} \tag{IV.11}$$

And the orientation angle of line BE can also be solved by using the arctan 2 function

$$\theta_{BE} = \arctan 2(Y_E - Y_B, X_E - X_B) \tag{IV.12}$$

In the same way, line BE can also be regarded as the edge of $\triangle BDE$. Based on the lengths of the triangle lengths, the inner angles of $\triangle BDE$ can also be solved according to the Law of Cosine

$$\theta_B = \arccos \frac{L_{BE}^2 + L_3^2 - L_4^2}{2 L_{BE} L_3} \tag{IV.13}$$

$$\theta_D = \arccos \frac{L_3^2 + L_4^2 - L_{BE}^2}{2 L_3 L_4} \tag{IV.14}$$

The $2R$ chain $B - D - E$ also has two configurations to reach the same position of Poit E: elbow-up and elbow-down

Solution of elbow-up Configuration

$$\theta_3 = \theta_{BE} + \theta_B, \tag{IV.15}$$
$$\theta_4 = -(\pi - \theta_D) \tag{IV.16}$$

Solution of elbow-down Configuration

$$\theta_3 = \theta_{BE} - \theta_B, \tag{IV.17}$$
$$\theta_4 = \pi - \theta_D \tag{IV.18}$$

Therefore, totally there will be four solutions for the inverse kinematics of the 5-bar linkage. Here we choose the combination of elbow-up configuration for the kinematic chain $O - A - C - P$ and the elbow-down configuration for the kinematic chain $O - B - D - E$

When close the loop at Point E, the value of θ_5 can be calculated

$$\theta_5 = (\theta_1 + \theta_2) - (\theta_3 + \theta_4) \tag{IV.19}$$

For verify the inverse geometric model based on the four solutions, the 5-bar linkage is parameterized as: $L_0 = 100mm, L_1 = L_2 = L_3 = L_4 = 50mm, L_5 = 0mm$. Choose a position of the end-effector is at the point $(0, 70)$, the 5-bar linkage can reach the same point with 4 working modes in terms of the inverse geometric model, as is shown in Fig. IV.2.

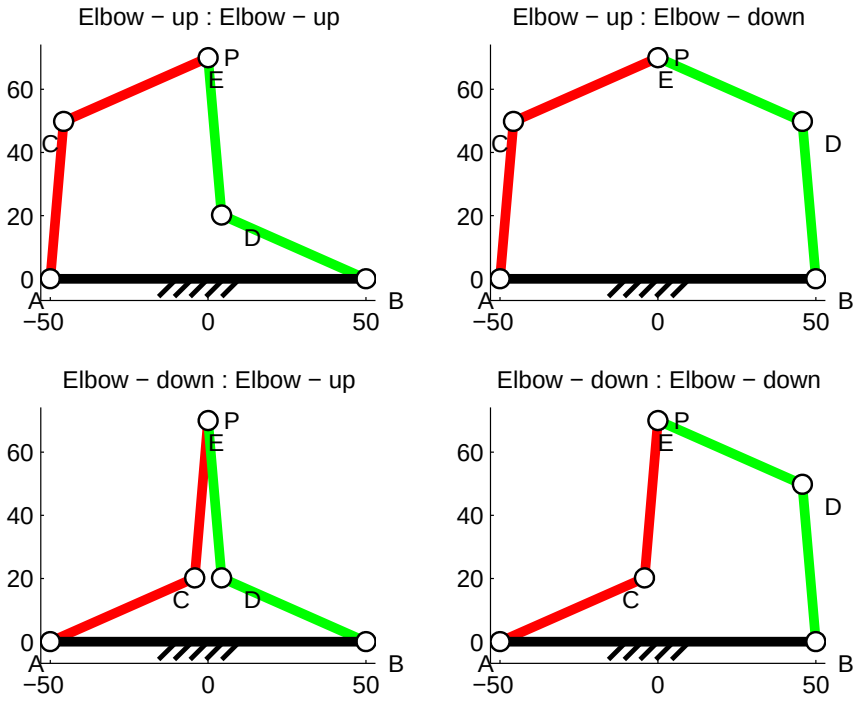


Figure IV.2 – Configurations of the Four Working Modes

IV.2 D-H Parameters of the 5-bar Linkage

With given angles in the kinematic chains, the pose of the end-effector can be derived. Here, Denavit-Hartenberg method [DH55] is applied. Since the 5-bar linkage is a closed-loop robot, the new kinematic notations [Kha86] are employed. As is shown in Fig. IV.3, different frames are defined for the 5-bar linkage. The base frame, which is attached to Link $L0$ is F_0 with its origin at Point O, Frame F_1 is attached to Link L_1 with its origin at Point A, Frame F_2 is attached to Link L_2 with its origin at Point C, Frame F_3 is attached to Link L_3 with its origin at Point B, Frame F_4 is attached to Link L_4 with its origin at Point D. At Point E, there are two frames attached to Link $L5$: Frame F_5 and Frame F_6. But the antecedent of Frame F_5 is Frame F_4, the antecedent of Frame F_6 is Frame F_2. The frame that is attached to the end-effector is Frame F_7.

Note that the 5-bar linkage is a close-loop mechanism, we open the loop at Point E and two frames are attached to the same link but with different antecedents. To do the

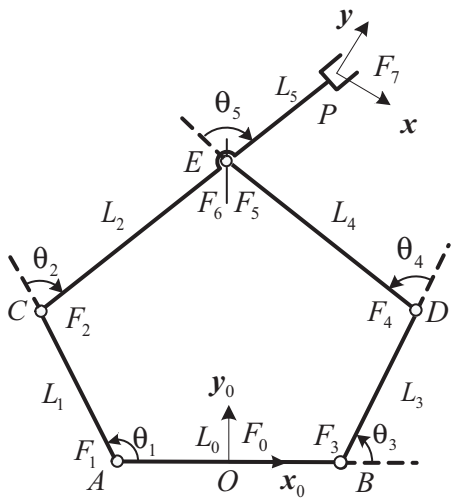


Figure IV.3 – Frames Attached to the 5-bar Linkage

direct geometry, the modified D-H parameters [Kha86] are applied here. Tab. IV.1 shows the modified D-H parameters

Table IV.1 – Modified D-H Parameters for the 5-bar Linkage Mechanism

j	$a(j)$	μ_j	σ_j	γ_j	b_j	α_j	d_j	θ_j	r_j
1	0	1	0	0	0	0	$-\frac{1}{2}L_0$	θ_1	0
2	1	0	0	0	0	0	L_1	θ_2	0
3	0	1	0	0	0	0	$\frac{1}{2}L_0$	θ_3	0
4	3	0	0	0	0	0	L_3	θ_4	0
5	4	0	0	0	0	0	L_4	θ_5	0
6	2	0	2	0	0	0	L_2	0	0
7	6	0	2	0	0	0	L_5	0	0

Based on the D-H parameters, the transformation matrix between one frame to another can be calculated by Eq. (III.7), and the transformation matrix from the base to the end-effector can be formulated from the kinematic chains: $O-A-C-E-P$ and $O-B-D-E-P$. Here the kinematic chain $O-A-C-E-P$ is denoted as Leg 1, the kinematic chain $O-B-D-E-P$ is denoted as Leg 2, so the transformation matrix from the base to the end-effector can be computed in the two legs separately

$$^0\mathbf{S}_7 = \mathbf{S}_{1,1}\mathbf{S}_{1,2}\mathbf{S}_{1,6}\mathbf{S}_{1,7} \tag{IV.20}$$

$$^0\mathbf{S}_7 = \mathbf{S}_{2,3}\mathbf{S}_{2,4}\mathbf{S}_{2,5}\mathbf{S}_{2,6}\mathbf{S}_{2,7} \tag{IV.21}$$

IV.3 Sensitivity Analysis Due to Joint Clearances

Totally, 5-bar linkage has 5 joints, with 2 active joints and 3 passive joints. Here we choose the two joints at the base to be actuated. As is shown in Fig. IV.3. The joint clearance at each joint can be modeled as an error screw in terms of Section III.1.1, and the error screw

can be mapped to the pose error of the end-effector throughout the kinematic chains in terms of Section III.1.2.

Here, the close-loop 5-bar linkage is decomposed into 2 legs: Leg 1 ($O-A-C-F$) and Leg 2 ($O-B-D-E-F$). The error screws due to joint clearances $\delta\boldsymbol{s}_{1,1}$ and $\delta\boldsymbol{s}_{1,2}$ can be mapped to the pose error by Leg 1, and the error screws due to joint clearances $\delta\boldsymbol{s}_{2,3}, \delta\boldsymbol{s}_{2,4}$ and $\delta\boldsymbol{s}_{2,5}$ can be mapped to the pose error by Leg 2. Thus the total pose error due to the joint clearances can be calculated by summing the pose errors mapped from the two legs.

As the sensitivity analysis due to the joint clearance model is based on transformation matrix, the transformation matrix of $\mathbf{S}_{i,j}$ (i is the number of leg, j is the number of frame) should first be computed in terms of the modified D-H parameters (Tab. IV.1) for the two legs. Then the rotation matrix $\mathbf{R}_{i,j}$ and the translational vector $\boldsymbol{t}_{i,j}$ of each transformation matrix can be extracted from the transformation matrix, and consequently the adjoint map $\mathtt{adj}(\mathbf{S}_{i,j})$ can be obtained. Finally, the error mapping matrix $\mathbf{M}_i$ can be constructed

For Leg 1

$$\mathbf{M}_1 = \begin{bmatrix} \mathbf{M}_{1,1} & \mathbf{M}_{1,2} \end{bmatrix} \tag{IV.22}$$

For Leg 2

$$\mathbf{M}_2 = \begin{bmatrix} \mathbf{M}_{2,3} & \mathbf{M}_{2,4} & \mathbf{M}_{2,5} \end{bmatrix} \tag{IV.23}$$

Therefore, the pose error of the end-effector can be approximately estimated

$$\delta\boldsymbol{p} = \mathbf{M}_1\delta\boldsymbol{s}_1 \tag{IV.24}$$

$$\delta\boldsymbol{p} = \mathbf{M}_2\delta\boldsymbol{s}_2 \tag{IV.25}$$

where, $\delta\boldsymbol{s}_1 = \begin{bmatrix} \delta\boldsymbol{s}_{1,1} \\ \delta\boldsymbol{s}_{1,2} \end{bmatrix}$ and $\delta\boldsymbol{s}_2 = \begin{bmatrix} \delta\boldsymbol{s}_{2,3} \\ \delta\boldsymbol{s}_{2,4} \\ \delta\boldsymbol{s}_{2,5} \end{bmatrix}$.

Obviously, Eq. (IV.24) and Eq. (IV.25) are very direct for predicting the pose error of the end-effector from the two legs, if the error screw is known in hand. But there are some constraints to be imposed, for example, the translational part and rotational part of the error screw should be bounded by the joint clearances. Moreover pose error computed from the two legs should be equal to each other. In this case, it is possible to find the maximum pose error by the optimization problems with inequality constraints and equality constraints. Consequently, with the QR factorization method, equality constraints can be integrated into the inequality constraints.

The optimization based methods (III.1.4) and (III.43) for find the maximum positional error and rotational error are adopted here, with $m = 2, n_1 = 2, n_2 = 3$ and the number decision variables is

$$v = 6 + 6 \times [(2-1) + (3-1)] = 24 \tag{IV.26}$$

For the sensitivity analysis, the second working mode of Fig. IV.2 is selected. And a symmetric structure is selected, thus the robot is parameterized as Tab. IV.2

Table IV.2 – Parameterization of the 5-bar Linkage

$L_0 : [mm]$	$L_1 : [mm]$	$L_2 : [mm]$	$L_3 : [mm]$	$L_4 : [mm]$	$L_5 : [mm]$
100	50	50	50	50	0

Here we also choose a work space that contains parallel singularity, so as to see how the Type 2 singularity affects the sensitivity due to the joint clearances. The joint clearances are denoted to the corresponding joints, as is shown in Fig. IV.4. And a rectangular Cartesian Space is selected, with $x_{min} = -20mm, y_{min} = 30mm, x_{max} = 20mm, y_{max} = 70mm$. The Cartesian Space is meshed into 41×41 grids, the sensitivity analysis is done for each point.

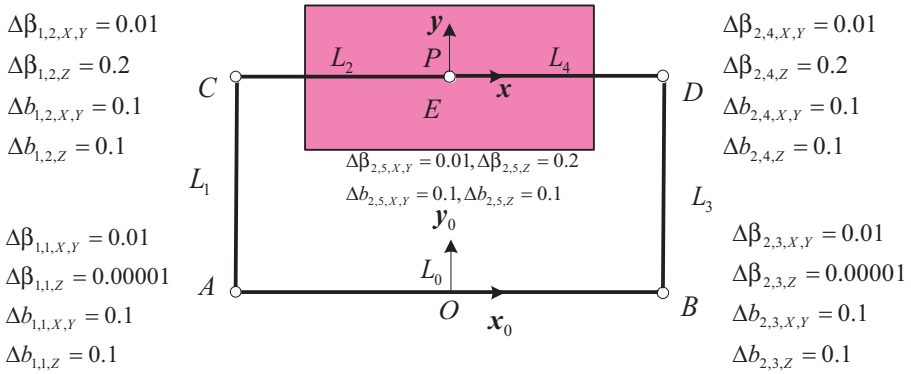


Figure IV.4 – Case Study for a Symmetric 5-bar Linkage

In Fig. IV.4, Joint A and Joint B are actuated joints, normally they are regarded as locked, here assign very small rotation around the z-axis, but for Joint C, Joint D and Joint E, they are all passive joints, due to the joint clearances, even though the active joints are locked, the passive joints will still have some idle motions, that is some relative larger rotation around its rotation axis. To be sure that the idle motions will not be ignored, larger bounds are assigned to $\Delta\beta_{i,j,Z}$. Here we assign $\Delta\beta_{i,j,Z} = 0.2rad$ for the passive joints, of course a rotation of $0.2rad$ is not a small motion, but due to the mutual constraints of the two legs in the close loop mechanism, the actual idle motion won't be that large.

The maximum positional error and rotational error can be computed by solving the optimization problems with objective functions as Eq. (III.1.4) and Eq. (III.43). Here, the optimization process is carried out by using $fmincon$ function which can only give local minimum. But we need a global minimum, since the parallel computation method is used here. Here we choose 200 start points, each point can reach a local minimum, and by overlaying local minimums, we can approximately reach a global minimum. Of course, the more start points we choose, the more global minimum we will get. But regarding to the computation time, we can not choose infinit number of start points. After several tests, we found that 200 start points are enough for the global minimum. To do this optimization problem is very time consuming. But fortunately, we still can get the results after running the program for several days on a workstation.

In Fig. IV.5, (a) shows the perspective view, and (b) shows the top view of the surface

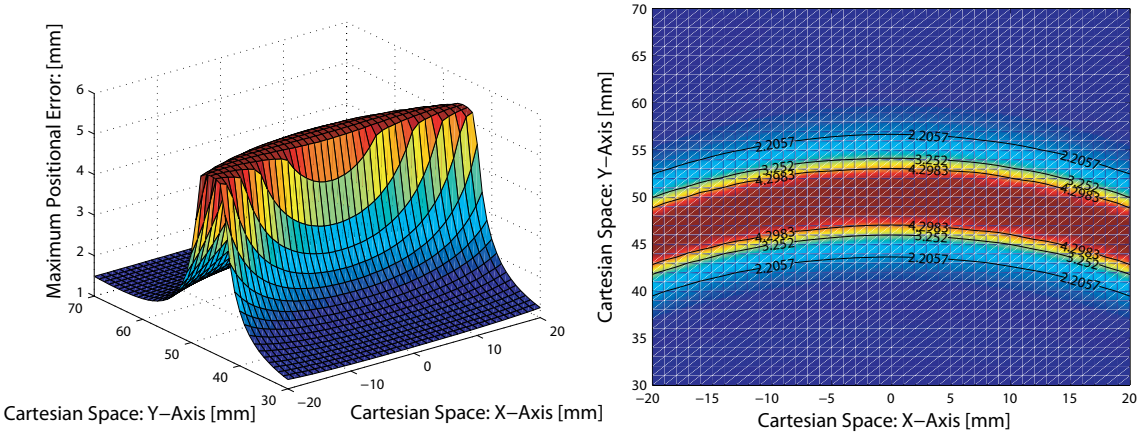


Figure IV.5 – Maximum Positional Error of the 5-bar Linkage Throughout A Cartesian Space

of the maximum positional error throughout the Cartesian Space. Obviously, the maximum positional error is symmetric about the y-axis. Moreover, throughout the cartesian space, the maximum positional error varies but the surface keeps continuous, and at the vicinity of parallel singularity it grow

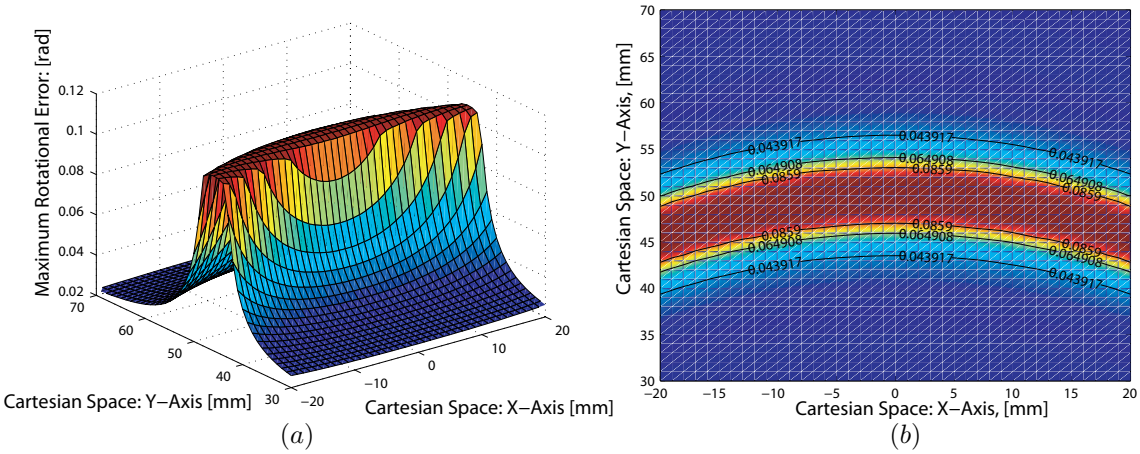


Figure IV.6 – Maximum Rotational Error of the 5-bar Linkage Throughout A Cartesian Space

From the two figures Fig. IV.5 and Fig. IV.6, it can be seen that at the vincinity of parallel singularity, both the positional error and the rotational error are quite large. Because the idle motions of the passive joints are quite large at that area. Moreover, the closer the distance between the position and the parallel singularity, the larger the positional and rotational errors. And the maximum errors are exactly at the singular configurations. In this case, when design the close-loop robot, it is better to avoid the parallel singularity, and make the workspace of the robot far from the parallel singularities.

Intuitively, $\delta \boldsymbol{p} = \delta \boldsymbol{\theta} \times \boldsymbol{L}$, if the link length L is quite long, even small variations in angular values will cause large errors in the positional errors. And by checking the optimum error screw of at each joint, it can be seen that the idle motion of the passive joint is relatively larger, and the idle motion is the main source of the pose error. Thus, if the length of the link that is connected to the passive joint is smaller, the error of the idle motion won't be

amplified that much, and the positional error can be smaller.

IV.4 Comparison Between Closed-loop and the Open-loop Mechanisms

The sensitivity analysis due to joint clearances has been done in Section IV.3. From the results shown in Fig. IV.5 and Fig. IV.6, it can be seen that when the 5-bar linkage meets parallel singularity or in the vicinity of parallel singularity, both the positional error and the rotational error gets quite large. And the accuracy of the closed-loop 5-bar linkage seems not so high.

Note that the 5-bar linkage is composed of two kinematic chains, each of which can perform the positioning task in 2D plane along, it will be quite interesting to study the sensitivities of the two kinematic chains separately. Since the parallel singularity is just for parallel robots, the serial kinematic chains won't have any parallel singularities. By opening the loop at Point E, the 5-bar linkage can be decomposed into two serial robots: $O-A-B-P$ and $O-C-D-E-P$, as is shown in Fig. IV.7 and Fig. IV.8. Leg 1 can be regarded as a 2R Scara robot, and Leg 2 can be regarded as a 3R robot.

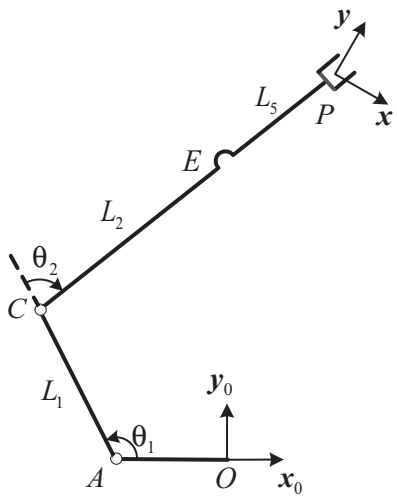


Figure IV.7 – Serial Kinematic Chain Leg 1

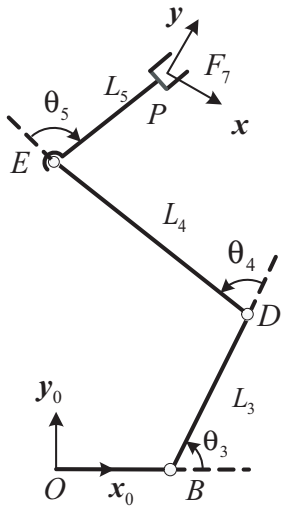
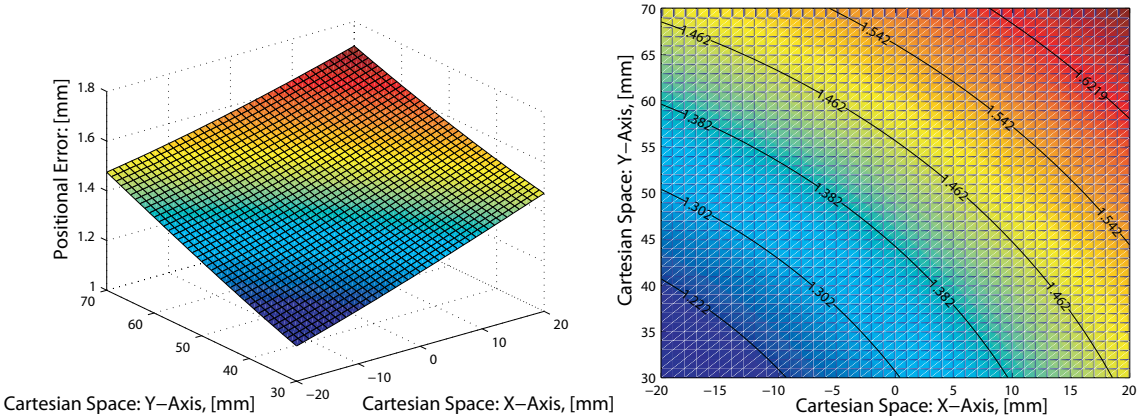


Figure IV.8 – Serial Kinematic Chain Leg 2

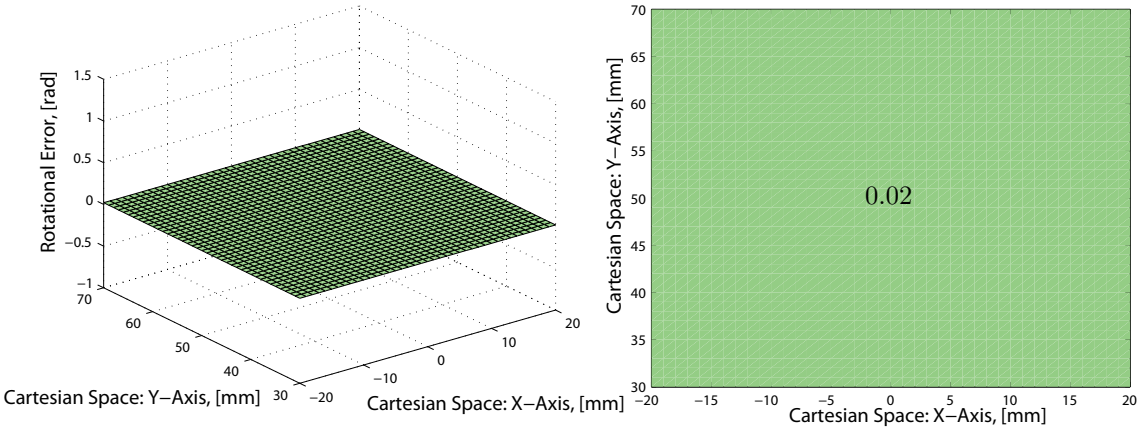
When using the serial robots to perform the positioning task, certainly there won't be any passive joints in serial robots, and all the joints in the kinematic chain of the serial robot should be actuated. When analyzing the sensitivity to the joint clearances for serial robots, all the joints are regarded as locked, that is very small values are assigned to $\Delta\beta_{i,j,Z}$. With the same dimensions as is shown in Tab. IV.2, and the same workspace defined in Fig. IV.4, optimization problems are solved separately for each leg to find the maximum positional error and rotational error.

