Visual Calibration, Identification and Control of 6-RSS Parallel Robots

Pengcheng Li

A Thesis

in

The Department

of

Mechanical, Industrial & Aerospace Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy (Mechanical Engineering) at

Concordia University

Montréal, Québec, Canada

June 2020

CONCORDIA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Pengcheng Li

Entitled: Visual Calibration, Identification and Control of 6-RSS Parallel Robots

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

Doctor of Philosophy (Mechanical Engineering)

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

Signed by the final examining commitee:

______________________________ Chair
Dr. Anjan Bhowmick

______________________________ External Examiner
Dr. Zheng Hong (George) Zhu

______________________________ External to Program
Dr. Shahin Hashtrudi Zad

______________________________ Examiner
Dr. Brandon Gordon

______________________________ Examiner
Dr. Youmin Zhang

______________________________ Thesis Supervisor
Dr. Wen-Fang Xie

Approved by ______________________________
Dr. Ivan Contreras, Graduate Program Director

June 15, 2020 ______________________________
Dr. Amir Asif, Dean
Gina Cody School of Engineering & Computer Science

Abstract

Visual Calibration, Identification and Control of 6-RSS Parallel Robots

Pengcheng Li, Ph.D.

Concordia University, 2020

Parallel robots present some outstanding advantages in high force-to-weight ratio, better stiffness and theoretical higher accuracy compared with serial manipulators. Hence parallel robots have been utilized increasingly in various applications. However, due to the manufacturing tolerances and defections in the robot structure, the positioning accuracy of parallel robots is basically equivalent with that of serial manipulators according to previous researches on the accuracy analysis of the Stewart Platform [1], which is difficult to meet the precision requirement of many potential applications. In addition, the existence of closed-chain mechanism yields difficulties in designing control system for practical applications, due to its highly coupled dynamics.

Visual sensor is a good choice for providing non-contact measurement of the end-effector pose (position and orientation) with simplicity in operation and low cost compared to other measurement methods such as the coordinate measurement machine (CMM) [2] and the laser tracker [3]. In this research, a series of solutions including kinematic calibration, dynamic identification and visual servoing are proposed to improve the positioning and tracking performance of the parallel robot based on the visual sensor.

The main contributions of this research include three parts. In the first part, a relative pose-based algorithm (RPBA) is proposed to solve the kinematic calibration problem of a six-revolute-spherical-spherical (6-RSS) parallel robot by using the optical CMM sensor. Based on the relative poses between the candidate and the initial configurations, a calibration algorithm is proposed to determine the optimal error parameters of the robot kinematic model and external parameters introduced by the optical sensor. The experimental results demonstrate that the proposal RPBA using optical CMM is an implementable and effective method for the parallel robot calibration.

The second part focuses on the dynamic model identification of the 6-RSS parallel robots. A visual closed-loop output-error identification method based on an optical CMM sensor is proposed for the purpose of the advanced model-based visual servoing control design of parallel robots. By using an outer loop visual servoing controller to stabilize both the parallel robot and the simulated model, the visual closed-loop output-error identification method is developed and the model parameters are identified by using a nonlinear optimization technique. The effectiveness of the proposed identification algorithm is validated by experimental tests.

In the last part, a dynamic sliding mode control (DSMC) scheme combined with the visual servoing method is proposed to improve the tracking performance of the 6-RSS parallel robot based on the optical CMM sensor. By employing a position-to-torque converter, the torque command generated by DSMC can be applied to the position controlled industrial robot. The stability of the proposed DSMC has been proved by using Lyapunov theorem. The real-time experiment tests on a 6-RSS parallel robot demonstrate that the developed DSMC scheme is robust to the modeling errors and uncertainties. Compared with the classical kinematic level controllers, the proposed DSMC exhibits the superiority in terms of tracking performance and robustness.

Acknowledgements

More than four years of doctoral career is coming to an end. Every step of my growth is inseparable from the encouragement and help of mentors, families and friends.

Here, I sincerely thank you. First of all, I would like to take the opportunity to greatly thank my supervisor, Dr. Wen- Fang Xie for her nonstop supports, suggestions, and guidance, not only on my research, but also in life. She enrolled me into the door of the subject area and let me have the chance to do the research, which change my life to a wonderful world. I am deeply inspired by her careful thinking, unremitting and rigorous work attitude, broad and profound knowledge. Whenever encounter difficulties, I can always be encouraged by her with patient guidance and full enthusiasm in time. The period of working with Dr. Xie will be the wealth that accompany and benefit me all my life. On the completion of this thesis, I would like to to express the most sincere gratitude, the highest respect and the best wishes to my supervisor for all her contributions, guidance and encouragements.

Hearty thanks also go to my colleagues and friends who accompanied me along these years: Dr. Dongdong Zheng, Dr. Xiaoming Zhang, Dr. Rui Zeng, Dr. Shutong Li, Mrs. Tingting Shu, Mr. Ronghua Zhang, Mr. Xiaoyang Zhang and Dr. Ahmad Ghasemi. I would like to thank my professor Dr. Wei Tian who gives me guidance and support. I am also grateful to all those people who have given me support and I didn't have the chance to thank them. I ultimately would like to thank my lovely parents Mr. Keming Li and Mrs. Xiaoli Min for their support, regardless of my choices, and for having allowed me, with their efforts, to get to where I am now. My deepest heartfelt gratitude goes out to my wife, Mrs. Jianyun Chen. She supported and encouraged me from the very beginning step of my PhD study to the end. I am really grateful for having her in my life.

Dedicated to my beloved wife,

Jianyun Chen

and my lovely parents,

Keming Li

and

Xiaoli Min

Table of Contents

List of Tables

List of Figures