Fig. IV.9 shows the maximum positional error throughout the Cartesian space for the serial kinematic chain Leg 1, and Fig. IV.10 shows its maximum rotational error throughout the Cartesian space. For the positional error, it seems the further the point in the Cartesian



(a) (b)

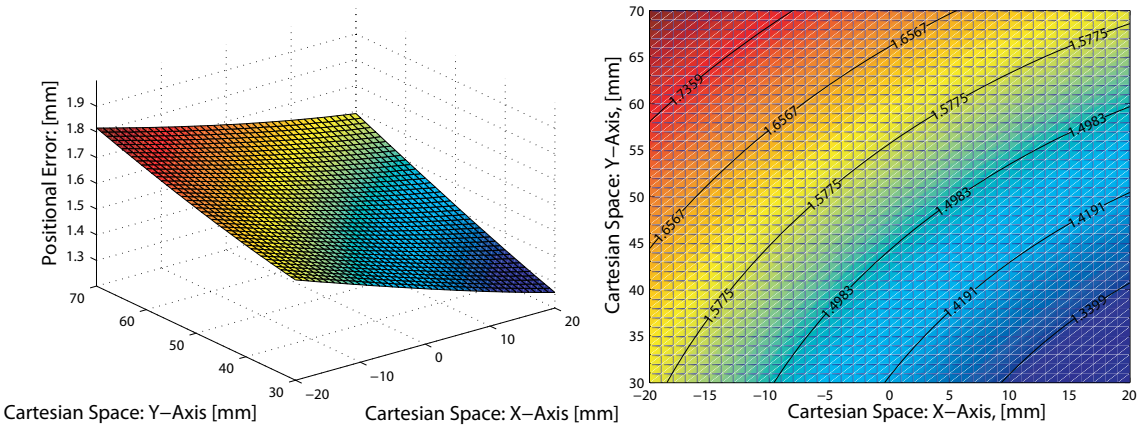
Figure IV.9 – Positional Error of Leg 1



(a) (b)

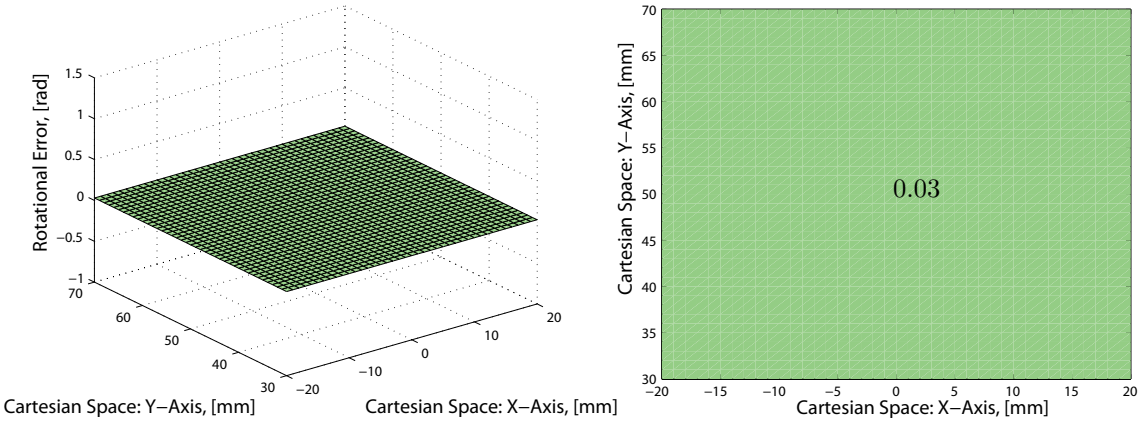
Figure IV.10 – Rotational Error of Leg 1

space from the origin of Leg 1, the larger the positional error. But the rotational error, as is already presented in Chapter III, is constant throughout the Cartesian Space, and is equal to equal to $0.02rad$. Note that, $\Delta\beta_{i,j,X,Y} = 0.01$, and $\Delta\beta_{i,j,Z}$ is set to a very small value that is close to zero. The maximum rotational error from Leg 1 is equal to the sum of the maximum rotational errors the error screws can reach at the joints.



(a) (b)

Figure IV.11 – Positional Error of Leg 2



(a) (b)

Figure IV.12 – Rotational Error of Leg 2

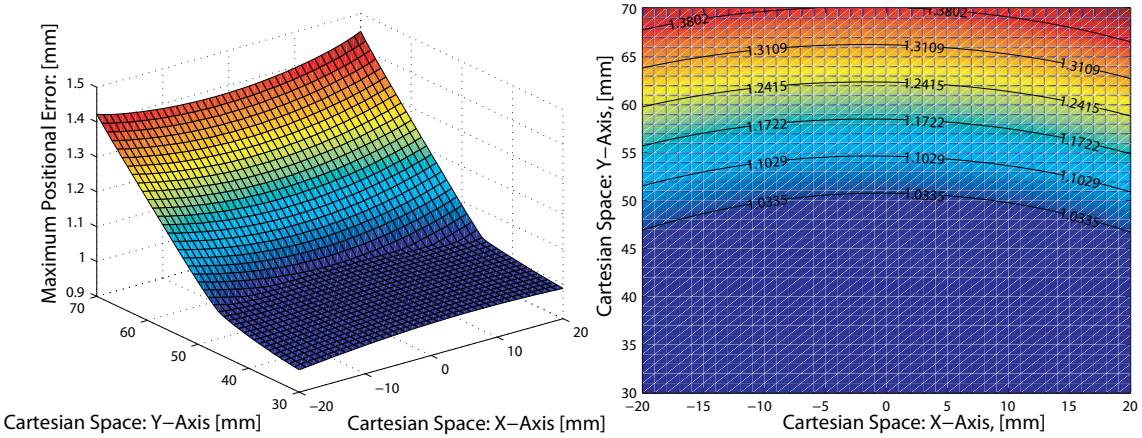
For the sensitivity analysis due to joint clearances of Leg 2, Fig. IV.11 shows the maximum positional error throughout the Cartesian space, and Fig. IV.12 shows the maximum rotational errors throughout the Cartesian space. Similar to the analysis results of Leg 1, the maximum positional error gets larger if the point is further from the origin of Leg 2, and the maximum rotational error keeps constant throughout the Cartesian space. But the maximum rotational error is equal to $0.03rad$, that is because there are 3 revolute joints in Leg 2, and the maximum rotational error that the error screw at the joint can reach is $0.01rad$.

By comparing the maximum positional errors and rotational errors, it is obvious that if we use serial legs work independantly, better accuracy can be obtained. That is because in parallel robots, there are some passive joints, and there will be some idle motions even

when the actuated joints are locked if there are some clearances in joints. Moreover, the parallel robots will have parallel singularities, and parallel singularity greatly increase the pose error of the end-effector.

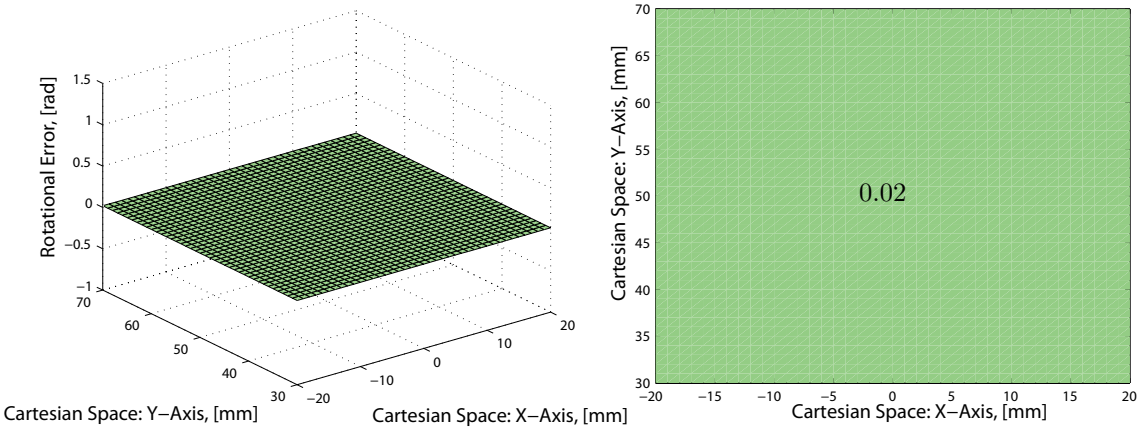
Thus, to improve the accuracy of parallel robots, the idle motions and parallel singularities should be avoided. One way to make this is to use actuation redundancy, that is to actuate even the passive joints.

For the 5-bar linkage, normally just the two joints at the base are actuated, and the other 3 joints are passive joints. To use the actuation reduncy to reduce the idle motions of the passive joints, here we choose to actuate Joint θ_2 and Joint θ_4. With this, when doing the sensitivity analysis, the two joints are also regarded as locked and very small values are assigned to $\Delta\beta_{i,j,Z}$. And then we get the following results, as is shown in Fig. IV.13 for the maximum positional error and in Fig. IV.14 for the maximum rotational error.



(a) (b)

Figure IV.13 – Positional Error with Passive Joints Actuated



(a) (b)

Figure IV.14 – Rotational Error with Passive Joints Actuated

The maximum rotational error shown in Fig. IV.14, is constantly equal to $0.02rad$, because the it is dominated by Leg 1 of the 5-bar linkage. Leg 1 can only reach maximum rotational error of $0.02rad$. By comparing the results calculated with actuate redundancy

in Fig. IV.13 and Fig. IV.14, with the results calculated without actuation redundancy shown in Fig. IV.5 and Fig. IV.6, as well as the results calculated separately for the legs in Fig. IV.9, Fig. IV.10, Fig. IV.11 and Fig. IV.12. Obviously, the maximum positional error and rotational error throughout the Cartesian of the 5-bar linkage with actuation redundancy is the smallest. Thus, the accuray of the parallel robots can be improved by using actuation redundancy.

IV.5 Sensitivity Analysis Due to Variations in Geometric Parameters

Besides joint clearances, variations in geometric parameters can also affect the pose error of the end-effector. Here is to study the influences of the variations in geometric parameters for 5-bar linkage. A sensitivity matrix that maps variations in geometric parameters to the pose error will be formulated. So that the influence of each geomettric parameter will be known.

As discussed previously, 5-bar linkage is a closed-loop mechanism, and it can be regarded as consisting of two kinematic chains: $O-A-C-E$ and $O-B-D-E$. For each kinematic chain, the close-loop equation can be modeled as the following:

Kinematic chain: $O-A-C-E$

$$\vec{OE} = \vec{OA} + \vec{AC} + \vec{CE} \tag{IV.27}$$

Kinematic chain: $O-B-D-E$

$$\vec{OE} = \vec{OB} + \vec{BD} + \vec{DE} \tag{IV.28}$$

Assume the position vector of Point E is $\boldsymbol{e}$ with respect to the base frame, $\boldsymbol{e}$ can be calculated in the kinematic chain $O-A-C-E$ by the following equation

$$\boldsymbol{e} = -\frac{1}{2}L_0\boldsymbol{i} + L_1\boldsymbol{\mu}_1 + L_2\boldsymbol{v}_1 \tag{IV.29}$$

Where, $\boldsymbol{i}$ is the unit vector along the x-axis of the base frame, $\boldsymbol{\mu}_1$ is the unit vector along Link L_1, $\boldsymbol{v}_1$ is the unit vector along Link L_2

Differentiating the formula with respect to the geometric parameters, the pose error can be expressed by the variations in geometric parameters

$$\delta\boldsymbol{e} = -\frac{1}{2}\delta L_0\boldsymbol{i} + \delta L_1\boldsymbol{\mu}_1 + L_1\delta\theta_1\mathbf{E}\boldsymbol{\mu}_1 + \delta L_2\boldsymbol{v}_1 + L_2(\delta\theta_1+\delta\theta_2)\boldsymbol{E}\boldsymbol{v}_1 \tag{IV.30}$$

where, δL_0 is the variation in the link length L_0, δL_1 is the variation in the link length L_1, $\delta\theta_1$ is the variation in the joint angle θ_1, δL_2 is the variation in the link length L_2, $\delta\theta_2$ is the variation in the joint angle θ_2, and $\mathbf{E} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$

Since in the kinematic chain $O-A-C-E$ θ_1 is the active joint angle, and θ_2 is the passive joint angle, and in the parallel mechanism, the passive joint can adjust by itself, $\delta\theta_2$

can be expressed by $\delta\theta_1$, for doing this, multiplying $(\mathbf{E}\boldsymbol{v}_1)^T$ by the two sides of Eq. (IV.30), here we get

$$(\mathbf{E}\boldsymbol{v}_1)^T\delta\boldsymbol{e} = -\frac{1}{2}(\mathbf{E}\boldsymbol{v}_1)^T\boldsymbol{i}\delta L_0 + (\mathbf{E}\boldsymbol{v}_1)^T\boldsymbol{\mu}_1\delta L_1 + (\mathbf{E}\boldsymbol{v}_1)^T\mathbf{E}\boldsymbol{\mu}_1 L_1\delta\theta_1 + L_2(\delta\theta_1 + \delta\theta_2) \quad \text{(IV.31)}$$

By solving this equation, $\delta\theta_2$ can be expressed as

$$\begin{aligned}\delta\theta_2 = {} & \frac{1}{L_2}(\mathbf{E}\boldsymbol{v}_1)^T\delta\boldsymbol{e} + \frac{1}{2L_2}(\mathbf{E}\boldsymbol{v}_1)^T\boldsymbol{i}\delta L_0 \\ & - \frac{1}{L_2}(\mathbf{E}\boldsymbol{v}_1)^T\boldsymbol{\mu}_1\delta L_1 - \frac{1}{L_2}(\mathbf{E}\boldsymbol{v}_1)^T\mathbf{E}\boldsymbol{\mu}_1 L_1\delta\theta_1 - \delta\theta_1\end{aligned} \quad \text{(IV.32)}$$

In the same way, for the kinematic chain $O - B - D - E$

$$\boldsymbol{e} = \frac{1}{2}L_0\boldsymbol{i} + L_3\boldsymbol{\mu}_2 + L_4\boldsymbol{v}_2 \quad \text{(IV.33)}$$

Its differentiation is,

$$\delta\boldsymbol{e} = \frac{1}{2}\delta L_0\boldsymbol{i} + \delta L_3\boldsymbol{\mu}_2 + L_3\delta\theta_3\mathbf{E}\boldsymbol{\mu}_2 + \delta L_4\boldsymbol{v}_2 + L_4(\delta\theta_3 + \delta\theta_4)\mathbf{E}\boldsymbol{v}_2 \quad \text{(IV.34)}$$

where, δL_3 is the variation in the link length L_3, $\delta\theta_3$ is the variation in the joint angle θ_3, δL_4 is the variation in the link length L_4, $\delta\theta_4$ is the variation in the joint angle θ_4.

In the 2-DOF 5-bar linkage mechanism, the Joint 1 and Joint 3 are actuated joints, and the Joint 2, Joint 4 and Joint 5 are passive joints. Since the passive joints can adjust by themselves, it is necessary to eliminate the influences of the variations in the passive joints in the variation equations.

Multiplying the row vector $\boldsymbol{v}_1^T$ to the two sides of Eq. (IV.30), we get

$$\boldsymbol{v}_1^T\delta\boldsymbol{e} = -\frac{1}{2}\boldsymbol{v}_1^T\boldsymbol{i}\delta L_0 + \boldsymbol{v}_1^T\boldsymbol{\mu}_1\delta L_1 + \boldsymbol{v}_1^T\mathbf{E}\boldsymbol{\mu}_1 L_1\delta\theta_1 + \boldsymbol{v}_1^T\boldsymbol{v}_1\delta L_2 \quad \text{(IV.35)}$$

Multiply the row vector $\boldsymbol{v}_2^T$ to the two sides of Eq. (IV.34), here we get

$$\boldsymbol{v}_2^T\delta\boldsymbol{e} = \frac{1}{2}\boldsymbol{v}_2^T\boldsymbol{i}\delta L_0 + \boldsymbol{v}_2^T\boldsymbol{\mu}_2\delta L_3 + \boldsymbol{v}_2^T\mathbf{E}\boldsymbol{\mu}_2 L_3\delta\theta_3 + \boldsymbol{v}_2^T\boldsymbol{v}_2\delta L_4 \quad \text{(IV.36)}$$

The two equations can be assembled due to the common variation of position in the end-effector, and finally $\delta\boldsymbol{e}$ can be expressed as

$$\begin{aligned}\delta\boldsymbol{e} = {} & \begin{bmatrix} \boldsymbol{v}_1^T \\ \boldsymbol{v}_2^T \end{bmatrix}^{-1} \\ & \begin{bmatrix} -\frac{1}{2}\boldsymbol{v}_1^T\boldsymbol{i} & \boldsymbol{v}_1^T\boldsymbol{\mu}_1 & \boldsymbol{v}_1^T\boldsymbol{v}_1 & 0 & 0 & 0 & \boldsymbol{v}_1^T\mathbf{E}\boldsymbol{\mu}_1 L_1 & 0 \\ \frac{1}{2}\boldsymbol{v}_2^T\boldsymbol{i} & 0 & 0 & \boldsymbol{v}_2^T\boldsymbol{\mu}_2 & \boldsymbol{v}_2^T\boldsymbol{v}_2 & 0 & 0 & \boldsymbol{v}_2^T\mathbf{E}\boldsymbol{\mu}_2 L_3 \end{bmatrix} \begin{bmatrix} \delta L_0 \\ \delta L_1 \\ \delta L_2 \\ \delta L_3 \\ \delta L_4 \\ \delta L_5 \\ \delta\theta_1 \\ \delta\theta_3 \end{bmatrix}\end{aligned} \quad \text{(IV.37)}$$

So the error mapping matrix for the position error in Point E due to the variations in gemoetric parameters J_e can be defined as the following

$$\mathbf{J}_e = \begin{bmatrix} \boldsymbol{v}_1^T \\ \boldsymbol{v}_2^T \end{bmatrix}^{-1} \begin{bmatrix} -\frac{1}{2}\boldsymbol{v}_1^T\boldsymbol{i} & \boldsymbol{v}_1^T\boldsymbol{\mu}_1 & \boldsymbol{v}_1^T\boldsymbol{v}_1 & 0 & 0 & 0 & \boldsymbol{v}_1^T\mathbf{E}\boldsymbol{\mu}_1 L_1 & 0 \\ \frac{1}{2}\boldsymbol{v}_2^T\boldsymbol{i} & 0 & 0 & \boldsymbol{v}_2^T\boldsymbol{\mu}_2 & \boldsymbol{v}_2^T\boldsymbol{v}_2 & 0 & 0 & \boldsymbol{v}_2^T\mathbf{E}\boldsymbol{\mu}_2 L_3 \end{bmatrix} \tag{IV.38}$$

Recall the expression for $\delta\theta_2$ in Eq. (IV.32), it can be rewritten by substituting the error mapping matrix $\mathbf{J}_e$ and extracting the variations in geometric parameters in the following way

$$\delta\theta_2 = \{\frac{1}{L_2}(\mathbf{E}\boldsymbol{v}_1)^T\mathbf{J}_e + \begin{bmatrix} \frac{1}{2L_2}(\mathbf{E}\boldsymbol{v}_1)^T\boldsymbol{i} & -\frac{1}{L_2}(\mathbf{E}\boldsymbol{v}_1)^T\boldsymbol{\mu}_1 & 0 & 0 & 0 & 0 & -\frac{1}{L_2}(\mathbf{E}\boldsymbol{v}_1)^T\mathbf{E}\boldsymbol{\mu}_1 L_1 - 1 & 0 \end{bmatrix}\} \begin{bmatrix} \delta L_0 \\ \delta L_1 \\ \delta L_2 \\ \delta L_3 \\ \delta L_4 \\ \delta L_5 \\ \delta\theta_1 \\ \delta\theta_3 \end{bmatrix} \tag{IV.39}$$

So the error mapping matrix for the variation in $\delta\theta_2$ can be defined as the following

$$\begin{aligned} \mathbf{J}_{\theta_2} &= \frac{1}{L_2}(\mathbf{E}\boldsymbol{v}_1)^T\mathbf{J}_e + \\ &\begin{bmatrix} \frac{1}{2L_2}(\mathbf{E}\boldsymbol{v}_1)^T\boldsymbol{i} & -\frac{1}{L_2}(\mathbf{E}\boldsymbol{v}_1)^T\boldsymbol{\mu}_1 & 0 & 0 & 0 & 0 & -\frac{1}{L_2}(\mathbf{E}\boldsymbol{v}_1)^T\mathbf{E}\boldsymbol{\mu}_1 L_1 - 1 & 0 \end{bmatrix} \end{aligned} \tag{IV.40}$$

The position of the end-effector $\boldsymbol{p}$ can be expressed by the position $\boldsymbol{e}$

$$\boldsymbol{p} = \boldsymbol{e} + L_5\boldsymbol{v}_1 \tag{IV.41}$$

Differentiate Eq. (IV.41), here we get

$$\delta\boldsymbol{p} = \delta\boldsymbol{e} + \delta L_5\boldsymbol{v}_1 + L_5\mathbf{E}\boldsymbol{v}_1(\delta\theta_1 + \delta\theta_2) \tag{IV.42}$$

Extracting the variations in geometric parameters, the sensitivity matrix of the 5-bar linkage can be deduced

$$\mathbf{J}_p = \mathbf{J}_e + \boldsymbol{v}_1\mathbf{J}_{L_5} + L_5\mathbf{E}\boldsymbol{v}_1(\mathbf{J}_{\theta_1} + \mathbf{J}_{\theta_2}) \tag{IV.43}$$

where,

$$\mathbf{J}_{L_5} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\mathbf{J}_{\theta_1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

V

Sensitivity Analysis Due to Joint Clearances for Robots with Hybrid Legs

Sensitivity analysis method due to joint clearances is presented in Section III.1, this method is not only used to do the analysis for serial robots, but also for parallel robots. Chapter IV studies the sensitivity analysis for the 5-bar linkage based on this method. The sensitivity analysis due to joint clearances without actuation redundancy, with actuation redundancy, as well as for the two legs separately in the 5-bar linkage have been studied. The sensitivity method has been verified for both the closed-loop and open-loop mechanisms.

However, the 5-bar linkage and the legs decomposed from the 5-bar linkage are just with simple serial kinematic chains. There are also some robots with complicated hybrid kinematic chains in legs. Taking the IRSBot-2 robot as an example, each leg is composed of a parallelogram and a distal module. The parallelogram and distal module are both closed-loop structures. The two closed-loop structures are connected in a serial way to form the leg of the IRSBot-2 robot. Thus, the IRSBot-2 robot is comprising of hybrid legs. The previous method for doing sensitivity analysis due to joint clearances is not sufficient to do the sensitivity analysis of the IRSBot-2 robot.

In this chapter, new method for doing sensitivity analysis due to joint clearances is developed, not particularly for the IRSBot-2 robot, but also can be used for the general robots that are with complicated hybrid legs or just with simple serial legs. Section V.1 presents the error estimation model for the robots with hybrid legs. Section V.2 presents the optimization based method to find the maximum positional error and the maximum rotational error for robots with hybrid legs. Section V.3 modifies the optimization method developed in Section V.2, and integrates the equality constraints into inequality constraints.

V.1 Model of Error Estimation

Based on the optimization method developed in Section III.1.3, the maximum pose error of the serial robots and parallel robots can be easily and precisely estimated. But it is just applicable for the robots with serial kinematic chains in legs. However, not all the robots are with simple serial kinematic chains connecting the base and the moving platform, there are also some robots with complicated architecture in legs, as shown in Fig. V.1, which are

called hybrid legs. Robots can not only have identity hybrid legs, but also a combination of different hybrid legs. In this case, the robot gets more complicated. The method developed in Section III.1 won't be sufficient to do the sensitivity analysis for those complicated robot architectures.

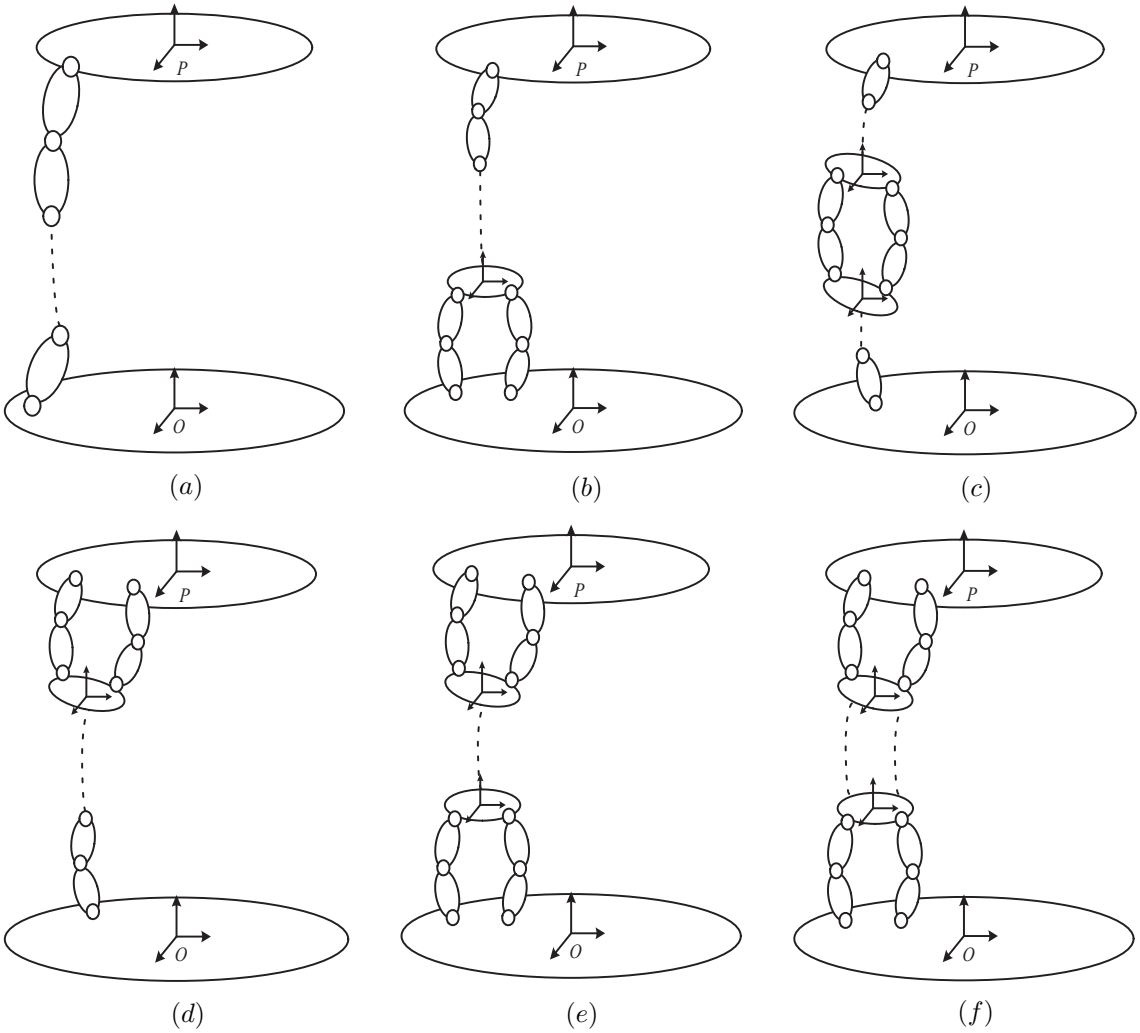


Figure V.1 – Hybrid Legs in Robots

Still, based on the method in Section III.1, the model of sensitivity analysis due to joint clearances for robots with hybrid legs is developed here. As shown in Fig. V.2, the robot is constructed by hybrid legs like Leg i and Leg m, $(i = 1, 2, \ldots, m)$, where m is the number of legs of the robot. The pose of end-efector can be computed from each leg as $\mathbf{P}_i, (i = 1, 2, \ldots, m)$. Due to the hybrid structure of the robot, in each leg there can be some sub-end-effectors. Such as, $E_{i,1}$, $E_{i,2}$, ..., E_{i,n_i}, here n_i is the number of sub-end-effectors in i^{th} leg. For the k^{th} sub-end-effector $E_{i,k}$, $1 \leqslant k \leqslant n_i$, it is connected to the $(k-1)^{th}$ via $m_{i,k}$ legs, and each leg contains $n_{i,k,j}$ joints, here $1 \leqslant j \leqslant m_{i,k}$. Obviously, this structure composed by $E_{i,k}$ and $E_{i,k-1}$ which are connected by m_k legs in between is exactly similar to the robot structure as shown in Fig. III.2, let's call it sub-robot. Please note that, when $m_{i,k} = 1$, the sub-robot is just a serial kinematic chain.

The end-effector of the robot is connected to the last sub-end-effector of each leg E_{i,n_1},

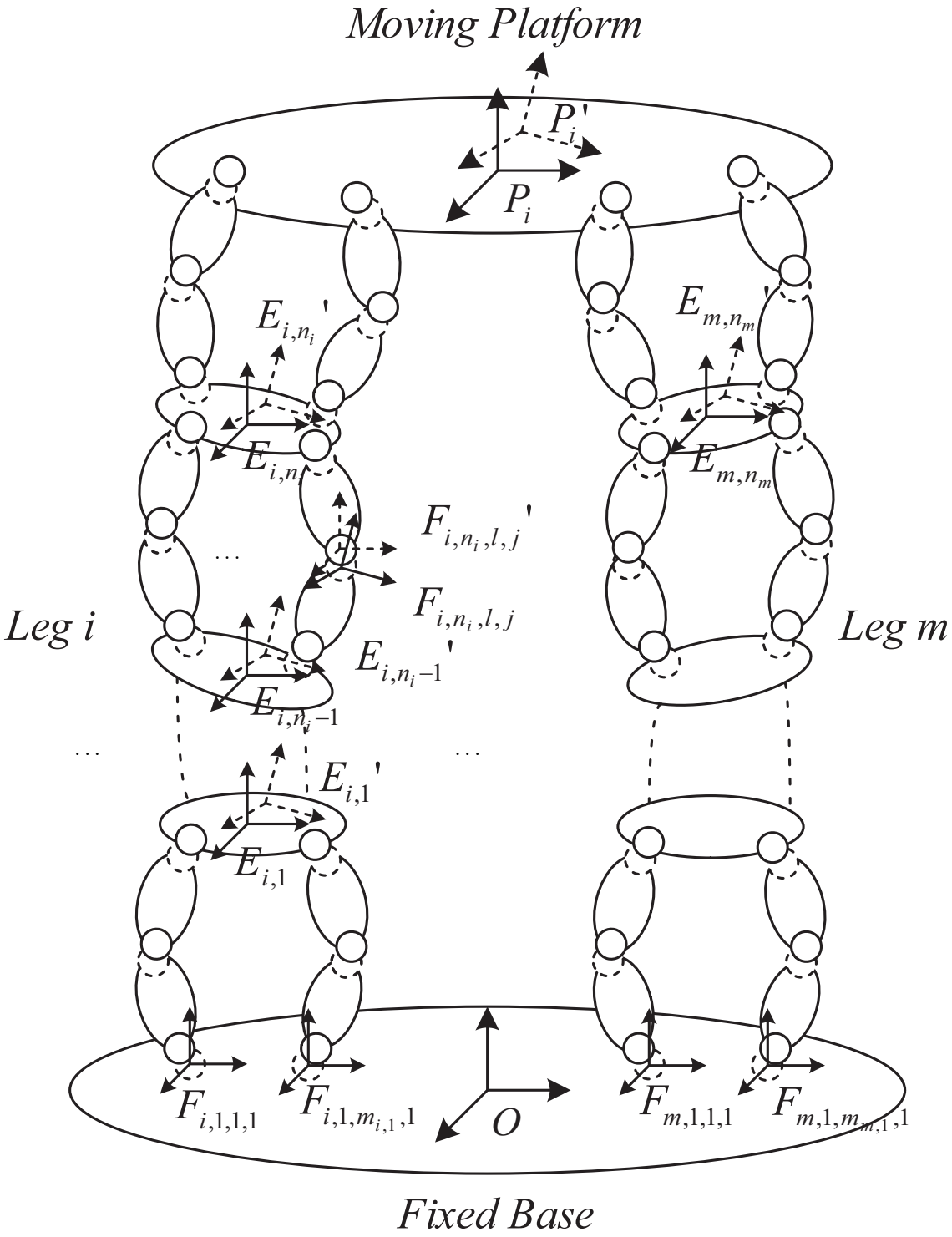


Figure V.2 – Sensitivity Analysis Due to Joint Clearances for Robots with Hybrid Legs

E_{i,n_2}, ..., E_{i,n_i} by $\sum_{i=1}^{m} m_{i,p}$ legs, where $m_{i,p}$ is the number of sbu-legs directly connect from sub-end-effector to the end-effector. Certainly, all these sub-legs are of serial kinematic chains in architecture. The pose error of the end-effector can be estimated by each of the legs by Eq. (V.1)

$$\delta \boldsymbol{p}_{i,n_{i,p},l} = \mathbf{M}_{i,n_{i,p},l}\delta \boldsymbol{s}_{i,n_{i,p},l} + \mathbf{M}^{e}_{i,n_{i,p},l}\delta \boldsymbol{e}_{i,n_i} \tag{V.1}$$

where, i is the index of the leg of the robot, $n_{i,p}$ is equal to $n_i + 1$, that is to say the end-effector/moving-platform of the robot is regarded as the $n_{i,p}^{th}$ sub-end-effector. l is the index of the leg that connects the moving platform to the sub-end-effector E_{i,n_i}, $\boldsymbol{p}_{i,n_{i,p},l}$ is the pose error computed from this sun-leg, $\mathbf{M}_{i,n_{i,p},l}$ is the error mapping matrix for all the error screws in joint clearances of this sub-leg, $\delta \boldsymbol{e}_{i,n_i}$ is the pose error of the sub-end-effector E_{i,n_i},$\mathbf{M}^{e}_{i,n_{i,p},l}$ is the error mapping matrix of the pose error of $\delta \boldsymbol{e}_{i,n_i}$ to the end-effector of the robot.

By doing this, the pose error can be estimated by all the sub-legs that connect the moving platform to the last sub-end-effectors of the legs in terms of Eq. (V.1)

$$\begin{aligned}
\delta \boldsymbol{p}_{1,n_{1,p},1} &= \mathbf{M}_{1,n_{1,p},1}\delta \boldsymbol{s}_{1,n_{1,p},1} + \mathbf{M}^{e}_{1,n_{1,p},1}\delta \boldsymbol{e}_{1,n_1} \\
&\vdots \\
\delta \boldsymbol{p}_{1,n_{1,p},l} &= \mathbf{M}_{1,n_{1,p},l}\delta \boldsymbol{s}_{1,n_{1,p},l} + \mathbf{M}^{e}_{1,n_{1,p},l}\delta \boldsymbol{e}_{1,n_1} \\
&\vdots \\
\delta \boldsymbol{p}_{1,n_{1,p},m_1} &= \mathbf{M}_{1,n_{1,p},m_1}\delta \boldsymbol{s}_{1,n_{1,p},m_1} + \mathbf{M}^{e}_{1,n_{1,p},m_1}\delta \boldsymbol{e}_{1,n_1} \\
&\vdots \\
&\vdots \\
\delta \boldsymbol{p}_{i,n_{i,p},1} &= \mathbf{M}_{i,n_{i,p},1}\delta \boldsymbol{s}_{1,n_{i,p},1} + \mathbf{M}^{e}_{1,n_{i,p},1}\delta \boldsymbol{e}_{i,n_i} \\
&\vdots \\
\delta \boldsymbol{p}_{i,n_{i,p},l} &= \mathbf{M}_{i,n_{i,p},l}\delta \boldsymbol{s}_{1,n_{i,p},l} + \mathbf{M}^{e}_{1,n_{i,p},l}\delta \boldsymbol{e}_{i,n_i} \\
&\vdots \\
\delta \boldsymbol{p}_{i,n_{i,p},m_i} &= \mathbf{M}_{i,n_{i,p},m_i}\delta \boldsymbol{s}_{i,n_{i,p},m_i} + \mathbf{M}^{e}_{i,n_{i,p},m_i}\delta \boldsymbol{e}_{i,n_i} \\
&\vdots \\
&\vdots \\
\delta \boldsymbol{p}_{m,n_{m,p},1} &= \mathbf{M}_{m,n_{m,p},1}\delta \boldsymbol{s}_{m,n_{m,p},1} + \mathbf{M}^{e}_{m,n_{m,p},1}\delta \boldsymbol{e}_{m,n_m} \\
&\cdots \\
\delta \boldsymbol{p}_{m,n_{m,p},l} &= \mathbf{M}_{m,n_{m,p},l}\delta \boldsymbol{s}_{m,n_{m,p},l} + \mathbf{M}^{e}_{m,n_{m,p},l}\delta \boldsymbol{e}_{m,n_m} \\
&\vdots \\
\delta \boldsymbol{p}_{m,n_{m,p},m_m} &= \mathbf{M}_{m,n_{m,p},m_m}\delta \boldsymbol{s}_{m,n_{m,p},m_m} + \mathbf{M}^{e}_{m,n_{m,p},m_m}\delta \boldsymbol{e}_{m,n_m}
\end{aligned} \tag{V.2}$$

where, m_i is the number of legs that connects from the end-effector of the robot to the sub-end-effector E_{i,n_i}, m_m is the number of legs that connects the end-effector of the robot to the sub-end-effector E_{m,n_m}, $n_{i,p}$ is equal to $n_i + 1$. $n_i + 1$ stands for the sequence of the end-effector in Leg i, when it is considered also as a sub-end-effector in Leg i.

Obviously,

$$\begin{aligned}\delta\boldsymbol{p} &= \delta\boldsymbol{p}_{1,n_{1,p},1} = \cdots = \delta\boldsymbol{p}_{1,n_{1,p},l} = \cdots = \delta\boldsymbol{p}_{1,n_{1,p},m_1} = \cdots \\ &= \delta\boldsymbol{p}_{i,n_{i,p},1} = \cdots = \delta\boldsymbol{p}_{i,n_{i,P},l} = \cdots = \delta\boldsymbol{p}_{i,n_{i,p},m_i} = \cdots \\ &= \delta\boldsymbol{p}_{m,n_{m,p},1} = \cdots = \delta\boldsymbol{p}_{m,n_{m,p},l} = \cdots = \delta\boldsymbol{p}_{m,n_{m,p},m_m}\end{aligned} \tag{V.3}$$

For the i^{th} leg ($1 \leqslant i \leqslant m$), k^{th} sub-end-effector ($1 \leqslant k \leqslant n_i$), the sensitivity of the sub-end-effector $\delta\boldsymbol{e}_{i,k}$ can be estimated by the joint clearances in the k^{th} sub-robot and the sensitivity of the $(k-1)^{th}$ sub-end-effector, as Eq. (V.5)

$$\delta\boldsymbol{e}_{i,k,l} = \mathbf{M}_{i,k,l}\delta\boldsymbol{s}_{i,k,l} + \mathbf{M}^e_{i,k,l}\delta\boldsymbol{e}_{i,k-1} \tag{V.4}$$

In the same way, for the 1^{st} sub-end-effector of Leg i, the pose error is

$$\delta\boldsymbol{e}_{i,1,l} = \mathbf{M}_{i,1,l}\delta\boldsymbol{s}_{i,1,l} + \mathbf{M}^e_{i,1,l}\delta\boldsymbol{e}_{i,0} \tag{V.5}$$

where, $\delta\boldsymbol{e}_{i,0} = \delta\boldsymbol{e}_0$ is the screw of pose error of the fixed base.

V.2 Optimization-based Maximum Pose Error

This section aims to find the maximum pose error of the robots with hybrid architecture in legs. As Section III.1, an optimization based method is also employed here,

Let $m_{i,p}$ be the number of legs that connect the end-effector P to the sub-end-effector E_{i,n_i} $n_{i,p,l}$ be the number of the joints for the l^{th} sub-leg that connects the end-effector P to the sub-end-effector E_{i,n_i} $n_{i,p}$ be the number of the sequence of the sub-end-effector in Leg 1, here just consider that the end-effector of the robot is also a sub-end-effector in Leg i, as presented before $n_{i,p} = n_i + 1$

$$\begin{aligned}-p^2_{max} \quad = \quad & \texttt{minimize} - \sum_{k=4,5,6}(\boldsymbol{e}^T_{6,k}\delta\boldsymbol{p})^2, \\ \texttt{over} \quad & \delta\boldsymbol{p}, \delta\boldsymbol{s}_{i,n_{i,p},l,j}, \delta\boldsymbol{e}_{i,n_i} \\ \texttt{subject to} \quad & (\boldsymbol{e}^T_{6,1}\delta\boldsymbol{s}_{i,n_{i,p},l,j})^2 + (\boldsymbol{e}^T_{6,2}\delta\boldsymbol{s}_{i,n_{i,p},l,j})^2 - \Delta\beta^2_{i,n_{i,p},l,j,X,Y} \leqslant 0, \\ & (\boldsymbol{e}^T_{6,3}\delta\boldsymbol{s}_{i,n_{i,p},l,j})^2 - \Delta\beta^2_{i,n_{i,p},l,j,Z} \leqslant 0 \\ & (\boldsymbol{e}^T_{6,4}\delta\boldsymbol{s}_{i,n_{i,p},l,j})^2 + (\boldsymbol{e}^T_{6,5}\delta\boldsymbol{s}_{i,n_{i,p},l,j})^2 - \Delta b^2_{i,n_{i,p},l,j,X,Y} \leqslant 0, \\ & (\boldsymbol{e}^T_{6,6}\delta\boldsymbol{s}_{i,n_{i,p},l,j})^2 - \Delta b^2_{i,n_{i,p},l,j,Z} \leqslant 0, \\ & \delta\boldsymbol{p} = \mathbf{M}_{i,n_{i,p},l}\delta\boldsymbol{s}_{i,n_{i,p},l} + \mathbf{M}^e_{i,n_{i,p},l}\delta\boldsymbol{e}_{i,n_i}, \\ & j = 1, ..., n_{i,p,l}, l = 1, ..., m_{i,p}, i = 1, ..., m.\end{aligned} \tag{V.6}$$

whereas, $\delta \boldsymbol{e}_{i,n_i}$ should be satisfied by the following constraints

$$\begin{aligned} \delta \boldsymbol{e}_{i,n_i} &= \mathbf{M}_{i,n_i,l}\delta \boldsymbol{s}_{i,n_i,l} + \mathbf{M}^{e}_{i,n_i,l}\delta \boldsymbol{e}_{i,n_i-1} \\ \texttt{over} &\quad \delta \boldsymbol{s}_{i,n_i,l}, \delta \boldsymbol{e}_{i,n_i-1} \end{aligned} \tag{V.7}$$

where, $\delta \boldsymbol{s}_{i,n_i,l}$ is the joint clearances in the l^{th} sub-leg of the n_i^{th} sub-robot in Leg i, $\mathbf{M}_{i,n_i,l}$ is the error mapping matrix for the l^{th} sub-leg of the n_i^{th} sub-robot in Leg i, $\delta \boldsymbol{e}_{i,n_i-1}$ is the error pose of the $(n_i-1)^{th}$ sub-end-effector, and $\mathbf{M}^{e}_{i,n_i,l}$ is the error mapping matrix through Leg l.

As $\delta \boldsymbol{s}_{i,n_i,l}$ is the vector that contains all the error screws due to joint clearances in Leg l, it should also be constrained as following

$$\begin{aligned} (\boldsymbol{e}^T_{6,1}\delta \boldsymbol{s}_{i,n_i,l,j})^2 + (\boldsymbol{e}^T_{6,2}\delta \boldsymbol{s}_{i,n_i,l,j})^2 - \Delta\beta^2_{i,n_i,l,j,X,Y} &\leqslant 0 \\ (\boldsymbol{e}^T_{6,3}\delta \boldsymbol{s}_{i,n_i,l,j})^2 - \Delta\beta^2_{i,n_i,l,j,Z} &\leqslant 0 \\ (\boldsymbol{e}^T_{6,4}\delta \boldsymbol{s}_{i,n_i,l,j})^2 + (\boldsymbol{e}^T_{6,5}\delta \boldsymbol{s}_{i,n_i,l,j})^2 - \Delta b^2_{i,n_i,l,j,X,Y} &\leqslant 0 \\ (\boldsymbol{e}^T_{6,6}\delta \boldsymbol{s}_{i,n_{i,p},l,j})^2 - \Delta b^2_{i,n_{i,p},l,j,Z} &\leqslant 0 \end{aligned}$$

And $\delta \boldsymbol{e}_{i,n_i}$ can be computed in the same way, with the joint clearances $\delta \boldsymbol{s}_{i,n_i-1,l,j}$ constrained in the same way. Thus, recursively, all the joint clearances can be covered for the optimization problem to find the maximum positional error of the end-effector.

The optimization problem for finding the maximum positional error can be rewritten in the following way

$$
\begin{aligned}
-p_{max}^2 \quad = \quad & \texttt{minimize} - \sum_{k=4,5,6} (\boldsymbol{e}_{6,k}^T \delta \boldsymbol{p})^2, && \text{(V.8)}\\
& \texttt{over} \quad \delta \boldsymbol{p}, \delta \boldsymbol{s}_{i,n_{i,p},l,j}, \delta \boldsymbol{s}_{i,n_i,l,j}, \ldots, \delta \boldsymbol{s}_{i,1,l,j}, \delta \boldsymbol{e}_{i,n_i}, \ldots, \delta \boldsymbol{e}_{i,1}, \delta \boldsymbol{e}_{i,0}\\
& \texttt{subject to} \quad (\boldsymbol{e}_{6,1}^T \delta \boldsymbol{s}_{i,n_{i,p},l,j})^2 + (\boldsymbol{e}_{6,2}^T \delta \boldsymbol{s}_{i,n_{i,p},l,j})^2 - \Delta\beta_{i,n_{i,p},l,j,X,Y}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,3}^T \delta \boldsymbol{s}_{i,n_{i,p},l,j})^2 - \Delta\beta_{i,n_{i,p},l,j,Z}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,4}^T \delta \boldsymbol{s}_{i,n_{i,p},l,j})^2 + (\boldsymbol{e}_{6,5}^T \delta \boldsymbol{s}_{i,n_{i,p},l,j})^2 - \Delta b_{i,n_{i,p},l,j,X,Y}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,6}^T \delta \boldsymbol{s}_{i,n_{i,p},l,j})^2 - \Delta b_{i,n_{i,p},l,j,Z}^2 \leqslant 0,\\
& \qquad \delta \boldsymbol{p} = \mathbf{M}_{i,n_{i,p},l} \delta \boldsymbol{s}_{i,n_{i,p},l} + \mathbf{M}_{i,n_{i,p},l}^e \delta \boldsymbol{e}_{i,n_i}, && \text{(V.9)}\\
& \qquad j = 1, ..., n_{i,p,l}, l = 1, ..., m_{i,p}, i = 1, ..., m;\\
& \qquad \delta \boldsymbol{e}_{i,n_i} = \mathbf{M}_{i,n_i,l} \delta \boldsymbol{s}_{i,n_i,l} + \mathbf{M}_{i,n_i,l}^e \delta \boldsymbol{e}_{i,n_i-1}, && \text{(V.10)}\\
& \qquad (\boldsymbol{e}_{6,1}^T \delta \boldsymbol{s}_{i,n_i,l,j})^2 + (\boldsymbol{e}_{6,2}^T \delta \boldsymbol{s}_{i,n_i,l,j})^2 - \Delta\beta_{i,n_i,l,j,X,Y}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,3}^T \delta \boldsymbol{s}_{i,n_i,l,j})^2 - \Delta\beta_{i,n_i,l,j,Z}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,4}^T \delta \boldsymbol{s}_{i,n_i,l,j})^2 + (\boldsymbol{e}_{6,5}^T \delta \boldsymbol{s}_{i,n_i,l,j})^2 - \Delta b_{i,n_i,l,j,X,Y}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,6}^T \delta \boldsymbol{s}_{i,n_i,l,j})^2 - \Delta b_{i,n_i,l,j,Z}^2 \leqslant 0,\\
& \qquad j = 1, ..., n_{i,n_i,l}, l = 1, ..., m_{i,n_i}, i = 1, ..., m;\\
& \qquad \vdots\\
& \qquad \delta \boldsymbol{e}_{i,k} = \mathbf{M}_{i,k,l} \delta \boldsymbol{s}_{i,k,l} + \mathbf{M}_{i,k,l}^e \delta \boldsymbol{e}_{i,k-1}, && \text{(V.11)}\\
& \qquad (\boldsymbol{e}_{6,1}^T \delta \boldsymbol{s}_{i,k,l,j})^2 + (\boldsymbol{e}_{6,2}^T \delta \boldsymbol{s}_{i,k,l,j})^2 - \Delta\beta_{i,k,l,j,X,Y}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,3}^T \delta \boldsymbol{s}_{i,k,l,j})^2 - \Delta\beta_{i,k,l,j,Z}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,4}^T \delta \boldsymbol{s}_{i,k,l,j})^2 + (\boldsymbol{e}_{6,5}^T \delta \boldsymbol{s}_{i,k,l,j})^2 - \Delta b_{i,k,l,j,X,Y}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,6}^T \delta \boldsymbol{s}_{i,k,l,j})^2 - \Delta b_{i,k,l,j,Z}^2 \leqslant 0,\\
& \qquad j = 1, ..., n_{i,k,l}, l = 1, ..., m_{i,k}, i = 1, ..., m;\\
& \qquad \vdots\\
& \qquad \delta \boldsymbol{e}_{i,1} = \mathbf{M}_{i,1,l} \delta \boldsymbol{s}_{i,1,l} + \mathbf{M}_{i,1,l}^e \delta \boldsymbol{e}_{i,0}, && \text{(V.12)}\\
& \qquad (\boldsymbol{e}_{6,1}^T \delta \boldsymbol{s}_{i,1,l,j})^2 + (\boldsymbol{e}_{6,2}^T \delta \boldsymbol{s}_{i,1,l,j})^2 - \Delta\beta_{i,1,l,j,X,Y}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,3}^T \delta \boldsymbol{s}_{i,1,l,j})^2 - \Delta\beta_{i,1,l,j,Z}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,4}^T \delta \boldsymbol{s}_{i,1,l,j})^2 + (\boldsymbol{e}_{6,5}^T \delta \boldsymbol{s}_{i,1,l,j})^2 - \Delta b_{i,1,l,j,X,Y}^2 \leqslant 0,\\
& \qquad (\boldsymbol{e}_{6,6}^T \delta \boldsymbol{s}_{i,1,l,j})^2 - \Delta b_{i,1,l,j,Z}^2 \leqslant 0,\\
& \qquad j = 1, ..., n_{i,1,l}, l = 1, ..., m_{i,1}, i = 1, ..., m;
\end{aligned}
$$

where, m is the number of legs of the robot, n_i is the number of sub-end-effector of Leg i, $m_{i,p}$ is the number of sub-legs that connect the end-effector P to the sub-end-effector E_{i,n_i},$n_{i,p,l}$ is the number of the joints for the l^{th} sub-leg that connects the end-effector P to the sub-end-effector E_{i,n_i} $n_{i,p}$ as discussed before, $n_{i,p} = n_i + 1$; $\mathbf{M}_{i,k,l}$ is the error mapping matrix for error screw in Leg l of the k^{th} sub-robot in Leg i, $\mathbf{M}_{i,k,l}^e$ is the error mapping matrix for pose error of the sub-end-effector $\delta \boldsymbol{e}_{i,k-1}$ throughout the l^{th} sub-leg.

For finding the maximum rotational error of the end-effector, it is in the same way above. Where the objective function would be Eq. (V.13), and the constraints are same.

$$-r_{max}^2 = \texttt{minimize} - \sum_{k=1,2,3} (\boldsymbol{e}_{6,k}^T \delta \boldsymbol{p})^2 \tag{V.13}$$

V.3 Maximum Pose Error with Only Inequality Constraints

V.3.1 QR Factorization for Sub-legs Connected to the Moving-Platform

Optimization problems for finding the maximum positional error and the rotational error contain both inequality constraints and equality constraints. Since the inequality constraints can be easily satisfied but whereas, the equality constraints can not. It is better to eliminate the equality constraints. As is presented in section III.1.3, a QR factorization based method has been proposed to eliminate the equality constraints for finding the maximum pose error of the robots with just simple serial chain in their legs. In the same way, the optimization problems for finding the maximum pose error of the robots with complicated hybrid legs can also be adapted in the optimization problems with only inequality constraints.

Eq. (V.2) and Eq. (V.3) are the expressions for the pose error of the end-effector. Thus, in the optimization problem for finding the maximum pose error, we use the equality constraints. By adapting the form of the equations, it will be possible to use the QR factorization, and then integrate the equality constraints into the inequality constraints. Here we go,

$$\underbrace{\delta \boldsymbol{p}_{i,n_{i,p},l} - \mathbf{M}_{1,n_{i,p},l}^{e} \delta \boldsymbol{e}_{i,n_{i,p}}}_{\delta \boldsymbol{p}_{i,n_{i,p},l,p}} = \mathbf{M}_{i,n_{i,p},l} \delta \boldsymbol{s}_{i,n_{i,p},l} \tag{V.14}$$

In Eq. (V.14), the transpose of matrix $\mathbf{M}_{i,n_i,l}$ can be decomposed by QR factorization, as the following

$$\begin{aligned} \mathbf{M}_{i,n_{i,p},l}^T &= \mathbf{V}_{i,n_{i,p},l} \mathbf{U}_{i,n_{i,p},l} && \text{(V.15)} \\ &= \begin{bmatrix} \mathbf{V}_{i,n_{i,p},l,1} & \mathbf{V}_{i,n_{i,p},l,2} \end{bmatrix} \begin{bmatrix} \mathbf{U}_{i,n_{i,p},l,1} \\ \mathbf{0}_{(6n_{i,p,l}-6)\times 6} \end{bmatrix} \\ &= \mathbf{V}_{i,n_{i,p},l,1} \mathbf{U}_{i,n_{i,p},l,1} && \text{(V.16)} \end{aligned}$$

where, $n_{i,p,l}$ is the number of joints in the l^{th} sub-leg that connects the moving platform to the sub-end-effector in Leg i of the robot.

Similar to Eq. (III.31), here we define

$$\delta \boldsymbol{q}_{i,n_{i,p},l} = \begin{bmatrix} \delta \boldsymbol{q}_{i,n_{i,p},l,1} \\ \delta \boldsymbol{q}_{i,n_{i,p},l,2} \end{bmatrix} = \mathbf{V}_{i,n_{i,p},l}^T \delta \boldsymbol{s}_{i,n_{i,p},l} \tag{V.17}$$

Similar to Eq. (III.35),

$$\delta \boldsymbol{s}_{i,n_{i,p},l} = \mathbf{V}_{i,n_{i,p},l,1}\mathbf{U}^{-T}_{i,n_{i,p},l,1}\delta \boldsymbol{p}_{i,n_{i,p},l,p} + \mathbf{V}_{i,n_{i,p},l,2}\delta \boldsymbol{q}_{i,n_{i,p},l,2} \tag{V.18}$$

Taking into consideration of all the joint clearances in sub-legs that connect from the moving platform to the sub-end-effector E_{i,n_i}, the remaining optimization variables can be grouped into array

$$\delta \boldsymbol{u}_{i,n_{i,p},l} = \left[\begin{array}{cccc} \delta \boldsymbol{p}^T_{i,n_{i,p},l,p}, & \delta \boldsymbol{q}^T_{1,n_{1,p},1,2}, & \delta \boldsymbol{q}^T_{1,n_{1,p},2,2}, & \ldots, \quad \delta \boldsymbol{q}^T_{m,n_{m,p},m_{m,p},2} \end{array} \right]^T \in \mathbb{R}^{v_p} \tag{V.19}$$

where,

$$v_p = 6 + 6\sum_{i=1}^{m}\sum_{l=1}^{m_{i,p}}(n_{i,n_{i,p},l} - 1) \tag{V.20}$$

It should be noted that,

$$\delta \boldsymbol{p}_{i,n_{i,p},l,p} = \delta \boldsymbol{p}_{i,n_{i,p},l} - \mathbf{M}^e_{1,n_{i,p},l}\delta \boldsymbol{e}_{i,n_i} \tag{V.21}$$

Thanks to Eq. (V.3), the pose errors estamited from all the legs that connect the moving-platform and the sub-end-effector E_{i,n_i} are equal to each other, so the Eq. (V.21) can be rewritten as

$$\delta \boldsymbol{p}_{i,n_{i,p},l,p} = \delta \boldsymbol{p} - \mathbf{M}^e_{1,n_{i,p},l}\delta \boldsymbol{e}_{i,n_i} \tag{V.22}$$

Since $\mathbf{M}^e_{1,n_{i,p},l}\delta \boldsymbol{e}_{i,n_i}$ computed from different legs are not the same, $\delta \boldsymbol{p}_{i,n_{i,p},l,p}$ varies from leg to leg. Therefore, for the array $\delta \boldsymbol{u}_{i,n_{i,p},l}$, the first 6 elements are not the same, while the left $v_p - 6$ elements keep the same for all the legs. Here we have,

$$\delta \boldsymbol{u}_{1,n_{1,p},1} = \left[\begin{array}{cccc} \delta \boldsymbol{p}^T_{1,n_{1,p},1,p}, & \delta \boldsymbol{q}^T_{1,n_{1,p},1,2}, & \delta \boldsymbol{q}^T_{1,n_{1,p},2,2}, & \ldots, \quad \delta \boldsymbol{q}^T_{m,n_{m,p},m_{m,p},2} \end{array} \right]^T \in \mathbb{R}^{v_p} \tag{V.23}$$

$$\vdots$$

$$\delta \boldsymbol{u}_{i,n_{i,p},l} = \left[\begin{array}{cccc} \delta \boldsymbol{p}^T_{i,n_{i,p},l,p}, & \delta \boldsymbol{q}^T_{1,n_{1,p},1,2}, & \delta \boldsymbol{q}^T_{1,n_{1,p},2,2}, & \ldots, \quad \delta \boldsymbol{q}^T_{m,n_{m,p},m_{m,p},2} \end{array} \right]^T \in \mathbb{R}^{v_p} \tag{V.24}$$

$$\vdots$$

$$\delta \boldsymbol{u}_{m,n_{m,p},m_{m,p}} = \left[\begin{array}{cccc} \delta \boldsymbol{p}^T_{m,n_{m,p},m_{m,p},p}, & \delta \boldsymbol{q}^T_{1,n_{1,p},1,2}, & \delta \boldsymbol{q}^T_{1,n_{1,p},2,2}, & \ldots, \quad \delta \boldsymbol{q}^T_{m,n_{m,p},m_{m,p},2} \end{array} \right]^T \in \mathbb{R}^{v_p} \tag{V.25}$$

So that, $\delta \boldsymbol{s}_{i,n_{i,p},l}$ can be expressed as

$$\delta \boldsymbol{s}_{i,n_{i,p},l} = \boldsymbol{\Upsilon}_{i,n_{i,p},l}\delta \boldsymbol{u}_{i,n_{i,p},l} \tag{V.26}$$

where,

$$\boldsymbol{\Upsilon}_{i,n_{i,p},l} = \left[\begin{array}{ccc} \mathbf{V}_{i,n_{i,p},l,1}\mathbf{U}^{-T}_{i,n_{i,p},l,1} & \mathbf{V}_{i,n_{i,p},l,2} & \mathbf{0}_{6n_{i,n_{i,p},l}\times(v_p-12)} \end{array} \right] \in \mathbb{R}^{6n_{i,n_{i,p},l}\times v^p} \tag{V.27}$$

V.3.2 QR Factorization for Legs in Sub-robots

For the k^{th} sub-robot in Leg i, from Eq. (V.5), the pose error of the sub-end-effector $E_{i,k}$ can be estimated from any sub-leg that connects from $E_{i,k}$ to $E_{i,k-1}$. Of course, the pose error estimated from all the legs are equal to each other

$$\delta\boldsymbol{e}_{i,k} = \delta\boldsymbol{e}_{i,k,1} = \cdots = \delta\boldsymbol{e}_{i,k,l} = \cdots = \delta\boldsymbol{e}_{i,k,m_{i,k}} \tag{V.28}$$

Again, Eq. (V.5) can be adapted as

$$\underbrace{\delta\boldsymbol{e}_{i,k} - \mathbf{M}^{e}_{i,k,l}\delta\boldsymbol{e}_{i,k-1}}_{\delta\boldsymbol{e}_{i,k,l,e}} = \mathbf{M}_{i,k,l}\delta\boldsymbol{s}_{i,k,l} \tag{V.29}$$

Certainly, $\mathbf{M}^{e}_{i,k,l}\delta\boldsymbol{e}_{i,k-1}$ varies from leg to leg in the sub-robot, $\delta\boldsymbol{e}_{i,k,l,e}$ varies as well.

Matrix $\mathbf{M}_{i,k,l}$ in Eq. (V.29) can also be decomposed by QR factorization, similarly we get

$$\begin{aligned} \mathbf{M}^{T}_{i,k,l} &= \mathbf{V}_{i,k,l}\mathbf{U}_{i,k,l} && \text{(V.30)} \\ &= \left[\begin{array}{cc} \mathbf{V}_{i,k,l,1} & \mathbf{V}_{i,k,l,2} \end{array} \right] \left[\begin{array}{c} \mathbf{U}_{i,k,l,1} \\ \mathbf{0}_{(6n_{i,k,l}-6)\times 6} \end{array} \right] \\ &= \mathbf{V}_{i,k,l,1}\mathbf{U}_{i,k,l,1} && \text{(V.31)} \end{aligned}$$

where, $n_{i,k,l}$ is the number of joints in the l^{th} leg in the sub-robot k in Leg i of the robot.

In the same way as Eq. (V.17), here we define

$$\delta\boldsymbol{q}_{i,k,l} = \left[\begin{array}{c} \delta\boldsymbol{q}_{i,k,l,1} \\ \delta\boldsymbol{q}_{i,k,l,2} \end{array} \right] = \mathbf{V}^{T}_{i,k,l}\delta\boldsymbol{s}_{i,k,l} \tag{V.32}$$

Similar to Eq. (V.18),

$$\delta\boldsymbol{s}_{i,k,l} = \mathbf{V}_{i,k,l,1}\mathbf{U}^{-T}_{i,k,l,1}\delta\boldsymbol{p}_{i,k,l,p} + \mathbf{V}_{i,k,l,2}\delta\boldsymbol{q}_{i,k,l,2} \tag{V.33}$$

Here, the

$$\delta\boldsymbol{u}_{i,k,l} = \left[\begin{array}{ccccc} \delta\boldsymbol{p}^{T}_{i,k,l,e}, & \delta\boldsymbol{q}^{T}_{i,k,1,2}, & \delta\boldsymbol{q}^{T}_{i,k,2,2}, & \ldots, & \delta\boldsymbol{q}^{T}_{i,k,m_{i,k},2} \end{array} \right]^{T} \in \mathbb{R}^{v_{e_{i,k}}} \tag{V.34}$$

where,

$$v_{e_{i,k}} = 6 + 6\sum_{l=1}^{m_{i,k}}(n_{i,k,l} - 1) \tag{V.35}$$

V.3.3 Optimization Problem with Only Inequality Constraints

Based on the QR factorizations for the error mapping matrix of the sub-legs that directly connect to the moving-platform and those of sub-legs in the sub-robots, it is possible to eliminate the equality constraints in the optimization problems (V.8) and (V.13) by integrating the the equality constraints of the pose error of the end-effector and the pose error of the sub-end-effector into the inequality constraints.

According to Eq. (V.24) and Eq. (V.34), for the whole robot, an array of decision variables $\delta \boldsymbol{u}$ with a dimension of v can be chosen

$$\delta \boldsymbol{u} \in \mathbb{R}^v \tag{V.36}$$

where v can be computed as

$$v = v_p + \sum_{i=1}^{m} \sum_{k=1}^{n_i} v_{e_{i,k}} \tag{V.37}$$

Among the v decision variables for the optimization problems, it is noteworthy that the pose error of the robot is composed by the first 6 elements in $\delta \boldsymbol{u}$, that is

$$\delta \boldsymbol{p} = \delta \boldsymbol{u}(1:6) \tag{V.38}$$

Elements from the 7^{th} to the v_p^{th} are the elements for the sub-legs connected from the moving-platform to the last sub-end-effector of each leg of the robot.

$$\delta \boldsymbol{u}(7:v_p) = \left[\begin{array}{cccc} \delta \boldsymbol{q}_{1,n_{1,p},1,2}^T, & \delta \boldsymbol{q}_{1,n_{1,p},2,2}^T, & \ldots, & \delta \boldsymbol{q}_{m,n_{m,p},m_{m,p},2}^T \end{array} \right]^T \tag{V.39}$$

For the pose error of the last sub-end-effector in Leg i

$$\begin{aligned} \delta \boldsymbol{e}_{1,n_1} &= \delta \boldsymbol{u}(v_p + 1 : v_p + 6) && \text{(V.40)} \\ &\vdots \\ \delta \boldsymbol{e}_{i,n_i} &= \delta \boldsymbol{u}(v_p + \sum_{j=1}^{i-1} v_{e_{j,n_j}} + 1 : v_p + \sum_{j=1}^{i-1} v_{e_{j,n_j}} + 6) && \text{(V.41)} \\ &\vdots \\ \delta \boldsymbol{e}_{m,n_m} &= \delta \boldsymbol{u}(v_p + \sum_{j=1}^{m-1} v_{e_{j,n_j}} + 1 : v_p + \sum_{j=1}^{m-1} v_{e_{j,n_j}} + 6) && \text{(V.42)} \end{aligned}$$

For a general pose error of the sub-end-effector $E_{i,k}$, for $k < n_i$

$$\delta \boldsymbol{e}_{i,k} = \delta \boldsymbol{u}(num_{i,k} + 1 : num_{i,k} + 6) \tag{V.43}$$

where,

$$num_{i,k} = v_p + \sum_{j=1}^{m} v_{e_{j,n_j}} + \sum_{j=1}^{i-1}\sum_{l=n_j}^{1} v_{e_{j,l}} + \sum_{l=n_i}^{k} v_{e_{i,l}} \tag{V.44}$$

In the constraint function, the pose error $\delta\boldsymbol{p}$ of the robot and the pose error $\delta\boldsymbol{e}_{i,k}$ of the sub-end-effector should first be extracted from $\delta\boldsymbol{u}$, then construct the arrays of variables $\delta\boldsymbol{u}_{i,n_{i,p},l}$ and $\delta\boldsymbol{u}_{i,k,l}$.

So the inequality constraints for $\delta\boldsymbol{u}_{i,n_{i,p},l}$ will be

$$\begin{aligned} f_{i,n_{i,p},l,j,h} &= \delta\boldsymbol{u}^T_{i,n_{i,p},l,j}\mathbf{F}_{i,n_{i,p},l,j,h}\delta\boldsymbol{u}_{i,n_{i,p},l,j} - 1 \leqslant 0 \\ & h = 1,\ldots,4, j = 1,\ldots,n_{i,p,l}, \\ & l = 1,\ldots,m_{i,n_{i,p}}, \\ & i = 1,\ldots,m \end{aligned} \tag{V.45}$$

where,

$$\begin{aligned} \mathbf{F}_{i,n_{i,p},l,j,1} &= (1/\Delta\beta^2_{i,n_{i,p},l,j,X,Y})\Upsilon^T_{i,n_{i,p},l}(\boldsymbol{e}_{6n_{i,p,l},6j-5}\boldsymbol{e}^T_{6n_{i,p,l},6j-5} \\ &\quad +\boldsymbol{e}_{6n_{i,p,l},6j-4}\boldsymbol{e}^T_{6n_{i,p,l},6j-4})\Upsilon_{i,n_{i,p},l} \\ \mathbf{F}_{i,n_{i,p},l,j,2} &= (1/\Delta\beta^2_{i,n_{i,p},l,j,Z})\Upsilon^T_{i,n_{i,p},l}(\boldsymbol{e}_{6n_{i,p,l},6j-3}\boldsymbol{e}^T_{6n_{i,p,l},6j-3})\Upsilon_{i,n_{i,p},l} \\ \mathbf{F}_{i,n_{i,p},l,j,3} &= (1/\Delta b^2_{i,n_{i,p},l,j,X,Y})\Upsilon^T_{i,n_{i,p},l}(\boldsymbol{e}_{6n_{i,p,l},6j-2}\boldsymbol{e}^T_{6n_{i,p,l},6j-2} \\ &\quad +\boldsymbol{e}_{6n_{i,p,l},6j-1}\boldsymbol{e}^T_{6n_{i,p,l},6j-1})\Upsilon_{i,n_{i,p},l} \\ \mathbf{F}_{i,n_{i,p},l,j,4} &= (1/\Delta b^2_{i,n_{i,p},l,j,Z})\Upsilon^T_{i,n_{i,p},l}(\boldsymbol{e}_{6n_{i,p,l},6j}\boldsymbol{e}^T_{6n_{i,p,l},6j})\Upsilon_{i,n_{i,p},l} \end{aligned}$$

Similarly, the constraints for the k^{th} sub-robot in Leg i,

$$\begin{aligned} f_{i,k,l,j,h} &= \delta\boldsymbol{u}^T_{i,k,l,j}\mathbf{F}_{i,k,l,j,h}\delta\boldsymbol{u}_{i,k,l,j} - 1 \leqslant 0 \\ & h = 1,\ldots,4, j = 1,\ldots,n_{i,p,l}, \\ & l = 1,\ldots,m_{i,k}, \\ & i = 1,\ldots,m \end{aligned} \tag{V.46}$$

where,

$$\begin{aligned} \mathbf{F}_{i,k,l,j,1} &= (1/\Delta\beta^2_{i,k,l,j,X,Y})\Upsilon^T_{i,k,l}(\boldsymbol{e}_{6n_{i,p,l},6j-5}\boldsymbol{e}^T_{6n_{i,p,l},6j-5} \\ &\quad +\boldsymbol{e}_{6n_{i,p,l},6j-4}\boldsymbol{e}^T_{6n_{i,p,l},6j-4})\Upsilon_{i,k,l} \\ \mathbf{F}_{i,k,l,j,2} &= (1/\Delta\beta^2_{i,k,l,j,Z})\Upsilon^T_{i,k,l}(\boldsymbol{e}_{6n_{i,p,l},6j-3}\boldsymbol{e}^T_{6n_{i,p,l},6j-3})\Upsilon_{i,k,l} \\ \mathbf{F}_{i,k,l,j,3} &= (1/\Delta b^2_{i,k,l,j,X,Y})\Upsilon^T_{i,k,l}(\boldsymbol{e}_{6n_{i,p,l},6j-2}\boldsymbol{e}^T_{6n_{i,p,l},6j-2} \\ &\quad +\boldsymbol{e}_{6n_{i,p,l},6j-1}\boldsymbol{e}^T_{6n_{i,p,l},6j-1})\Upsilon_{i,k,l} \\ \mathbf{F}_{i,k,l,j,4} &= (1/\Delta b^2_{i,k,l,j,Z})\Upsilon^T_{i,k,l}(\boldsymbol{e}_{6n_{i,p,l},6j}\boldsymbol{e}^T_{6n_{i,p,l},6j})\Upsilon_{i,k,l} \end{aligned}$$

VI

Sensitivity Analysis of the IRSBot-2 Due to Joint Clearances

In this chapter, we focus on the sensitivity analysis due to joint clearances of the IRSBot-2 robot. As is presented in Chapter II, the IRSBot-2 robot is a novel 2-DOF translational robot. In addition, it is also a spatial robot, where the distal module is designed to overcome the traction, compression and torsion. Certainly, due to the joint clearances in the distal module and parallelograms, the end-effector will also exihibit pose error.

Generally it will be easier to assemble the robot with joint clearances. Especially for the distal module. As it can be seen from Fig. II.1, the distal module is composed of 4 links that are not parallel to each other. To define one plane, 3 independant points are enough. Here for the IRSBot-2 robot, 3 links will be enough to constrain the moving-platform. Hence with 4 links in the distal module, the robot is over constrained. In this case, one link should be dependant on the other three links so be able to assemble the distal module. But actually, sometimes we would fail to do this due to the geometric errors produced during the manufacturing phase. Thus, it is better to leave some joint clearances to make sure that the robot will be able to be assembled.

So here the problem comes, how the joint clearances affect the accuracy of the IRSBot-2 robot? Unlike the 2-DOF planar 5-bar linkage, the architecture of the 2-DOF IRSBot-2 robot is more complicated. In each leg, it contains two close loops connected in a serial way, which is called a hybrid leg. The model for sensitivity analysis due to joint clearances created in Chapter III can just be applicable for robots with simple serial kinematic chains in legs. For the robots with hybrid legs, the model developed in Chapter V should be adopted.

In this chapter, Section VI.1 studies the sensitivity to joint clearances just in the distal module of the IRSBot-2 robot. Section VI.2 studies the sensitivity to joint clearances, with taking into considerations of the joint clearances both in the parallelogram and the distal module of the IRSBot-2 robot. Section VI.3 compares the results of sensitivity analysis due to joint clearances in the distial module and to joint clearances in both the distal module and the parallelogram.

VI.1 Sensitivity Analysis Due to Joint Clearances in Distal Module

Regardless of the joint clearances in the parallelograms, there will only be joint clearances in the distal module. The distal part can also be regarded as a parallel robot with just serial

kinematic chains in legs. The only difference is that the base of this parallel robot is the movable elbows in the two legs. But the method for doing the sensitivity analysis due to joint clearances developed in Chapter III can still be applied. However, we can not avoid the joint clearances in the parallelograms even if they are small, it will be quite interesting and of practical significance to take into consideration of the joint clearances in the parallelogram, where the method develped in Chapter V will be applied.

The distal module is composed of 4 links ($E_{ji}F_{ji}$, $i = 1, 2, j = 1, 2$), which are not parallel with each other, as is shown in Fig. II.2. The 4 links are connected to the moving-platform and two elbows by the universal joints. Obviously, if the two elbows are fixed, the moving-platform won't be able to move. But with joint clearances in the distal module, the moving-platform will gain extra and uncontrolable motions.

Structurally, the distal module can be regarded as a parallel robot $(4 - UU)$ with 4 serial kinematic chains connected to movable bases. For doing the sensitivity analysis for the IRSBot-2 robot due to joint clearances in the distal module, the pose of the elbow should be computed first by using the modified D-H parameters as is show in Tab. II.1 and Tab II.2.

As is shown in Fig. II.7, Frame 6 is attached to Elbow 1 at Point E_1, and Frame 22 is attached to Elbow 2 at Point E_2. Thus the poses of the elbows can be calculated by the kinematic chain $O - A_i - B_i - E_i, i = 1, 2$:

For Elbow 1

$$\mathbf{S}_6 = \mathbf{S}_1\mathbf{S}_2\mathbf{S}_6 \tag{VI.1}$$

For Elbow 2

$$\mathbf{S}_{22} = \mathbf{S}_{17}\mathbf{S}_{18}\mathbf{S}_22 \tag{VI.2}$$

The distal module can be decomposed into 4 kinematic chains, $E_iE_{ji}F_{ji}P, (j = 1, 2, i = 1, 2)$. The pose computed by each kinematic chain should be equal to each other and equal to the pose of the moving-platform

$$\mathbf{S}_p = \mathbf{S}_{11} = \mathbf{S}_{16} = \mathbf{S}_{27} = \mathbf{S}_{32} \tag{VI.3}$$

where,

$$\begin{aligned}
\mathbf{S}_{11} &= \mathbf{S}_6\mathbf{S}_7\mathbf{S}_8\mathbf{S}_9\mathbf{S}_{10}\mathbf{S}_{11} \\
\mathbf{S}_{16} &= \mathbf{S}_6\mathbf{S}_{12}\mathbf{S}_{13}\mathbf{S}_{14}\mathbf{S}_{15}\mathbf{S}_{16} \\
\mathbf{S}_{27} &= \mathbf{S}_{22}\mathbf{S}_{23}\mathbf{S}_{24}\mathbf{S}_{25}\mathbf{S}_{26}\mathbf{S}_{27} \\
\mathbf{S}_{32} &= \mathbf{S}_{22}\mathbf{S}_{28}\mathbf{S}_{29}\mathbf{S}_{30}\mathbf{S}_{31}\mathbf{S}_{32}
\end{aligned}$$

The universal joint can be regarded as two orthogonal revolute joints, the pose error termed by the joint clearance in the revolute joint is represented by error screw, and the error screw can be mapped onto the pose error of end-effector in the kinematic chain by ajoin maps, as is shown in Eq. (III.14). And finally, the pose error of the end-effector can be estimated by Eq. (III.20):

Estimated from kinematic chain $E_1 - E_{11} - F_{11} - P$

$$\delta \boldsymbol{p}_{11|F_b} = \sum_{j=7}^{10} \left(\mathbf{N}_{6|F_b} \prod_{l=7}^{11} \mathbf{N}_l \right) \prod_{k=11}^{j+1} \left(\mathtt{adj}(\mathbf{S}_k) \right)^{-1} \delta \boldsymbol{s}_j \tag{VI.4}$$

Estimated from kinematic chain $E_1 - E_{21} - F_{21} - P$

$$\delta \boldsymbol{p}_{16|F_b} = \sum_{j=12}^{15} \left(\mathbf{N}_{6|F_b} \prod_{l=12}^{16} \mathbf{N}_l \right) \prod_{k=16}^{j+1} \left(\mathtt{adj}(\mathbf{S}_k) \right)^{-1} \delta \boldsymbol{s}_j \tag{VI.5}$$

Estimated from kinematic chain $E_2 - E_{12} - F_{12} - P$

$$\delta \boldsymbol{p}_{27|F_b} = \sum_{j=23}^{27} \left(\mathbf{N}_{22|F_b} \prod_{l=23}^{27} \mathbf{N}_l \right) \prod_{k=27}^{j+1} \left(\mathtt{adj}(\mathbf{S}_k) \right)^{-1} \delta \boldsymbol{s}_j \tag{VI.6}$$

Estimated from kinematic chain $E_2 - E_{22} - F_{22} - P$

$$\delta \boldsymbol{p}_{32|F_b} = \sum_{j=28}^{31} \left(\mathbf{N}_{22|F_b} \prod_{l=28}^{32} \mathbf{N}_l \right) \prod_{k=32}^{j+1} \left(\mathtt{adj}(\mathbf{S}_k) \right)^{-1} \delta \boldsymbol{s}_j \tag{VI.7}$$

where, $\mathbf{N}_{6|F_b}$ and $\mathbf{N}_{22|F_b}$ are the maps of the pose expressed in Frame 6 and Frame 22 to the base frame respectively.

Extracting from Eq. (VI.4), Eq. (VI.5), Eq. (VI.6) and Eq. (VI.7), the error mapping matrix can be constructed:

$$\mathbf{M}_{d,11} = \begin{bmatrix} \mathbf{M}_7 & \mathbf{M}_8 & \mathbf{M}_9 & \mathbf{M}_{10} \end{bmatrix} \tag{VI.8}$$

$$\mathbf{M}_{d,21} = \begin{bmatrix} \mathbf{M}_{12} & \mathbf{M}_{13} & \mathbf{M}_{14} & \mathbf{M}_{15} \end{bmatrix} \tag{VI.9}$$

$$\mathbf{M}_{d,12} = \begin{bmatrix} \mathbf{M}_{23} & \mathbf{M}_{24} & \mathbf{M}_{25} & \mathbf{M}_{26} \end{bmatrix} \tag{VI.10}$$

$$\mathbf{M}_{d,22} = \begin{bmatrix} \mathbf{M}_{28} & \mathbf{M}_{29} & \mathbf{M}_{30} & \mathbf{M}_{31} \end{bmatrix} \tag{VI.11}$$

Based on the error mapping matrix, the pose error can be estimated as

$$\delta \boldsymbol{p}_{11|F_b} = \mathbf{M}_{d,11} \delta \boldsymbol{s}_{d,11} \tag{VI.12}$$

$$\delta \boldsymbol{p}_{16|F_b} = \mathbf{M}_{d,21} \delta \boldsymbol{s}_{d,21} \tag{VI.13}$$

$$\delta \boldsymbol{p}_{27|F_b} = \mathbf{M}_{d,12} \delta \boldsymbol{s}_{d,12} \tag{VI.14}$$

$$\delta \boldsymbol{p}_{32|F_b} = \mathbf{M}_{d,22} \delta \boldsymbol{s}_{d,22} \tag{VI.15}$$

where, $\delta \boldsymbol{s}_{d,ji}$, $(j = 1, 2, i = 1, 2)$, is the array that contains 4 error screws for joint clearances.

Certainly, pose errors estimated from different kinematic chains of the distal module are equal to each other

$$\delta \boldsymbol{p} = \delta \boldsymbol{p}_{11|F_b} = \delta \boldsymbol{p}_{16|F_b} = \delta \boldsymbol{p}_{27|F_b} = \delta \boldsymbol{p}_{32|F_b} \tag{VI.16}$$

For finding the maximum pose error, the optimization based method (III.26) and (III.27) with equality constraints can be adopted. But note that the equality constraints sometimes

are hard to be satisfied, it is preferable to use the optimization based method with inequality constraints, as (III.1.4) and (III.43).

So the sensitivity analysis due to joint clearances in distal module of the IRSBot-2 robot becomes an optimization problem with 78 descision variables. It has been implemented in matlab, by using the optimization fuction *fmincon*. But *fmincon* just returns a local minimum, here what we expect is a global minimum, so as to evaluate better of the maximum pose error. In this case, the parallel computation technique is applied, adequate number of start points are defined, each start point can converge to a local minimum. With all the local minimums compared, it is possible to find a global minimum. Obviously, the more start points we use, the best global minimum can be obtained. But regarding to the computation time, we can not choose as many as we can. By checking the convergences of different start points, we found that most of the start points converge to the same local minimum. Thus, just a sufficient number is preferable.

For doing the sensitivity analysis due to joint clearances, the robot is parameterized in Tab. VI.4 and Tab. VI.5, and the tolerances of the joints are shown in Tab. VI.6

Table VI.1 – Parameterization of the IRSBot-2 Robot: Table 1

$b : [mm]$	$d : [mm]$	$\alpha_0 : [rad]$	$\alpha_d : [rad]$	$\beta : [rad]$	$e : [mm]$
50	50	0	$\frac{\pi}{6}$	$\frac{\pi}{6}$	0

Table VI.2 – Parameterization of the IRSBot-2 Robot: Table 2

$l_1 : [mm]$	$l_2 : [mm]$	$a_1 : [mm]$	$a_2 : [mm]$	$p : [mm]$
50	50	30	15	25

Table VI.3 – Tolearance of the Distal Module

$\Delta\beta_{i,j,X,Y} : [rad]$	$\Delta\beta_{i,j,Z} : [rad]$	$\Delta b_{i,j,X,Y} : [mm]$	$\Delta b_{i,j,Z} : [mm]$
0.01	0.2	0.1	0.1

Please note that for the distal module, we have the same qualities for joints. And since in the distal module there is no actuated joints, all the joints are passive, and for sure there will be some idle motions for these passive joints due to the joint clearances. Thus it is given a larger range for the $\Delta\beta_{i,j,Z}$, the value is assigned 0.2, it should be large enough for the idle motion.

Fig. VI.1 shows the maximum positional throughout a Cartesian Space. It can be seen from the top view Fig. VI.1-(b), the maximum positional error of the robot is symmetric about the z-axis, that is because of the symmetry of the robot structure and the symmetry of the Cartesian Space selected to do the analysis. However, even if the maximum positional error is distributed symmetrically about the z-axis, it varies in some points. Especially, it changes quite shaply in some part.

As is shown in Fig. VI.2, the square part is the Cartesian Space where the maximum positional error is analyzed here. Obviously, this area contains some parallel singularities

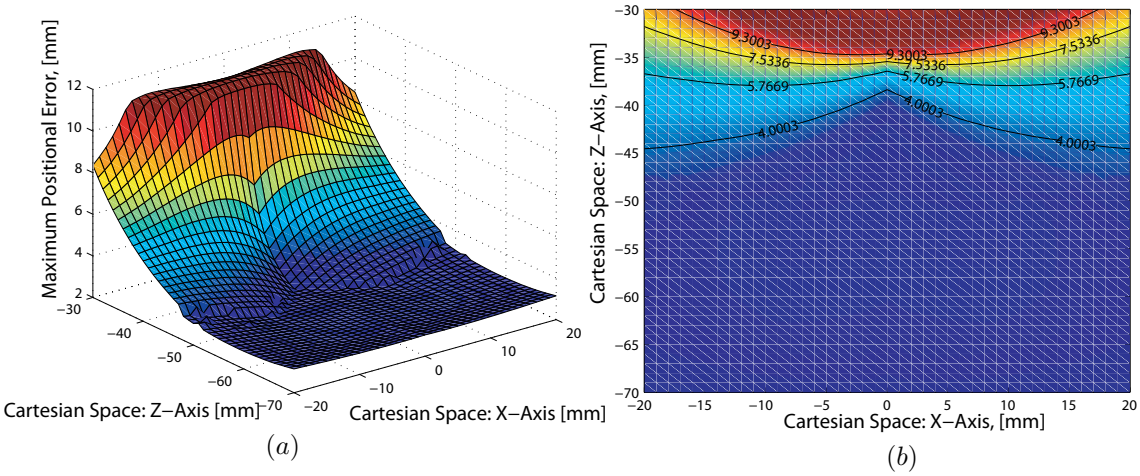


Figure VI.1 – Maximum Positional Error of the IRSBot-2 Robot Throughout A Cartesian Space

points, and there is one parallel singularity configuration plotted in Fig. VI.2. As can be seen from Fig. VI.2, when the robot gets to parallel singularities or in the vicinity of parallel singularities, the positional error becomes quite large, and reaches the maximum when it reaches parallel singularities points.

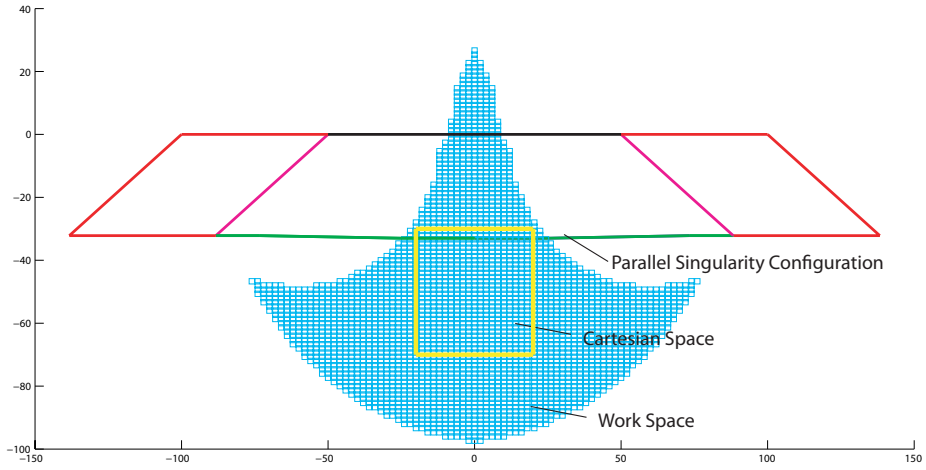
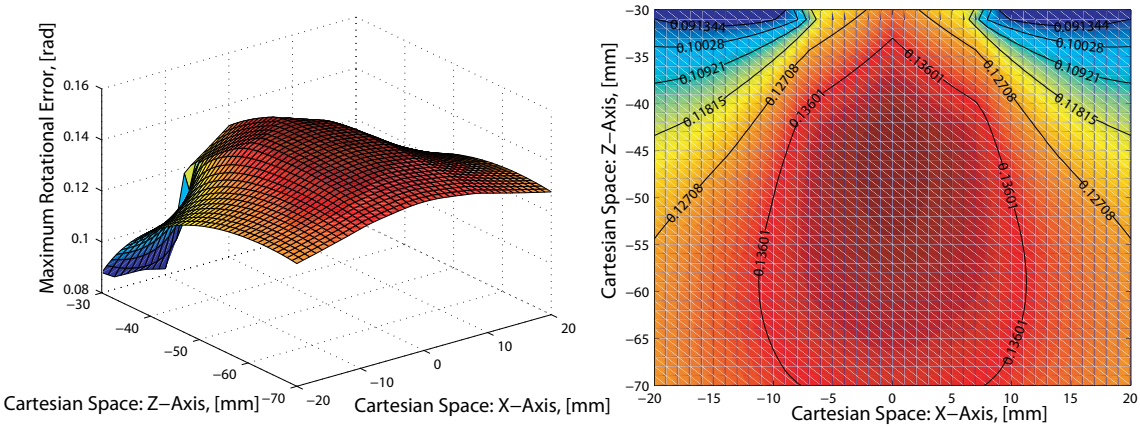


Figure VI.2 – Parallel Singularity Configuration of the IRSBot-2 Robot

Here, the maximum rotational error throughout the same Cartesian Space is also analyzed, as is shown in Fig. VI.3. Obviously, the rotational error throughout the Cartesian Space is also symmetric about the z-axis. But in the vicinity of parallel singularities, the rotational error is not the maximum. It is not necessary to expect that the rotational error should also be large in parallel singularities, because the parallel singularity affects more about the translational motions.



(a) (b)

Figure VI.3 – Maximum Rotational Error of the IRSBot-2 Robot Throughout A Cartesian Space

VI.2 Sensitivity Analysis of the IRSBot-2 Robot to Joint Clearances in Distal Module and Parallelograms

In this section, the joint clearances in parallelograms are also taken into account for doing the sensitivity analysis of the IRSBot-2 robot. Certainly, when there are joint clearances in the parallelogram, the pose error of the end-effector can also be affected. Because the elbows are attached to the end of parallelograms, due to the joint clearances in the parallelograms, the poses of the eblows can be affected and will generate some pose errors. In addition, the distal module connected directly to the end-effector is based on the elbows, if there are some pose errors in the elbows, the errors will be transmitted onto the pose of the end-effector.

Since the IRSBot-2 robot is a spatial robot with 2 hybrid legs, when considering the joint clearances of the parallelograms, the method developed in Chapter II won't be sufficient to do the analysis, it is preferable to use the method developed in Chapter V. The method presented in Chapter V seems to be quite complicated, here with the example of the IRSBot-2 robot, how to apply the method to do the sensitivity analysis due to joint clearances for complicated robots will be illustrated.

VI.2.1 Pose Error Estimation of the Eblows

According to Chapter V, the eblows E_1 and E_2 can be regarded as sub-end-effectors in Leg 1 and Leg 2. There is just 1 sub-end-effector in each leg of the IRSBot-2 robot. So the pose error of the robot estimated from each kinematic from Eq. (VI.12) to Eq. (VI.15) should be added by the influences of the pose errors from the elbows.

$$\delta \boldsymbol{p}_{11|F_b} = \mathbf{M}_{d,11}\delta \boldsymbol{s}_{d,11} + \mathbf{M}^e_{d,11}\delta \boldsymbol{e}_1 \tag{VI.17}$$

$$\delta \boldsymbol{p}_{16|F_b} = \mathbf{M}_{d,21}\delta \boldsymbol{s}_{d,21} + \mathbf{M}^e_{d,21}\delta \boldsymbol{e}_1 \tag{VI.18}$$

$$\delta \boldsymbol{p}_{27|F_b} = \mathbf{M}_{d,12}\delta \boldsymbol{s}_{d,12} + \mathbf{M}^e_{d,12}\delta \boldsymbol{e}_2 \tag{VI.19}$$

$$\delta \boldsymbol{p}_{32|F_b} = \mathbf{M}_{d,22}\delta \boldsymbol{s}_{d,22} + \mathbf{M}^e_{d,22}\delta \boldsymbol{e}_2 \tag{VI.20}$$

where, $\mathbf{M}^{e}_{d,ji}$ is the error mapping matrix for pose error $\delta \boldsymbol{e}_i$, for $j = 1, 2, i = 1, 2$.

The pose error of the sub-end-effector E_i can be estimated by the sub-robot (parallelogram) $A_iB_iC_iD_i$, there are two legs in the parallelogram. And the pose error of E_i can be estimated by the two kinematic chains $O - A_i - B_i - E_i$ and $O - A_i - D_i - C_i - B_i - E_i$

For computating the pose of E_1,

$$\mathbf{S}_{1,6} = \mathbf{S}_1\mathbf{S}_2\mathbf{S}_6 \tag{VI.21}$$

$$\mathbf{S}_{2,6} = \mathbf{S}_1\mathbf{S}_3\mathbf{S}_4\mathbf{S}_5\mathbf{S}_6 \tag{VI.22}$$

where, $\mathbf{S}_{1,6}$ is the pose computed from Leg 1 of the parallelogram, and $\mathbf{S}_{2,6}$ is the pose computed from Leg 2 of the parallelogram. And obviously, $\mathbf{S}_{1,6} = \mathbf{S}_{2,6}$.

Based on the adjoin map defined by Eq. (III.14), the pose error of the sub-end-effector can be estimated from the two kinematic chains in the parallelogram

$$\delta \boldsymbol{e}_{1,1} = \mathbf{M}_{p,11}\delta \boldsymbol{s}_{p,11} \tag{VI.23}$$

$$\delta \boldsymbol{e}_{2,1} = \mathbf{M}_{p,21}\delta \boldsymbol{s}_{p,21} \tag{VI.24}$$

where, $\delta \boldsymbol{s}_{p,11}$ is constructed by the error screws in Joint 1 and Joint 2, $\delta \boldsymbol{s}_{p,21}$ is constructed by the error screws in Joint 3 and Joint 4. The error mapping matrix are constructd as the following

$$\mathbf{M}_{p,11} = \begin{bmatrix} \mathbf{M}_1 & \mathbf{M}_2 \end{bmatrix} \tag{VI.25}$$

$$\mathbf{M}_{p,21} = \begin{bmatrix} \mathbf{M}_3 & \mathbf{M}_4 \end{bmatrix} \tag{VI.26}$$

where,

$$\mathbf{M}_1 = \left(\mathbf{N}_{6|F_b}\right)\left(\mathtt{adj}(\mathbf{S}_6)\right)^{-1}\left(\mathtt{adj}(\mathbf{S}_2)\right)^{-1} \tag{VI.27}$$

$$\mathbf{M}_2 = \left(\mathbf{N}_{6|F_b}\right)\left(\mathtt{adj}(\mathbf{S}_6)\right)^{-1} \tag{VI.28}$$

$$\mathbf{M}_3 = \left(\mathbf{N}_{6|F_b}\right)\left(\mathtt{adj}(\mathbf{S}_6)\right)^{-1}\left(\mathtt{adj}(\mathbf{S}_5)\right)^{-1}\left(\mathtt{adj}(\mathbf{S}_4)\right)^{-1} \tag{VI.29}$$

$$\mathbf{M}_4 = \left(\mathbf{N}_{6|F_b}\right)\left(\mathtt{adj}(\mathbf{S}_6)\right)^{-1}\left(\mathtt{adj}(\mathbf{S}_5)\right)^{-1} \tag{VI.30}$$

Obviously, the error pose estimated from the two kinematic chain are equal to each other

$$\delta \boldsymbol{e}_1 = \delta \boldsymbol{e}_{1,1} = \delta \boldsymbol{e}_{2,1} \tag{VI.31}$$

Similarly, the pose error of Elbow 2 can be estimated as

$$\delta \boldsymbol{e}_{1,2} = \mathbf{M}_{p,12}\delta \boldsymbol{s}_{p,12} \tag{VI.32}$$

$$\delta \boldsymbol{e}_{2,2} = \mathbf{M}_{p,22}\delta \boldsymbol{s}_{p,22} \tag{VI.33}$$

And,

$$\delta \boldsymbol{e}_2 = \delta \boldsymbol{e}_{1,2} = \delta \boldsymbol{e}_{2,2} \tag{VI.34}$$

VI.2.2 Maximum Pose Error of the End-effector

The aim here is to find the maximum pose error of the end-effector with joint clearances both in the distal module and the parallelograms for the IRSBot-2 Robot. Based on the error estimation model of the IRSBot-2 robot developed in this chapter, an optimization based method developed in Chapter V for finding the maximum pose error is applied here. The optimization method with inequality constraints is adopted here, and has been employed for doing the sensitivity analysis of the IRSBot-2 robot due to joint clearances.

First of all, we would like to clarify some parameters for the IRSBot-2 robot. There are two hybrid legs, thus $m = 2$. For Leg 1, there is 1 sub-end-effector E_i, and thus $n_1 = 1$, $n_2 = 1$, the sub-robots in Leg 1 and Leg 2 contain $m_{1,1} = 2$ and $m_{1,1} = 2$ sub-legs separately. The end-effector is connected to sub-end-effector $E - 1$ by $m_{1,p} = 2$ legs, and connected to sub-end-effector E_2 by $m_{2,p} = 2$ sub-legs. There are $n_{1,p,l} = 4, (l = 1, 2)$, joints in the sub leg that connected from the end-effector to E_1. Symmetrically, there are also $n_{2,p,l} = 4(l = 1, 2)$ joints in the sub-leg connected from the end-effector to E_2. In the parallelograms (sub-robots), there are $n_{1,1,l} = 2(l = 1, 2)$ joints in sub-robot of Leg 1, and symmetrically there are also $n_{2,1,l} = 2(l = 1, 2)$ joints in sub-robot of Leg 2.

The number of decision variables for used for doing the optimization for the IRSBot-2 robot can be counted by Eq. (V.37), Eq. (V.35) and Eq. (V.20).

In terms of Eq. (V.35) and Eq. (V.20), the number of decision variables corresponding to the sub-robots and the upper part of the robot, where the end-effector is connected to the sub-end-effectors

$$v_{e_{1,1}} = 6 + 6 \times [(2-1) + (2-1)] = 18 \tag{VI.35}$$

$$v_{e_{2,1}} = 6 + 6 \times [(2-1) + (2-1)] = 18 \tag{VI.36}$$

$$v_p = 6 + 6 \times (4-1) \times 4 = 78 \tag{VI.37}$$

Thus, the number of the decision variables in terms of Eq. (V.37) is

$$v = v_p + v_{e_{1,1}} + v_{e_{2,1}} = 114 \tag{VI.38}$$

So the decision variables can be defined in an array as the following

$$\delta \boldsymbol{u} = [\underbrace{\delta \boldsymbol{p}^T, \delta \boldsymbol{q}_{d,11,2}^T, \ldots, \delta \boldsymbol{q}_{d,22,2}^T}_{v_p=78}, \underbrace{\delta \boldsymbol{e}_1^T, \delta \boldsymbol{q}_{p,11,2}^T, \delta \boldsymbol{q}_{p,21,2}^T}_{v_{e_{1,1}}=18}, \underbrace{\delta \boldsymbol{e}_2^T, \delta \boldsymbol{q}_{p,12,2}^T, \delta \boldsymbol{q}_{p,22,2}^T}_{v_{e_{2,1}}=18}]^T \tag{VI.39}$$

Correspondingly, the decision variables for the distal module can be extracted as $\delta \boldsymbol{u}_d$, the variables for the parallelogram in Leg 1 can be extracted as $\delta \boldsymbol{u}_{p,1}$, and the variables for the parallelogram in Leg 2 can be extracted as $\delta \boldsymbol{u}_{p,2}$.

Based on the QR factorization methodology presented in Chapter III and Chapter V, the space of decision variables can be mapped to the error screw space by a $\boldsymbol{\Upsilon}$ matrix, as the following

For the parallelogram in Leg 1,

$$\delta \boldsymbol{s}_{p,11} = \Upsilon_{p,11}\delta \boldsymbol{u}_{p,1} \quad \text{(VI.40)}$$

$$\delta \boldsymbol{s}_{p,21} = \Upsilon_{p,21}\delta \boldsymbol{u}_{p,1} \quad \text{(VI.41)}$$

For the parallelogram in Leg 2,

$$\delta \boldsymbol{s}_{p,12} = \Upsilon_{p,12}\delta \boldsymbol{u}_{p,2} \quad \text{(VI.42)}$$

$$\delta \boldsymbol{s}_{p,22} = \Upsilon_{p,22}\delta \boldsymbol{u}_{p,2} \quad \text{(VI.43)}$$

For the distal module,

$$\delta \boldsymbol{s}_{d,11} = \Upsilon_{d,11}\delta \boldsymbol{u}_{d} \quad \text{(VI.44)}$$

$$\delta \boldsymbol{s}_{d,21} = \Upsilon_{d,21}\delta \boldsymbol{u}_{d} \quad \text{(VI.45)}$$

$$\delta \boldsymbol{s}_{d,12} = \Upsilon_{d,12}\delta \boldsymbol{u}_{d} \quad \text{(VI.46)}$$

$$\delta \boldsymbol{s}_{d,22} = \Upsilon_{d,22}\delta \boldsymbol{u}_{d} \quad \text{(VI.47)}$$

According to Section V.3, the matrix $\Upsilon_{p,ji}$ and $\Upsilon_{d,ji}$ are constructed based on the QR factorization of the transpose of the error mapping matrix $\mathbf{M}_{p,ji}^T$ and $\mathbf{M}_{d,ji}^T$ respectively, where, $j = 1, 2$ and $i = 1, 2$.

With the decision variable $\delta \boldsymbol{u}$ and the Υ matrix, the equality constraints can be integrated into the inequality, and therefore an optimization problem for finding the maximum error pose with only inequality constraints is developed for the IRSBot-2 robot.

The objective function for finding the maximum positional error is defined as

$$-p_{max}^2 = -\delta \boldsymbol{u}^T \mathbf{F}_0 \delta \boldsymbol{u} \quad \text{(VI.48)}$$

where,

$$\mathbf{F}_0 = \sum_{l=4,5,6} \delta \boldsymbol{e}_{v,l} \delta \boldsymbol{e}_{v,l}^T$$

And similiarly, the objective function for finding the maximum rotational error is defined as

$$-r_{max}^2 = -\delta \boldsymbol{u}^T \mathbf{F}_0 \delta \boldsymbol{u} \quad \text{(VI.49)}$$

where,

$$\mathbf{F}_0 = \sum_{l=1,2,3} \delta \boldsymbol{e}_{v,l} \delta \boldsymbol{e}_{v,l}^T$$

Assuming the tolerances in the parallelograms are : $\Delta\beta_{p,ji,k,X,Y}$, $\Delta\beta_{p,ji,k,Z}$, $\Delta b_{p,ji,k,X,Y}$ and $\Delta b_{p,ji,k,Z}$, where $j = 1, 2$ is the index of the kinematic chain in the parallelogram,

$i = 1, 2$ is the index of the leg of the IRSBot-2 robot, $k = 1, 2$, is the index of the joint in the kinematic chain. Similarly, the tolerances in the distal module are defined as: $\Delta\beta_{d,ji,k,X,Y}$, $\Delta\beta_{d,ji,k,Z}$, $\Delta b_{d,ji,k,X,Y}$ and $\Delta b_{d,ji,k,Z}$, but here $k = 1, 2, 3, 4$, because in the distal module each kinematic chain contains 4 revolute joints.

The constraints should be first satisfied with the sub-robots, because the error estimation model for the end-effector is based on the pose errors of the sub-end-effector. Here, the constraints for sub-robot 1 and sub-robot 2 are first given, and should be checked first in the program.

For $\delta\boldsymbol{u}_{p,i}$, it should meet the constraints as the following:

$$\begin{aligned} f_{p,ji,k,l} &= \delta\boldsymbol{u}_{p,i}^T \mathbf{F}_{p,ji,k,l} \delta\boldsymbol{u}_{p,ji,k,l} - 1 \leqslant 0 \\ & l = 1, \ldots, 4, k = 1, 2 \\ & j = 1, 2, i = 1, 2 \end{aligned} \tag{VI.50}$$

where,

$$\begin{aligned} \mathbf{F}_{p,ji,k,1} &= (1/\Delta\beta_{p,ji,k,X,Y}^2)\Upsilon_{p,ji}^T(\boldsymbol{e}_{6n_{p,ji},6k-5}\boldsymbol{e}_{6n_{p,ji},6k-5}^T + \boldsymbol{e}_{6n_{p,ji},6k-4}\boldsymbol{e}_{6n_{p,ji},6k-4}^T)\Upsilon_{p,ji} \\ \mathbf{F}_{p,ji,k,2} &= (1/\Delta\beta_{p,ji,k,Z}^2)\Upsilon_{p,ji}^T(\boldsymbol{e}_{6n_{p,ji},6k-3}\boldsymbol{e}_{6n_{p,ji},6k-3}^T)\Upsilon_{p,ji} \\ \mathbf{F}_{p,ji,k,3} &= (1/\Delta b_{p,ji,k,X,Y}^2)\Upsilon_{p,ji}^T(\boldsymbol{e}_{6n_{p,ji},6k-2}\boldsymbol{e}_{6n_{p,ji},6k-2}^T + \boldsymbol{e}_{6n_{p,ji},6k-1}\boldsymbol{e}_{6n_{p,ji},6k-1}^T)\Upsilon_{p,ji} \\ \mathbf{F}_{p,ji,k,4} &= (1/\Delta b_{p,ji,k,Z}^2)\Upsilon_{p,ji}^T(\boldsymbol{e}_{6n_{p,ji},6j}\boldsymbol{e}_{6n_{p,ji},6j}^T)\Upsilon_{p,ji} \end{aligned}$$

where, i stands for the number of leg of the IRSBot-2 robot, j is the number of the sub-leg in the parallelogram, k is the number of joint in the sun-leg, $n_{p,ji}$ is the number of joints in the sub-leg j of the parallelogram in Leg i, and obviously $n_{p,ji} = 2$.

The constraints of the distal module is a bit different. From Eq. (VI.12) to Eq. (VI.15), the pose error of the end-effector is not only dependant on the error screws modeled for the joint clearances in the kinematic chains of the distal module, but also dependant on the pose errors of the sub-end-effectors connected by sub-legs to the end-effector. In addition, the matrix $\Upsilon_{d,ji}$ is derived by the QR factorization of the transpose of the error mapping matrix $\mathbf{M}_{d,ji}^T$. Some adaption has to be done for using the inequality constraints properly.

$$\underbrace{\delta\boldsymbol{p} - \mathbf{M}_{d,ji}^e \delta\boldsymbol{e}_i}_{\delta\boldsymbol{q}_{d,ji,p}} = \mathbf{M}_{d,ji}\delta\boldsymbol{s}_{d,ji} \tag{VI.51}$$

Both $\delta\boldsymbol{p}$ and $\delta\boldsymbol{e}_i$ are extracted from $\delta\boldsymbol{u}$, $\delta\boldsymbol{e}_i$ is part of $u_{p,i}$, and is constrained by the inequality constraint (VI.50). For the constraint of $\delta\boldsymbol{p}$ and $\delta\boldsymbol{q}_{d,ji,2}^T$, $\delta\boldsymbol{u}_{d,ji}$ is constructed here

$$\delta\boldsymbol{u}_{d,ji} = [\delta\boldsymbol{q}_{d,ji,p}^T, \delta\boldsymbol{q}_{d,11,2}^T, \ldots, \delta\boldsymbol{q}_{d,22,2}^T]^T \in \mathbb{R}^{v_p} \tag{VI.52}$$

where, $\delta\boldsymbol{q}_{d,ji,p} = \delta\boldsymbol{p} - \mathbf{M}_{d,ji}^e\delta\boldsymbol{e}_i$.

Since $\mathbf{M}_{d,ji}^e\delta\boldsymbol{e}_i$ varies from leg to leg, the first 6 components of $\delta\boldsymbol{u}_{d,ji}$ also varies, but $\delta\boldsymbol{u}_{d,ji}$ should be constrained as the following

$$f_{d,ji,k,l} = \delta \boldsymbol{u}_{d,ji}^T \mathbf{F}_{d,ji,k,l} \delta \boldsymbol{u}_{d,ji,k,l} - 1 \leqslant 0 \quad \text{(VI.53)}$$
$$l = 1, \ldots, 4, k = 1, 2$$
$$j = 1, 2, i = 1, 2$$

where,

$$\mathbf{F}_{d,ji,k,1} = (1/\Delta\beta_{d,ji,k,X,Y}^2)\Upsilon_{d,ji}^T(\boldsymbol{e}_{6n_{d,ji},6k-5}\boldsymbol{e}_{6n_{d,ji},6k-5}^T + \boldsymbol{e}_{6n_{d,ji},6k-4}\boldsymbol{e}_{6n_{d,ji},6k-4}^T)\Upsilon_{d,ji}$$
$$\mathbf{F}_{d,ji,k,2} = (1/\Delta\beta_{d,ji,k,Z}^2)\Upsilon_{d,ji}^T(\boldsymbol{e}_{6n_{d,ji},6k-3}\boldsymbol{e}_{6n_{d,ji},6k-3}^T)\Upsilon_{d,ji}$$
$$\mathbf{F}_{d,ji,k,3} = (1/\Delta b_{d,ji,k,X,Y}^2)\Upsilon_{d,ji}^T(\boldsymbol{e}_{6n_{d,ji},6k-2}\boldsymbol{e}_{6n_{d,ji},6k-2}^T + \boldsymbol{e}_{6n_{d,ji},6k-1}\boldsymbol{e}_{6n_{d,ji},6k-1}^T)\Upsilon_{d,ji}$$
$$\mathbf{F}_{d,ji,k,4} = (1/\Delta b_{d,ji,k,Z}^2)\Upsilon_{d,ji}^T(\boldsymbol{e}_{6n_{d,ji},6j}\boldsymbol{e}_{6n_{d,ji},6j}^T)\Upsilon_{d,ji}$$

In the IRSBot-2 robot, the number of joints in the kinematic chain $E_{ji}F_{ji}$ in distal module is $n_{d,ji} = 4$

VI.3 Comparison of the sensitivity analysis due to Joint Clearances

The section aims to compare the sensitivity analysis results of the IRSBot-2 robot by considering with and without joint clearances in the parallelogram. Intuitively, when considering the joint clearances in the parallelograms, there will be more error sources, and the maximum pose error of the end-effector will be larger.

Note that, the number of decision variables for considering the parallelograms is 114, and 78 without considering the parallelograms. Thus, it is a huge optimization problem. Hence, making the objective to converge is very time consuming. If we do the comparision throughout a Cartesian Space, it will take several days for the work station to finish computing. The results in Fig. VI.1 and Fig. VI.3 presented in section VI.1 took ten days to do the computation. To save the computation time while without losing the generality, the sensitivity corresponding to the four corners and the center of the maximum rectangle inside the work sapce is analyzed here.

For doing the comparision, the IRSBot-2 is parameterized as the following:

Table VI.4 – Parameterization of the IRSBot-2 Robot for Comparison: Table 1

$b : [mm]$	$d : [mm]$	$\alpha_0 : [rad]$	$\alpha_d : [rad]$	$\beta : [rad]$	$e : [mm]$
50	50	0	$\frac{\pi}{6}$	$\frac{\pi}{4}$	0

Table VI.5 – Parameterization of the IRSBot-2 Robot for Comparison: Table 2

$l_1 : [mm]$	$l_2 : [mm]$	$a_1 : [mm]$	$a_2 : [mm]$	$p : [mm]$
50	50	30	15	25

Joint clearances in the distal module for both of the sensitivity analysis are same. Joints in the distal module are all passive joints. Thus, $\Delta\beta_{d,ji,k,Z}$ is given a larger range by considering the possible idle motions in the joints, here it is defined as $\Delta\beta_{d,ji,k,Z} = 0.4rad$, it should be enough for the idle motion.

Table VI.6 – Tolearance of the Distal Module

$\Delta\beta_{d,ji,k,X,Y} : [rad]$	$\Delta\beta_{d,ji,k,Z} : [rad]$	$\Delta b_{d,ji,k,X,Y} : [mm]$	$\Delta b_{d,ji,k,Z} : [mm]$
0.01	0.4	0.1	0.1

When considering the joint clearances in the parallelograms, there exist actuated joint in each parallelogram. And for doing the sensitivity analysis, normally the actuated joints are regarded as locked. Thus, a very small rotation range is given to the actuated joint.

Table VI.7 – Tolearance of the Actuated Joint in Leg 1

$\Delta\beta_{p,11,1,X,Y} : [rad]$	$\Delta\beta_{p,11,1,Z} : [rad]$	$\Delta b_{p,11,1,X,Y} : [mm]$	$\Delta b_{p,11,1,Z} : [mm]$
0.01	0.001	0.1	0.1

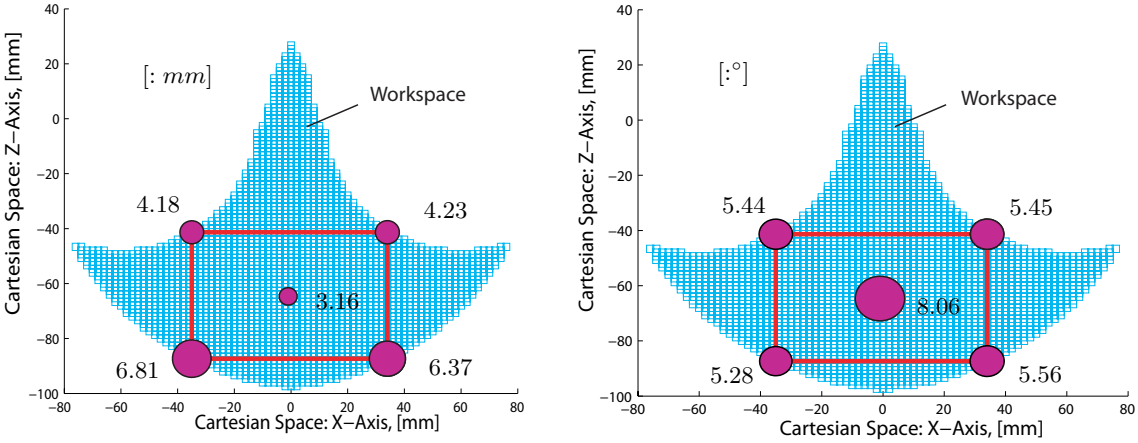
Table VI.8 – Tolearance of the Passive Joint in Parallelograms

$\Delta\beta_{p,12,1,X,Y} : [rad]$	$\Delta\beta_{p,12,1,Z} : [rad]$	$\Delta b_{p,12,1,X,Y} : [mm]$	$\Delta b_{p,12,1,Z} : [mm]$
0.01	0.001	0.1	0.1

Fig. VI.4 shows sensitivity analysis results of just considering joint clearances in the distal modules. Fig. VI.5 shows the sensitivity analysis results of considering joint clearances in the parallelogram and the distal module. Obviously, the positional errors and the rotational errors in Fig. VI.5 are larger than those in Fig. VI.4. The red rectangle in the figure is the maximum rectangle inside the workspace. The points chosen for doing the analysis are in the four corners and in the center, they can more or less represent the distribution of the sensitivity due to the joint clearances throughout the rectangular Cartesian Space.

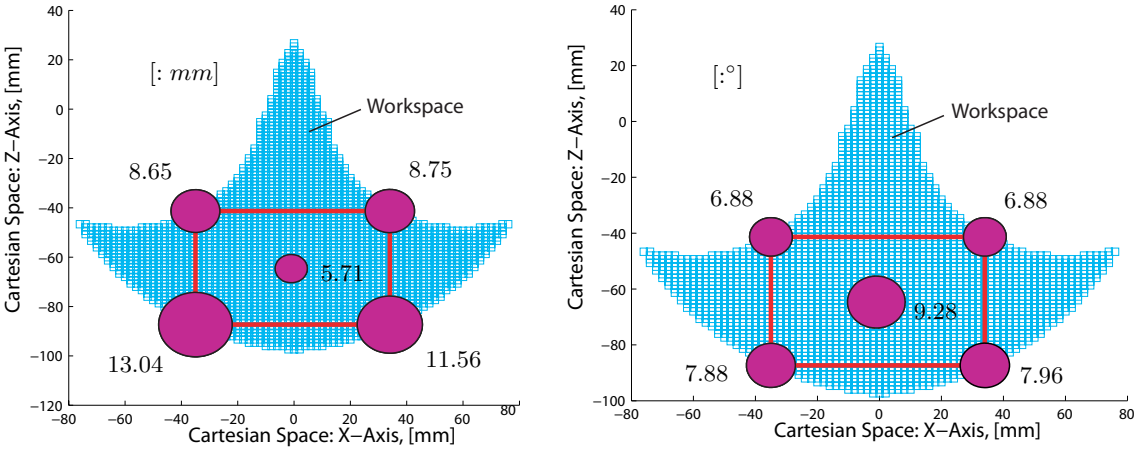
Table VI.9 – Tolearance of the Actuated Joint in Leg 2

$\Delta\beta_{p,ji,i,X,Y} : [rad]$	$\Delta\beta_{p,ji,i,Z} : [rad]$	$\Delta b_{p,ji,i,X,Y} : [mm]$	$\Delta b_{p,ji,i,Z} : [mm]$
0.01	0.2	0.1	0.1



(*a*) Positional Error (*b*) Rotational Error

Figure VI.4 – Pose Error of the Largest Cuboid-shaped Sub-workspace for IRSBot-2 Robot with Considering Joint Clearances in the Distal Module



(*a*) Positional Error (*b*) Rotational Error

Figure VI.5 – Pose Error of the Largest Cuboid-shaped Sub-workspace for IRSBot-2 Robot with Considering Joint Clearances in the Parallelogram and the Distal Module

VII

Tolerance Synthesis and Dimension Synthesis

The sensitivities due to joint clearances for 5-bar linkage and IRSBot-2 robot are studied in Chapter IV and Chapter VI. Knowing the dimensions of the robot and its tolerances in joint clearances, the maximum positional error and rotational error of the end-effector can be estimated based on the sensitivity analysis model. But sometimes we don't actually know the tolerances in joint clearances or dimensions of the robot. To design a robot with certain accuracy, the tolerances in joint clearances and the dimensions of the robot should be properly designed to meet the accuracy requirements.

This chapter is to study the tolerance synthesis and the dimension synthesis methods for robot designs. Accuracy is a very important performance for robots, we want to increase the accuracy, but at the same time we also want to control the cost of the robot. Thus befor manufacturing the robot, we need to figure out which tolerances and dimensions affect the accuracy of the robot more, and which tolerances and dimensions affect the accuracy less. For those affect more, more attentions should be paid when doing the manufacturing. In this chapter, Section VII.1 presents a tolerance synthesis method for robots, and the 5-bar linkage is studied. Section VII.2 presents a dimension synthesis method for robots, and the IRSBot-2 robot is studied.

VII.1 Tolerance Synthesis

VII.1.1 Tolerance Synthesis Method

Tolerance Synthesis is to define the tolerances of the joint clearances for robots, so as to reach a required accuracy. Unlike the sensitivity analysis, tolerance synthesis is an inverse problem. As presented in Section III.1, tolerances of joint clearances are defined by: $\Delta\beta_{i,j,X,Y}$, $\Delta\beta_{i,j,Z}$, $\Delta b_{i,j,X,Y}$ and $\Delta b_{i,j,Z}$, as shown in Fig. III.1. Among the four tolerances, $\Delta\beta_{i,j,X,Y}$ is dependant on $\Delta b_{i,j,X,Y}$ if the length of the joint is known. Here Eq. (III.2) is recalled:

$$\Delta\beta_{i,j,X,Y} = \frac{\Delta b_{i,j,X,Y}}{h}$$

where, h is the length of the joint.

Tolerances $\Delta b_{i,j,X,Y}$ and $\Delta b_{i,j,Z}$ depend on the manufacturing quality. $\Delta\beta_{i,j,Z}$ is the rotation along the joint axis. Generally for actuated joints, joints are regarded as locked by

motor. And this parameter should be assigned a very small value or zero. For passive joints, passive joints will gain some idle motions due to the joint clearances even if the actuated joints are locked. Regarding to the possible idle motions, $\Delta\beta_{i,j,Z}$ is generally assigned a relatively higher value when doing the sensitivity analysis.

Therefore, there are just two independant variables: $\Delta b_{i,j,X,Y}$ and $\Delta b_{i,j,Z}$. For doing tolerance synthesis, $\Delta b_{i,j,X,Y}$ and $\Delta b_{i,j,Z}$ should be defined in terms of the required accuracy of the robot. Generally for the accuracy of the robot, positional error and the rotational error of the robot should be contrained. But in case of some other requirements, a multi-objective problem is defined here, as shown in Fig. VII.1.

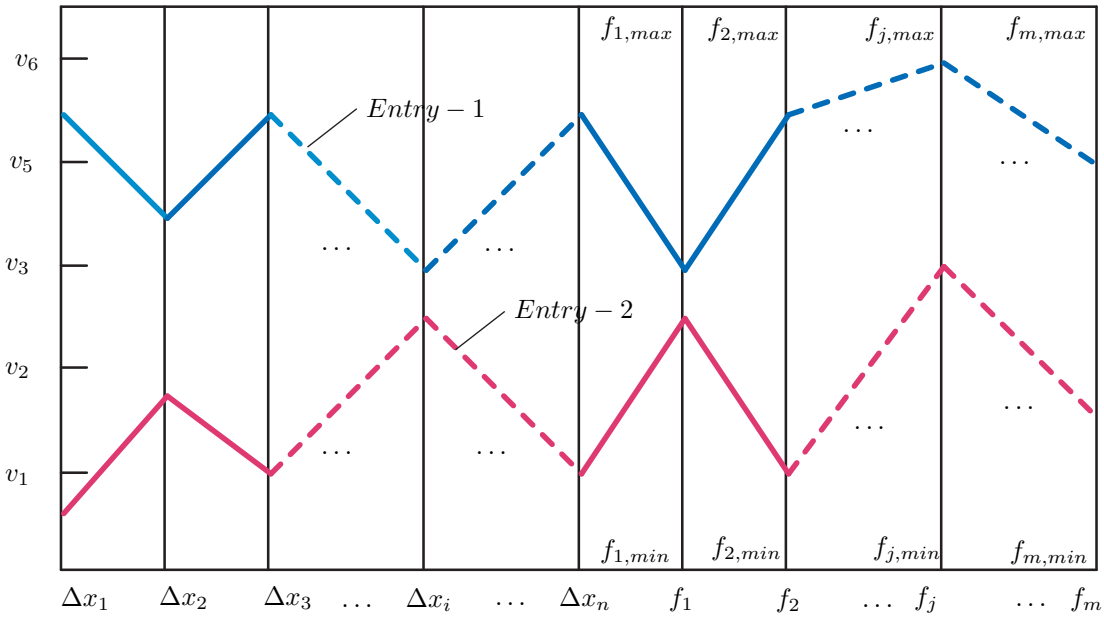


Figure VII.1 – Tolerance Synthesis Model

In Fig. VII.1, variables are tolerances: $\Delta x_1, \Delta x_2, \Delta x_3, \ldots, \Delta x_i, \ldots, \Delta x_n$, and the objective functions are: $f_1, f_2, \ldots, f_j, \ldots, f_m$. The values of the objectives are mapped to the same scales with a range of $[f_{j,min}, f_{j,max}]$. Tolerances are assigned different levels of values, such as $v_1, v_2, \ldots, v_6$. Entries are plotted by lines, such as Entry-1 and Entry-2 shown in the figure. Obviously, it is quite convenient to see the corresponding values of the objectives with some certain values in tolerances.

Based on Fig. VII.1, entries with different levels of values can be plotted in the same figure. By filtering from the objectives, it is possiple to see which tolerances affect which objectives more, and which affect which less. Therefore, if tolerances have to be defined to reach some values of objectives, the designer can refer to this figure, so as to have an idea about how to define tolerances.

VII.1.2 Tolerance Synthesis of the 5-bar linkage

5-bar linkage has 5 joints totally, including 2 active joints and 3 passive joints. According to Section VII.1, totally there will be 5 $\Delta b_{i,j,X,Y}$ and 5 $\Delta b_{i,j,Z}$ for tolerances when we come to the problem of tolerance synthesis. The objectives are the maximum positional error and

maximum rotational error.

For obtaining a figure to do the tolerance synthesis of the 5-bar linkage, the parameterizations of the 5-bar linkage shown in Tab. IV.2 is adopted. Four levels of tolerances for $\Delta b_{1,1,X,Y}$, $\Delta b_{1,2,X,Y}$, $\Delta b_{2,1,X,Y}$, $\Delta b_{2,2,X,Y}$, $\Delta b_{2,3,X,Y}$ are defined: 10^{-4}, 10^{-3}, 10^{-2} and 10^{-1}. Two levels of tolerances for $\Delta b_{1,1,Z}$, $\Delta b_{1,2,Z}$, $\Delta b_{2,1,Z}$, $\Delta b_{2,2,Z}$, $\Delta b_{2,3,Z}$ are defined: 10^{-3} and 10^{-1}. The position of the end-effector is $(0, 70)$. In order to see how different levels of tolerances affect the maximum positional error and rotational error of the robot, 1000 entries are plotted based on Fig. VII.1, shown in Fig. VII.2.

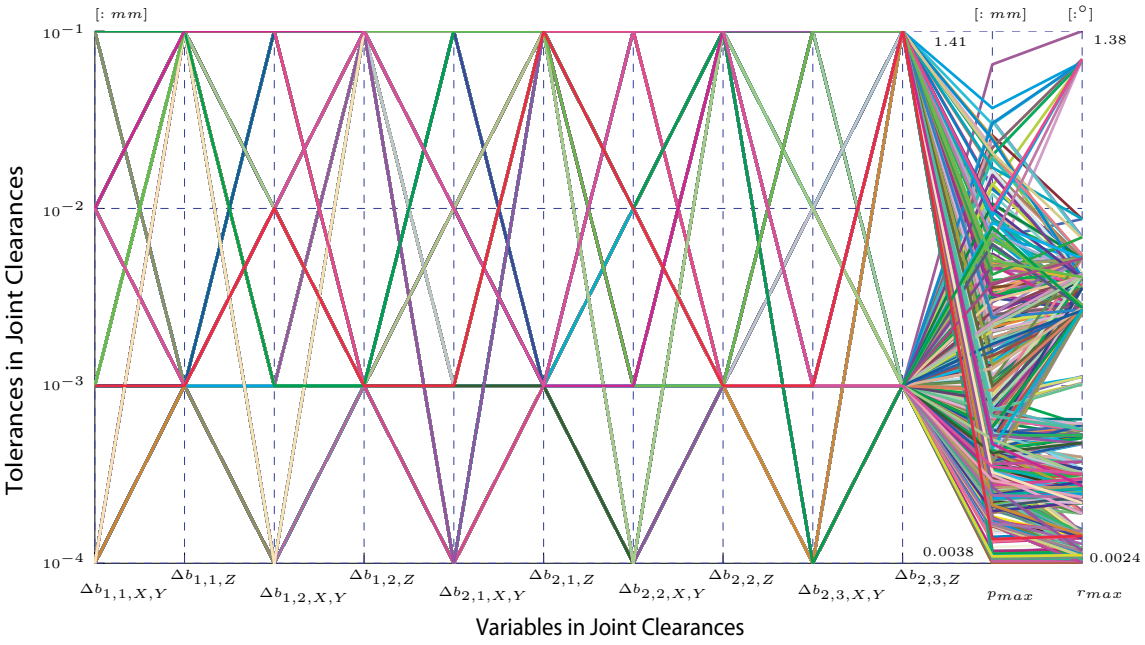


Figure VII.2 – Tolerance Synthesis of the 5-bar Linkage

It can be seen from Fig. VII.2, the range of the maximum positional error is $[0.0038mm, 1.41mm]$, the range of the maximum rotational error is $[0.0024°, 1.38°]$. But It is not so straightforward to see how the tolerances affect the positional and rotational error of the end-effector from Fig. VII.2. Because there are too many entries. For better knowing of the influences of the tolerances on pose errors, those entries are filtered.

The entries are filtered by $p_{max} \geqslant 1mm$, shown in Fig. VII.3. As can be seen from Fig. VII.3, to reach a high positional error, the tolerances of $\Delta b_{1,1,Z}$ and $\Delta b_{2,1,Z}$ stay in the level of $10^{-1}mm$. Both $\Delta b_{1,1,Z}$ and $\Delta b_{2,1,Z}$ are the tolerances of the actuated joints.

At the same time, we are also quite curious to see what if we filter the entries with small maximum positional error. Let's choose the entries with $p_{max} \leqslant 0.05mm$, as shown in Fig. VII.4.

From Fig. VII.4, we can see that to reach such high accuracy up to $0.05mm$, tolerances of all the $\Delta b_{i,j,X,Y}$ are less than $10^{-2}mm$. No matter if it is active joint or not. But for $\Delta b_{i,j,Z}$, it can either $10^{-3}mm$ or $10^{-1}mm$. It seems $\Delta b_{i,j,Z}$ does not affect the positional accuracy much.

Furthermore, we would like to decrease the range of filtering the entries. Here the entries with $p_{max} \leqslant 0.01mm$ are filtered, as shown in Fig. VII.5.

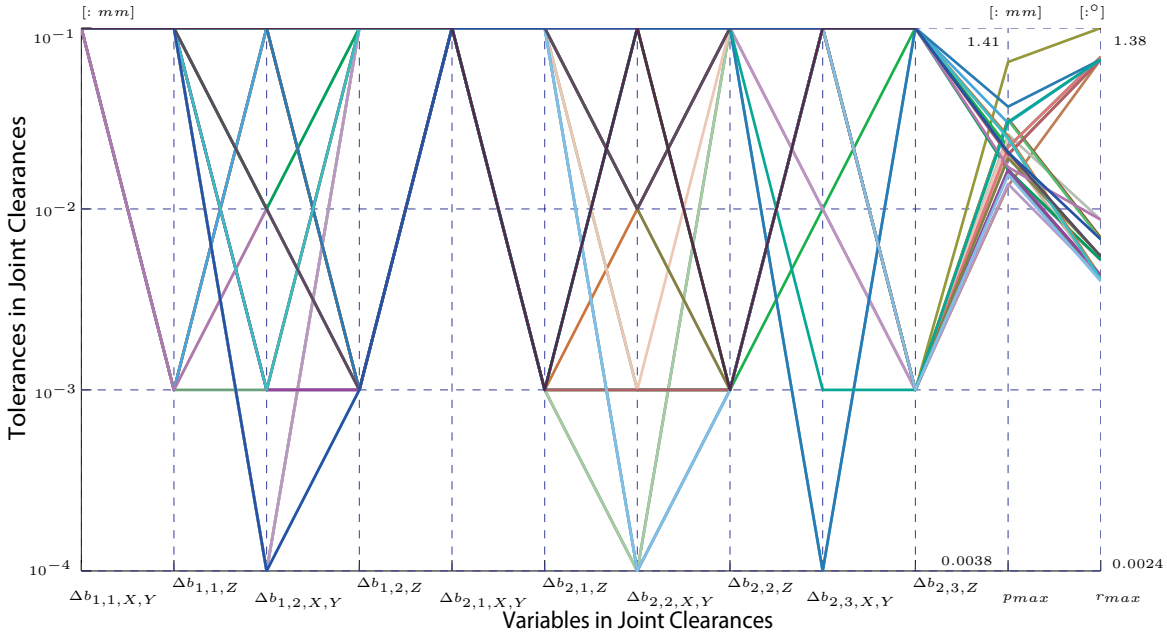


Figure VII.3 – Filtered Entries with Maximum Positional Errors Larger Than 1 mm

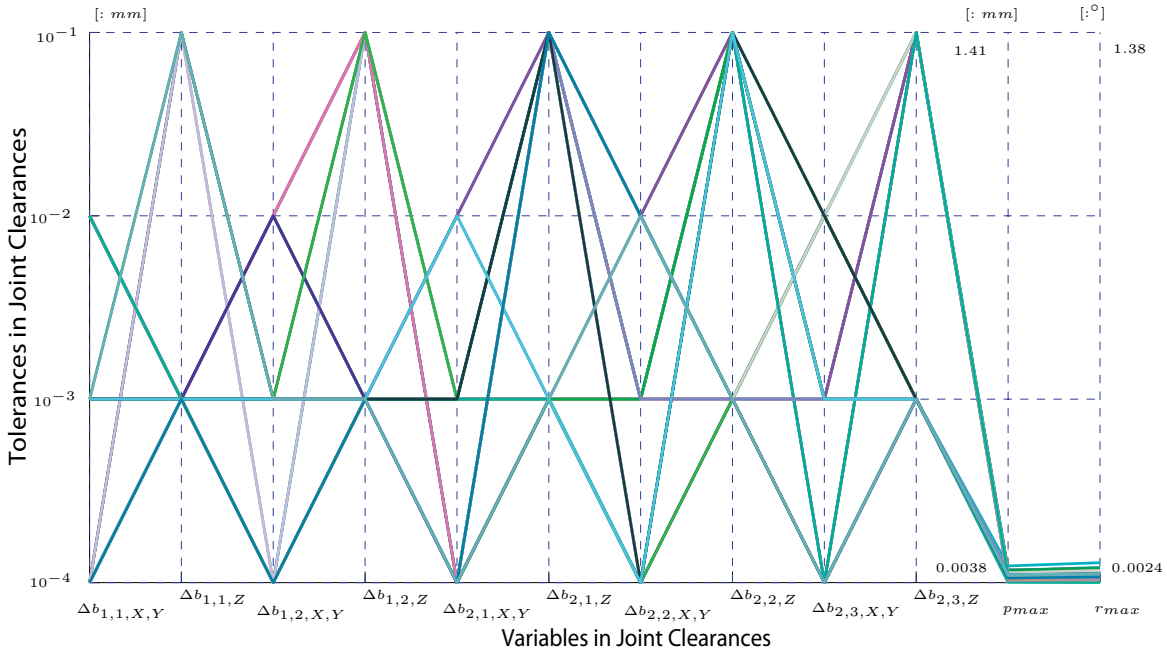


Figure VII.4 – Filtered Entries with Maximum Positional Errors Less Than 0.05 mm

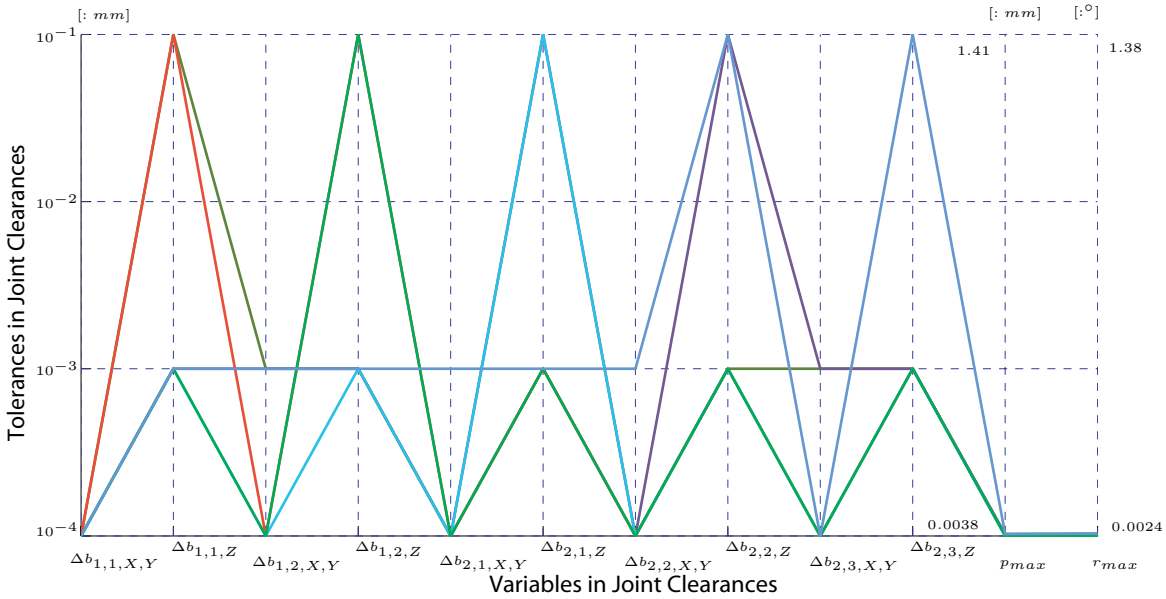


Figure VII.5 – Filtered Entries with Maximum Positional Error Less Than 0.01 mm

From Fig. VII.5, we can see that to reach such a high accuracy up to $0.01mm$, both the values of $\Delta b_{1,1,X,Y}$ and $\Delta b_{2,1,X,Y}$ keep $10^{-4}mm$. Values of $\Delta b_{i,j,X,Y}$ for passive joints can either be $10^{-4}mm$ or $10^{-3}mm$, but no greater than $10^{-3}mm$. As for the values of $\Delta b_{i,j,Z}$, it can still be either $10^{-3}mm$ or $10^{-1}mm$.

Based on the results obtained from Fig. VII.3, Fig. VII.4 and Fig. VII.5. It can be concluded that the tolerances of $\Delta b_{i,j,Z}$ for the joints either active or passive do not affect much of the positioning accuracy of the 5-bar linkage. But the tolerances of $\Delta b_{i,j,X,Y}$ affect much of the accuracy. Especially the active joints, to reach a high positional error, the tolerances of $\Delta b_{i,j,X,Y}$ for active joints should be high. While to reach a very low positional error, the tolerances of $\Delta b_{i,j,X,Y}$ for active joints should be quite low. Comparing with the $\Delta b_{i,j,X,Y}$ for passive joints, the active joints affect more.

In the same way, entries can be also filtered by the values of maximum rotational errors. Entries filtered by $r_{max} \geqslant 1°$ are shown in Fig. VII.6, entries filtered by $r_{max} \leqslant 0.05°$ are shown in Fig. VII.7, entries filtered by $r_{max} \leqslant 0.01°$ are shown in Fig. VII.8.

From Fig. VII.6, we can see that to reach a high rotational error, it seems that Leg 1 dominates the rotational error. $\Delta b_{1,1,X,Y}$ and $\Delta b_{1,2,X,Y}$ both stay $10^{-1}mm$. From Fig. VII.7 and Fig. VII.8, we can see that to reach a high accuracy, all the tolerances of $\Delta b_{i,j,X,Y}$ should be constrained within small values. Tolerances of $\Delta b_{i,j,Z}$ for active joints or passive joints do not affect much of the rotational accuracy.

VII.2 Dimension Synthesis

VII.2.1 Dimension Synthesis Method

Dimension synthesis is to find dimensions of the robot based on required size of workspace and accuracy. Like the tolerance synthesis, dimension synthesis is an inverse problem. For doing dimension synthesis, a multi-objective problem can be defined.

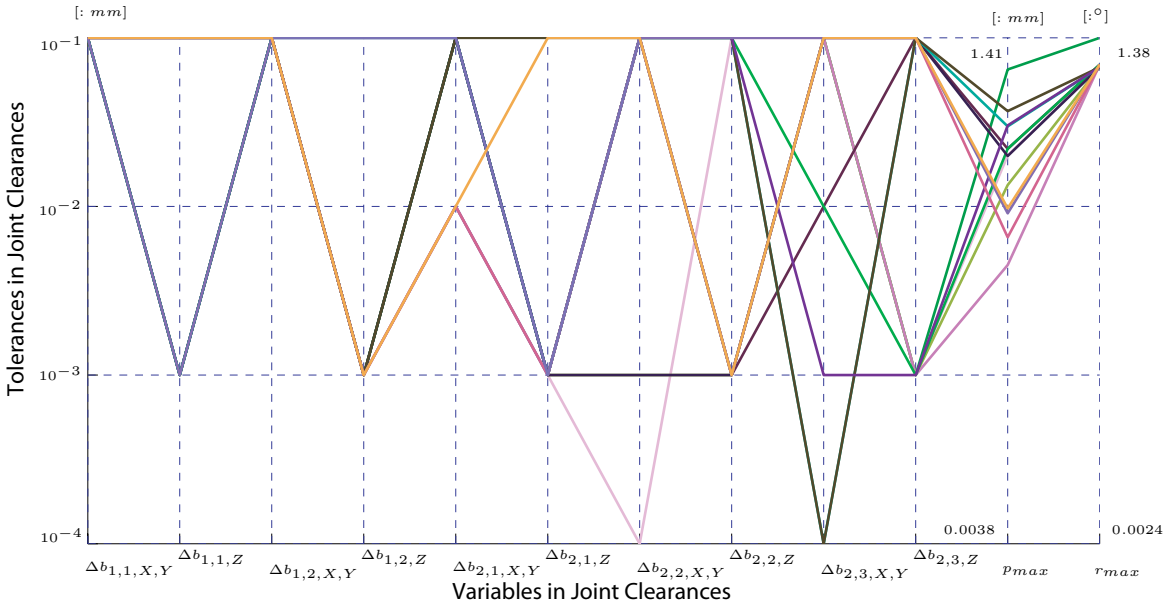


Figure VII.6 – Filtered Entries with Maximum Rotational Errors Larger Than 1 Degree

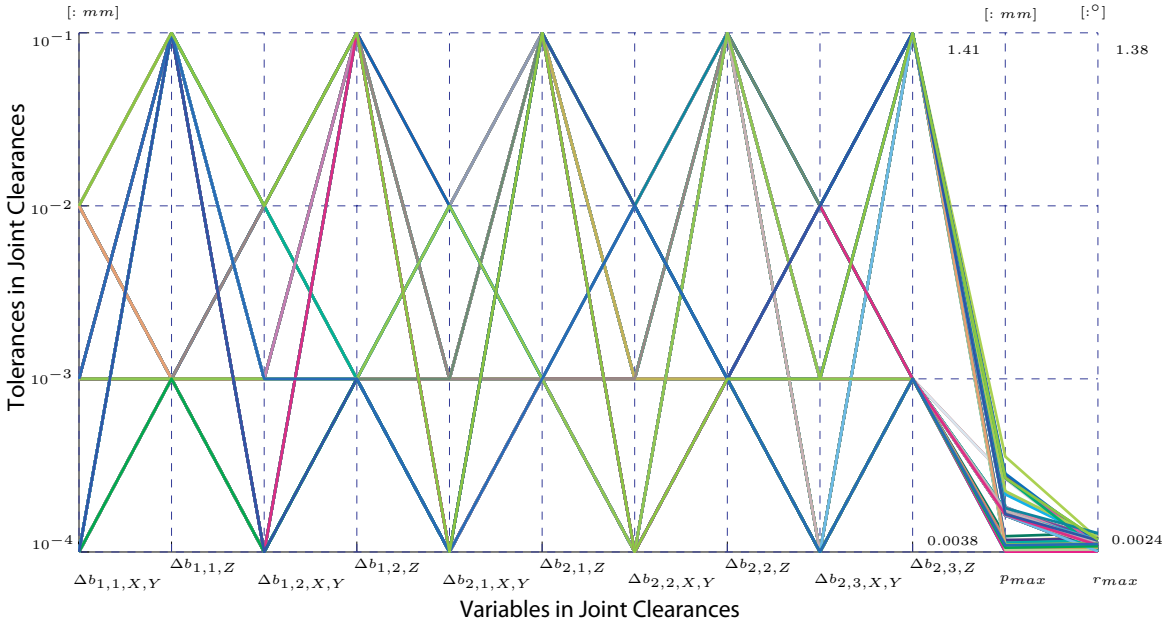


Figure VII.7 – Filtered Entries with Maximum Rotational Errors Less Than 0.05 Degree

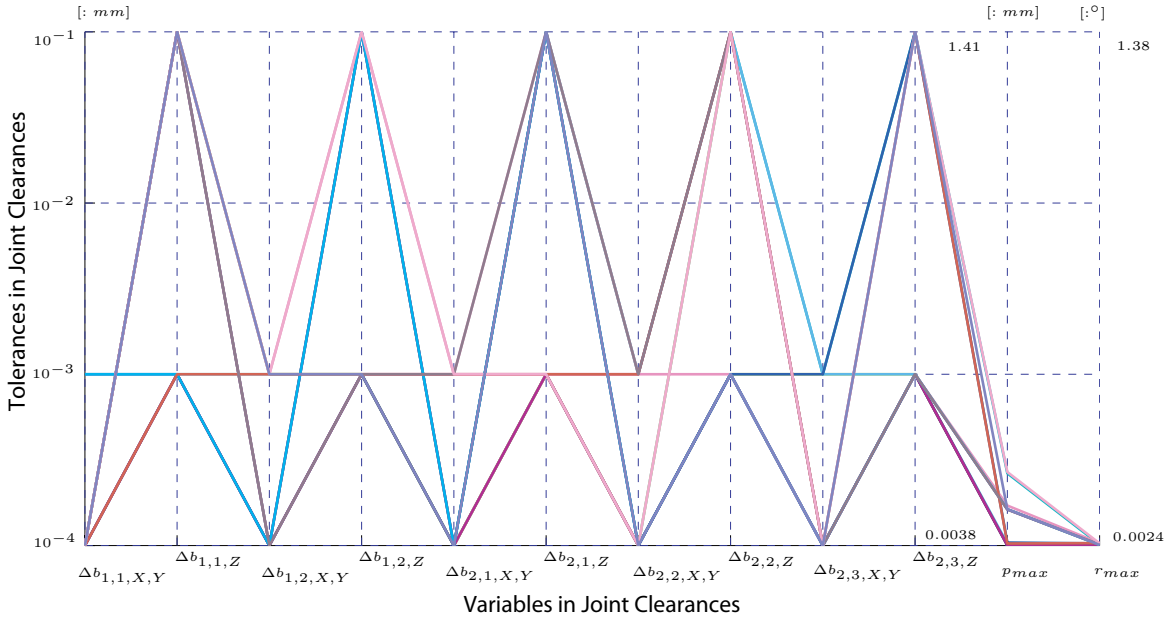


Figure VII.8 – Filtered Entries with Maximum Rotational Errors Less Than 0.01 Degree

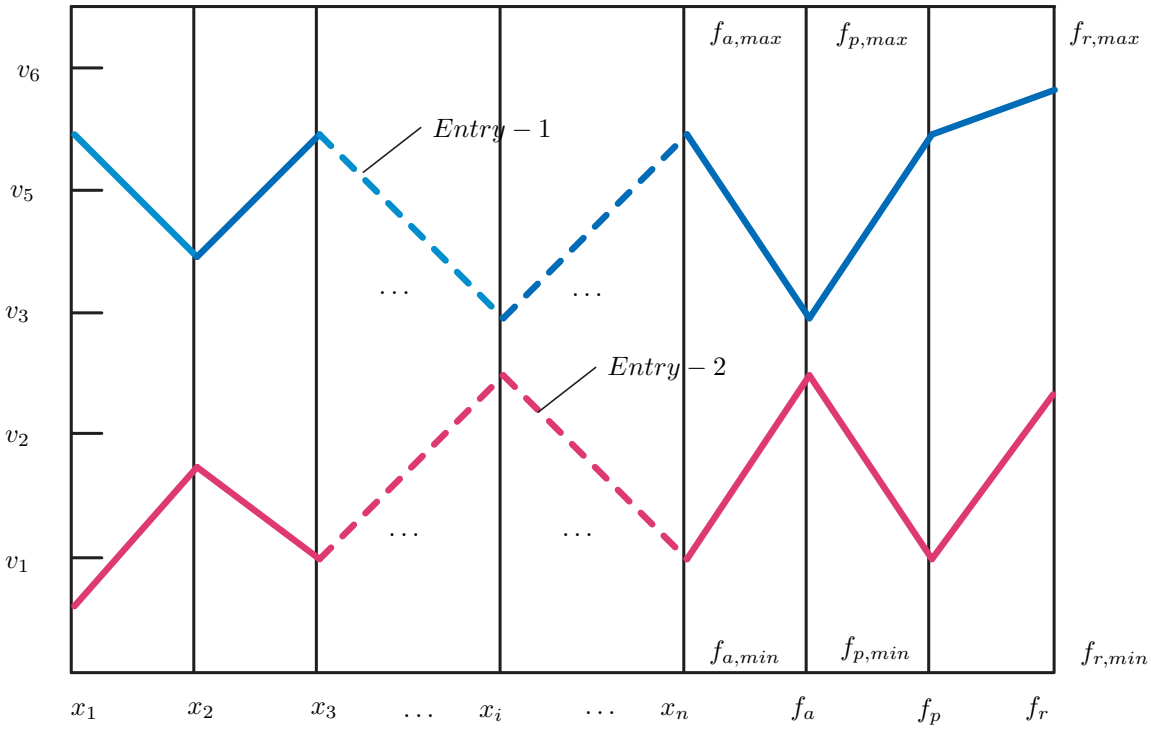


Figure VII.9 – Dimension Synthesis Model

The multi-objective problem defined here is similar to that of Section VII.1. However, variables are dimensions: $x_1, x_2, x_3, \ldots, x_i, \ldots, x_n$. For the objective functions, we are more interested in the area of the largest cuboid-shaped sub-workspace f_a, the maximum positional error f_p and the maximum rotational error f_r.

The largest cuboid-shaped sub-workspace, where the relevant criterion is higher or lower than the desired value, can be solved numerically, using the workspace discretisation [BPC10]. For the dimension synthesis problem , we fix the tolerances in joint clearances. The maximum positional errors and the maximum rotational errors corresponding to different dimensions can be found based on the methods studied in Chapter III and Chapter V.

Likewise, entries can be filtered by the values of the objective functions. From this we can study the influence of dimensions over the objective function. Designer can refer to the plotted entries to have an idea about how to define the dimensions of the robot.

VII.2.2 Dimension Synthesis of the IRSBot-2 Robot

As presented in Chapter II, the IRSBot-2 robot is a novel 2-DOF translational spatial robot, which can be used for pick-and-place operations. Thus, large workspace and high accuracy are required for the IRSBot-2 robot. Here the IRSBot-2 robot is studied based on the dimension synthesis method developed in Section VII.2.1. This results in a design which can have large-workspace and high-accuracy robot.

The IRSBot-2 robot shown in Fig. II.1 is parameterized in Fig. II.4. Here we suppose parameters in Tab. VI.4 are fixed, and the values in the table are adopted. Parameters l_1, l_2, a_1, a_2 and p are changeable. Three levels of values are defined for these parameters, as Tab. VII.1

Table VII.1 – Three Levels of Parameters of the IRSBot-2 Robot

	$l_1 : [mm]$	$l_2 : [mm]$	$a_1 : [mm]$	$a_2 : [mm]$	$p : [mm]$
1	50	50	10	5	15
2	100	100	30	15	25
3	200	200	50	30	40

The total number of combinations of all the parameters with different levels of values is 3^5. Fortunately, the robot can be assembled with any combination of these values. The tolerances in the joints are: $\Delta b_{i,j,X,Y} = \Delta b_{i,j,Z} = 0.1mm$, $\Delta \beta_{i,j,X,Y} = 0.01rad$. As for the value of $\Delta \beta i, j, Z$, it depends if it is active or passive joint. If it is active joint, very small value should be assigned. While if it is passive joint, relatively higher value should be assigned. Because the active joint can be regarded as locked by the motor, and can only have very small rotation or even no rotation, while the passive can have idle motions due to the joint clearances. The maximum positional errors and the maximum rotational errors of the IRSBot-2 robot to joint clearances in both the distal module and the parallelogram are computed at three points: the right-down point, the center point and the left-up point of the largest cuboid-shaped sub-workspace, as shown in Fig. VII.10. Pose errors at those three points are compared, and the maximums are plotted. Areas of the largest cuboid-shaped sub-workspace, maximum positional error and maximum rotational error corresponding to

different combinations of the three levels of values are computed, and all the entries are plotted in Fig. VII.11.

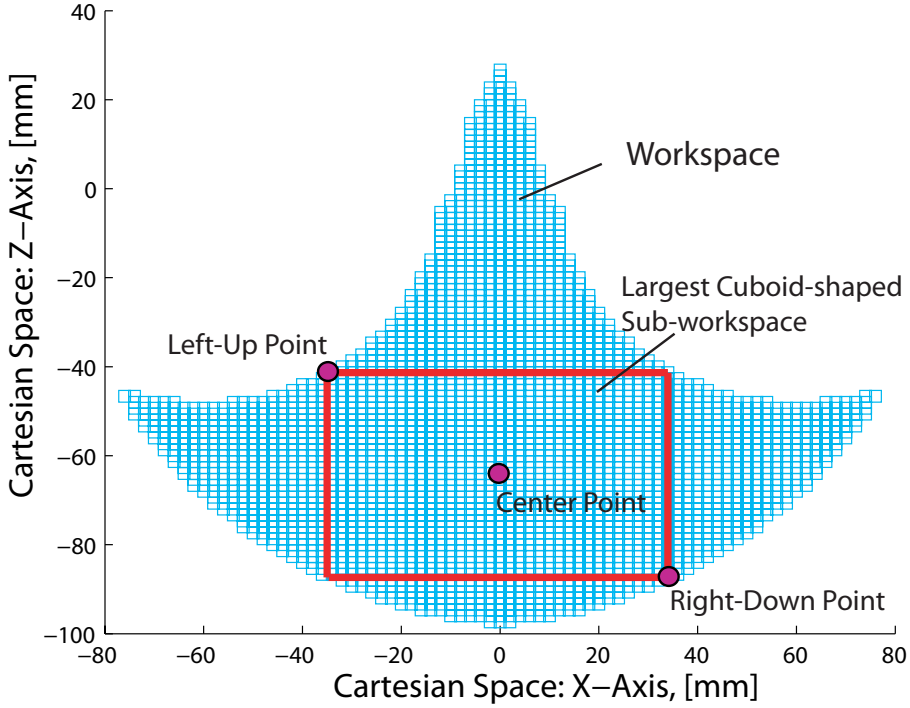


Figure VII.10 – Points Considered in the Largest Cuboid-shaped Sub-workspace

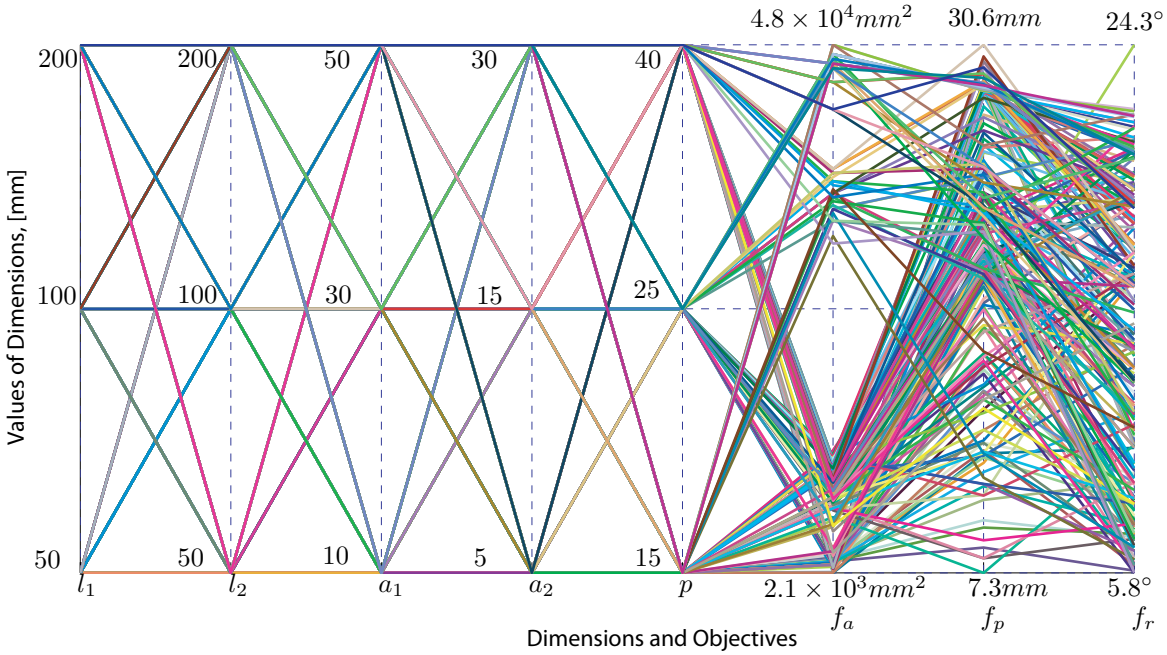


Figure VII.11 – Entries for Dimension Synthesis of the IRSBot-2 Robot

In Fig. VII.11, it can be seen that there are some entries with small cuboid-shaped sub-workspaces but with large pose errors. Obviously, these are bad designs.There are also some entries with large cuboid-shaped sub-workspaces but with low pose errors. Certainly, these are good designs. However, there are too many entries in Fig. VII.11, and it is not

straightforward to find out how the dimensions affect the workspace and the pose error. Therefore, the entries should be filtered, to simplify their analysis. First, entries are filtered by $f_a \geqslant 4 \times 10^4 mm^2$, as shown in Fig.VII.12

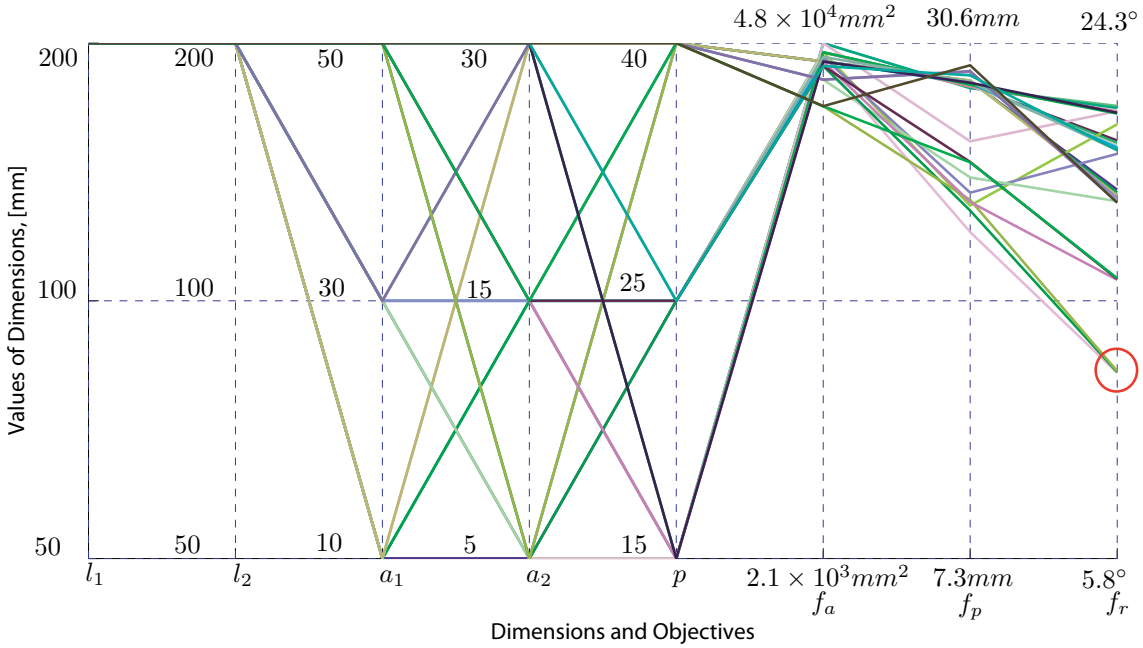


Figure VII.12 – Filtered Entries with Cucoid-shaped Sub-workspace Area Lager Than $4 \times 10^4 mm^2$

From Fig. VII.12, it can be seen that to have a large workspace, the link lengths of l_1 and l_2 should be high. Here, $l_1 = l_2 = 200mm$. In addition, most of the entries are with high pose errors, but there are also some entries with relatively lower maximum rotational errors, as pointed out by the small circle in Fig. VII.12.

In the same way, entries with small workspace ($f_a \leqslant 3 \times 10^3 mm^2$) can also be filtered, as shown in Fig. VII.13. To get a small workspace, it seems that the parameter l_1 should be a small value, as shown in Fig.VII.13. l_1 is constantly equal to $50mm$. As pointed out in Fig.VII.13, there are some entries with small workspace, but high pose error. Obviously, these are bad designs.

Entries can also be filtered by the values of maximum positional errors and maximum rotational errors. The range of maximum positional errors of these entries is $[7.3mm, 30.6mm]$ and the range of maximum rotational errors is $[5.8°, 24.3°]$. Here we filter the entries by $f_p \leqslant 10mm$

From Fig.VII.14, it can be seen that to reach a relatively high accuracy, the linke length of l_2 should be small. Here, $l_2 = 50mm$. As discussed before, passive joints will have some idle motions due to joint clearances, and generally idle motions are relatively large. For IRSBot-2 robot, link l_2 are connected by passive universal joints. The universal joints will have large idle motions due to the joint clearances, and if the link length is long, the positional error will be amplified.

In the same way, we filter the entries with small rotational errors, $f_r \leqslant 8°$, as shown in Fig.VII.15. But it is not so straight forward to see from the figure which parameter is more

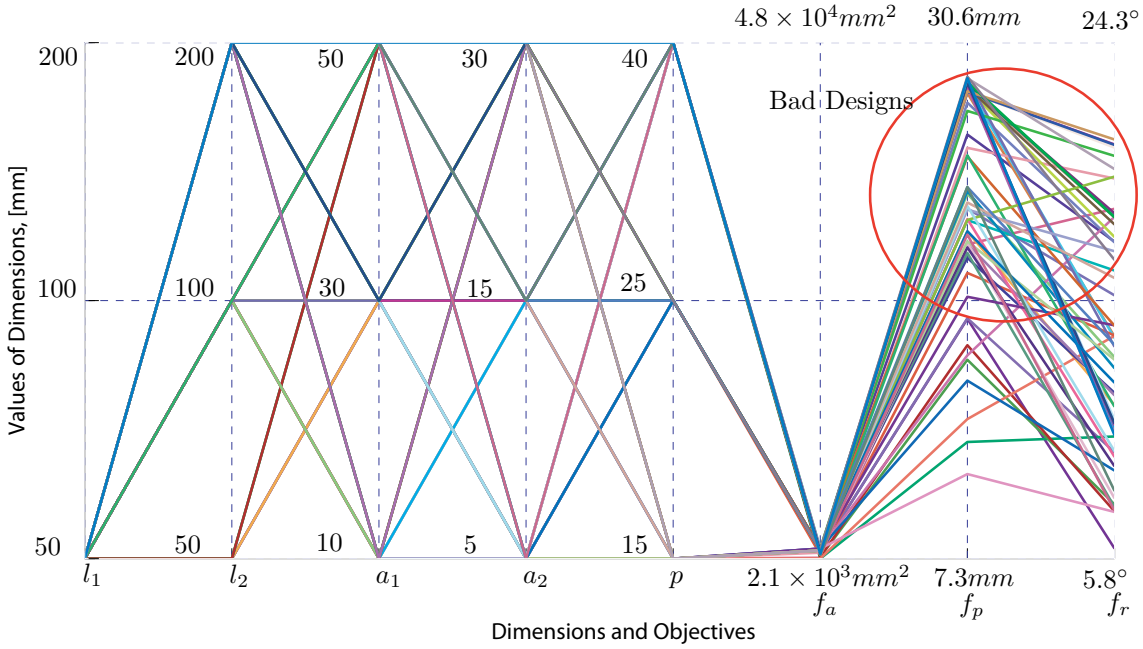


Figure VII.13 – Filtered Entries with Cuboid-shaped Sub-workspace Area Less Than $3 \times 10^3 mm^2$

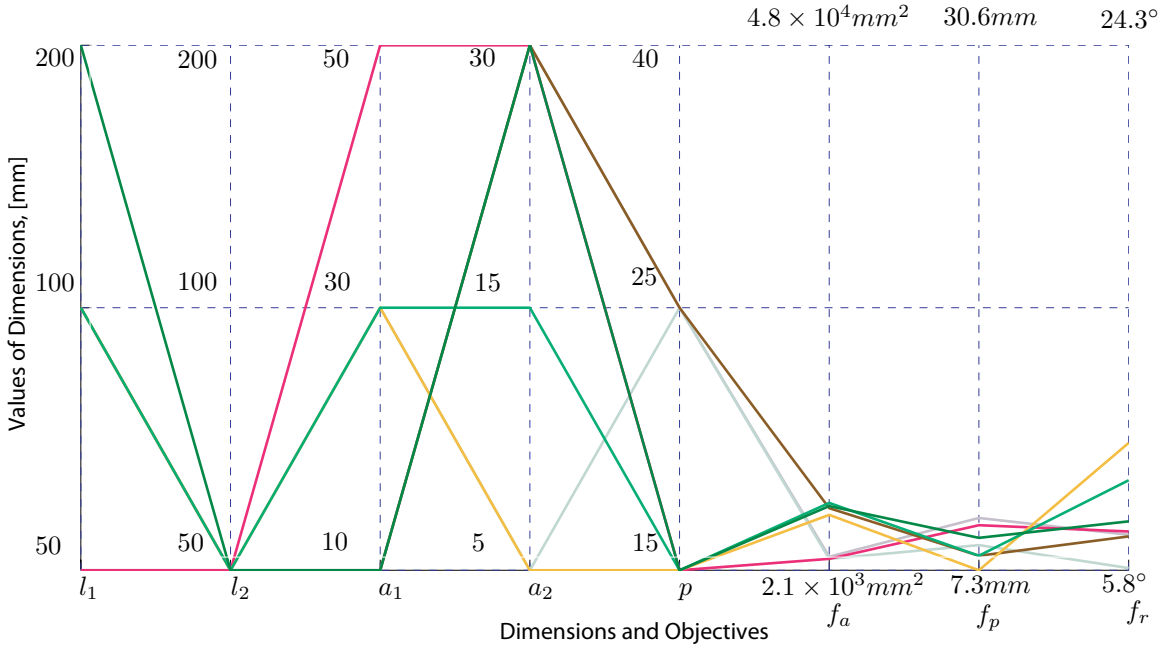


Figure VII.14 – Filtered Entries with Maximum Positional Error Less Than $10mm$

effective for the maximum rotational error. There are some entries with small workspace but large positional error. On the contrary, there also some entries with large workspace but small positional erros. Certainly, we prefer the entries with large workspace but small positional errors, as pointed out in Fig.VII.15.

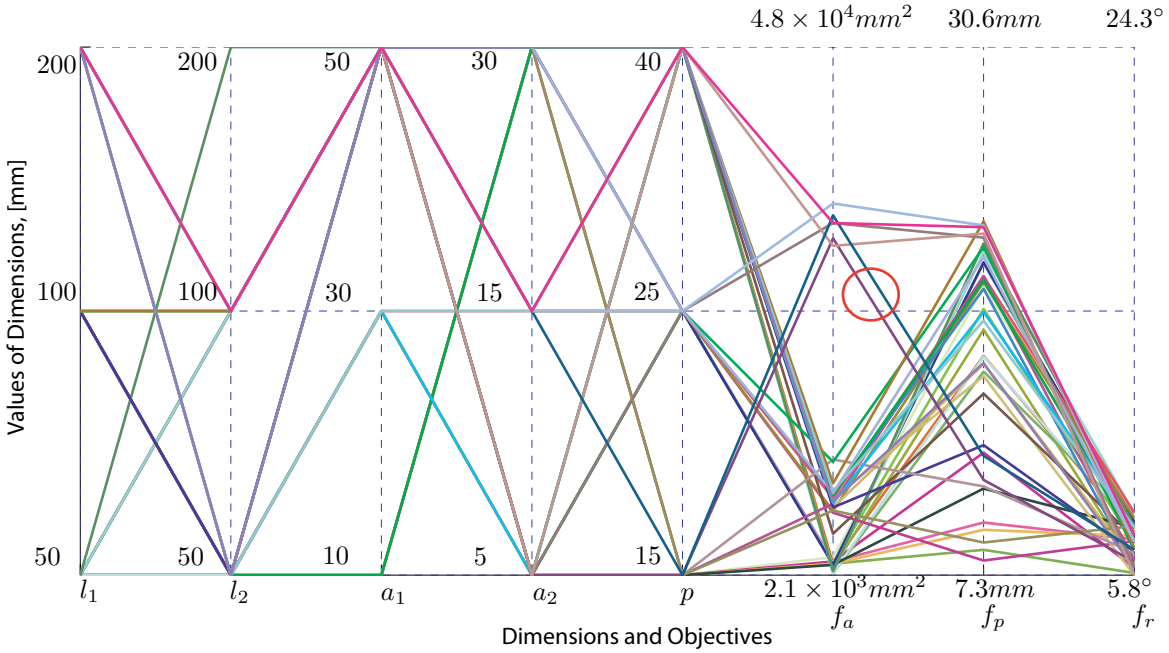


Figure VII.15 – Filtered Entries with Maximum Rotational Less Than 8°

We would like to see the entries with large workspace, small positional errors and small rotational errors pointed out in Fig. VII.15. Furthermore, entries can be filtered by $f_a \geqslant 3 \times 10^4 mm^2$, $f_p \leqslant 15mm$, $f_r \leqslant 8°$, as shown in Fig. VII.16.

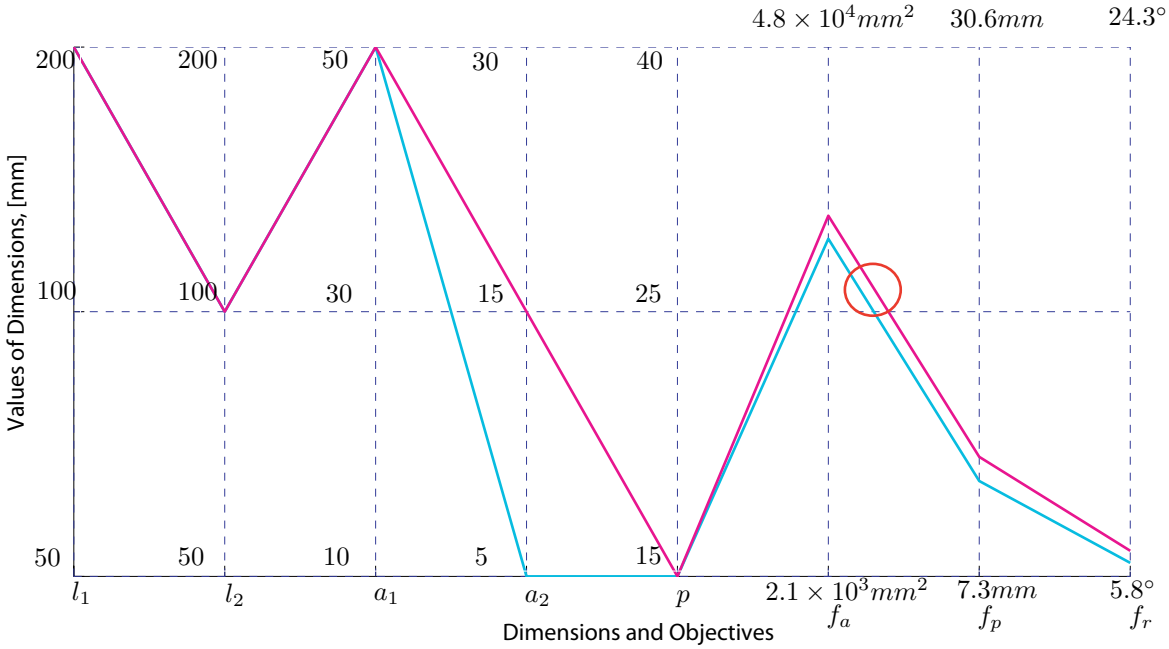


Figure VII.16 – Entries with Large Workspace, Small Positional Errors and Rotational Errors

Overally, the two entries shown in Fig. VII.16 are better designs, which will have large workspace, small maximum positional error and small rotational erros. The design parameters can be extracted from Fig. VII.16, as shown in Tab. VII.2.

Table VII.2 – Two Sets of Good Design Parameters of the IRSBot-2 Robot

	$l_1 : [mm]$	$l_2 : [mm]$	$a_1 : [mm]$	$a_2 : [mm]$	$p : [mm]$
1	200	100	50	15	15
2	200	100	50	5	15

VIII

Conclusions and Future Work

VIII.1 Conclusions

In this thesis, a novel 2-DOF translational spatial robot (the IRSBot-2 robot) was introduced and studied. Methods for sensitivity analysis due to joint clearances and variations in geometric parameters were presented. The 5-bar linkage was studied based on the sensitivity analysis methods. Sensitivity analysis model due to joint clearances for robots with complicated hybrid legs was developped. Sensitiviy of the IRSBot-2 robot to joint clearances was analyzed based on the method developped for robots with hybrid legs. A tolerance synthesis method and a dimension synthesis method were also introduced. Based on the research work, we conclude the following:

1. Both translational and rotational motions in the local frames of the joints termed by joint clearances affect the maximum positional errors and the maximum rotational errors of the end-effectors of the closed-loop robot.

2. The translational motions in the local frames of the joint termed by joint clearances do not affect the maximum rotational errors of the end-effectors of serial robots.

3. The closed-loop robot is not more accurate than the open-loop robot because of the idle motions of the passive joints.

4. The maximum pose errors are larger in singularity configurations and in the vicinity of the those configurations due to joint clearances.

5. Maximum rotational errors of serial robots due to joint clearances are constant throughout the cartesian workspace.

6. The accuracy of the closed-loop robot can be improved by using actuation redundancy, namely, by actuating some passive joints.

7. The sensitivity analysis method due to joint clearances for robots with hybrid legs is a general method, and can be applied to any robot containing closed-loop chains.

8. Maximum pose errors of the IRSBot-2 robot are larger while considering the clearances in the parallelogram joints than without.

9. Maximum rotational errors of the IRSBot-2 robot are large at the center of the largest cuboid-shaped sub-workspace.

10. Tolerances of the two actuated joints are the main sources of the maximum pose errors of the 5-bar linkage.

11. The dimensions of the IRSBot-2 robot with large workspace and small pose error were obtained thanks to proposed dimension synthesis approach.

VIII.2 Future Work

This thesis focuses on the sensitivity analysis due to joint clearances and variations in geometric parameters of robots. Sensitivity analysis models were verified by matlab, and expected results were obtained. Finally, the tolerance synthesis and dimension synthesis methods were developed based on sensitivity analysis approach. However, there are still some works to be done to improve the proposed methodologies for dimension and tolerance synthesis of parallel manipulators.

1. Develop a new method that is able to do sensitivity analysis for robots, by combining both the joint clearances and variations in geometric parameters.

2. Sensitivity analysis due to variations in geometric parameters of the IRSBot-2 robot can not be done because of the over constrained architecture of the distal module. Therefore, an isostatic architecture of the 2-DOF translational robot should be created.

3. The optimization-based sensitivity analysis due to joint clearances is very time consuming, better optimization algorithms need to be developed.

4. Build prototypes of the 5-bar linkage and the IRSBot-2 robot, for conducting some experiments. Then compare with the theoritical results obtained in the study of my thesis.

5. The flexibilities of the links should be also considered to analyze the assembly conditions of the five-bar linkage and the over-constrained IRSBot-2.

Bibliography

[ACKM06] Jorge Angeles, Stéphane Caro, Waseem Khan, and Alexei Morozov. The Kinetostatic Design of an Innovative Schoenflies Motion Generator. *Journal of Mechanical Engineering Science*, 7:935–944, 2006. 4

[BCC10] Nicolas Binaud, Philippe Cardou, and Stéphane Caro. The Kinematic Sensitivity of Robotic Manipulators to Joint Clearances. In *Proceedings of the ASME 2010 International Design Engineering Technical Conference & Computers and Information in Engineering Conference*, pages 1–10, Quebec,Canada, 2010. 11, 12, 13, 29

[BCW09] Nicolas Binaud, S Caro, and Philippe Wenger. Sensitivity Analysis of Degenerate and Non-Degenerate Planar Parallel Manipulators. In *Proceedings of EUCOMES 08*, volume 8, pages 505–512. Springer Netherlands, 2009. 11

[BPC10] Sébastien Briot, Anatol Pashkevich, and Damien Chablat. Optimal technology-oriented design of parallel robots for high-speed machining applications. *2010 IEEE International Conference on Robotics and Automation*, pages 1155–1161, May 2010. 86

[CBW05] Stéphane Caro, Fouad Bennis, and Philippe Wenger. Tolerance Synthesis of Mechanisms: A Robust Design Approach. *Journal of Mechanical Design*, 127(1):86, 2005. 11

[CBW09] Stéphane Caro, Nicolas Binaud, and Philippe Wenger. Sensitivity Analysis of 3-RPR Planar Parallel Manipulators. *Journal of Mechanical Design*, 131(12):121005, 2009. 11

[CWBC06] Stéphane Caro, Philippe Wenger, Fouad Bennis, and Damien Chablat. Sensitivity Analysis of the Orthoglide: A Three-DOF Translational Parallel Kinematic Machine. *Journal of Mechanical Design*, 128(2):392, 2006. 11, 13, 35, 36

[DH55] J. Denavit and R.S. Hartenberg. A kinematic notation for lower-pair mechanisms based on matrices. *Trans ASME J. Appl. Mech*, 23:215–221, 1955. 23, 40

[FWZC03] Kuang-Chao Fan, Hai Wang, Jun-Wei Zhao, and Tsan-Hwei Chang. Sensitivity analysis of the 3-PRS parallel kinematic spindle platform of a serial-parallel machine tool. *International Journal of Machine Tools and Manufacture*, 43(15):1561–1569, December 2003. 11

[GBC12] Coralie Germain, S Briot, and S Caro. Constraint Singularity-Free Design of the IRSBot-2. In *The 13th International Symposium on Advances in Robot Kinematics*, page 1, Innsbruck, Austria, 2012. 15

[GBGC11] Coralie Germain, S Briot, Victor Glazunov, and S Caro. IRSBOT-2: A Novel Two-Dof Parallel Robot for High-Speed Operations. *Proceedings of the ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, 2011. 8, 9, 15

[Ger10] Coralie Germain. *Analyse et Conception d'un Nouveau Manipulateur Parallèle à Deux Degrés de Liberté pour des Applications de Pick-and-Place*. PhD thesis, Ecole Centrale de Nantes, 2010. 15, 18

[HKKP02] Chanhee Han, Jinwook Kim, Jongwon Kim, and Frank Chongwoo Park. Kinematic sensitivity analysis of the 3-UPU parallel mechanism. *Mechanism and Machine Theory*, 37(8):787–798, August 2002. 10

[HLL+04] Tian Huang, Zhanxian Li, Meng Li, Derek G. Chetwynd, and Clement M. Gosselin. Conceptual Design and Dimensional Synthesis of a Novel 2-DOF Translational Parallel Robot for Pick-and-Place Operations. *Journal of Mechanical Design*, 126(3):449, 2004. 8

[KC00] Han S. Kim and Yong J. Choi. The kinematic error bound analysis of the Stewart platform. *Journal of Robotic Systems*, 17(1):63–73, January 2000. 10

[Kha86] Wisama Khalil. A new geometric notation for open and closed-loop robots. *Robotics and Automation.*, (I):1174–1179, 1986. 23, 40, 41

[KT03] Han Sung Kim and Lung-Wen Tsai. Design Optimization of a Cartesian Parallel Manipulator. *Journal of Mechanical Design*, 125(1):43, 2003. 10

[MZL09] Jian Meng, Dongjun Zhang, and Zexiang Li. Accuracy Analysis of Parallel Manipulators With Joint Clearance. *Journal of Mechanical Design*, 131(1):011013, 2009. 11

[SAPH08] Oscar Salgado, Oscar Altuzarra, Víctor Petuya, and Alfonso Hernández. Synthesis and Design of a Novel 3T1R Fully-Parallel Manipulator. *Journal of Mechanical Design*, 130(4):042305, 2008. 4

[VEU04] Philip Voglewede and Imme Ebert-Uphoff. Application of Workspace Generation Techniques to Determine the Unconstrained Motion of Parallel Manipulators. *Journal of Mechanical Design*, 126(2):283, 2004. 11

[VPC05] Stefano Venanzi and Vincenzo Parenti-Castelli. A New Technique for Clearance Influence Analysis in Spatial Mechanisms. *Journal of Mechanical Design*, 127(3):446, 2005. 11, 13

[WG01] Philippe Wenger and Clément Gosselin. A comparative study of parallel kinematic architectures for machining applications. In *2nd Workshop on Computational Kinematics*, volume 1, Seoul, South Korea, 2001. 9, 10

[WM93] J. Wang and O. Masory. On the Accuracy of a Stewart Platform-Part One, The Effect of Manufacturing Tolerance. In *Proceeding of the IEEE International Conference on Robotics Automation*, pages 114–120, Atlanta, USA, 1993. 10

[WR89] HHS Wang and B. Roth. Position errors due to clearances in journal bearings. *Journal of mechanisms, transmissions, and*, 111(February 1988):315–320, 1989. 11

Acknowledgement

This thesis was done under the framework of EMARO "European Master on Advanced Robotics" Program. Thanks to EMARO program, I had the opportunity to study in University of Genova, Italy for the first academic year, and then in Ecole Centrale de Nantes, France for the second academic year. I would like to express my deepest appreciation to all the professors who conducted lectures for us. Their profound knowledge and excellent instructions make EMARO a perfect master program on robotics and strengthen our interests and knowledge on robotics. Moreover, EMARO provides us a multi-culture environment, where I was able to know people from all over the world and get to know how to live and work in an international community. Thanks to all my colleagues for their kindness, friendliness and help.

I would like to express my sincest gratitudes to my supervisors Dr. Stéphane CARO and Dr. Sébastien BRIOT. They have supported me throughout my thesis with their patience and knowledge while giving me the room to work in my own way. I attribute the level of my master to their encouragement and effort. And the experience of working with them for the thesis will be my life-long benifit.

Before coming to EMARO, I had the chance to do my first master on mechanical design and theory in Shanghai Jiao Tong University, China under the guidance of Prof.Weizhong Guo and Prof. Feng Gao, with whom I started to do research work on parallel mechanisms and parallel robots. Thanks to them for introducing me into the field of robotics.

Moreover, I would like to express my deepest thanks to my parents, my brother, my sister and my girl friend. They have been supporting and encouraging me in my academic career. Their support and encouragement had been strong background for me to go forward.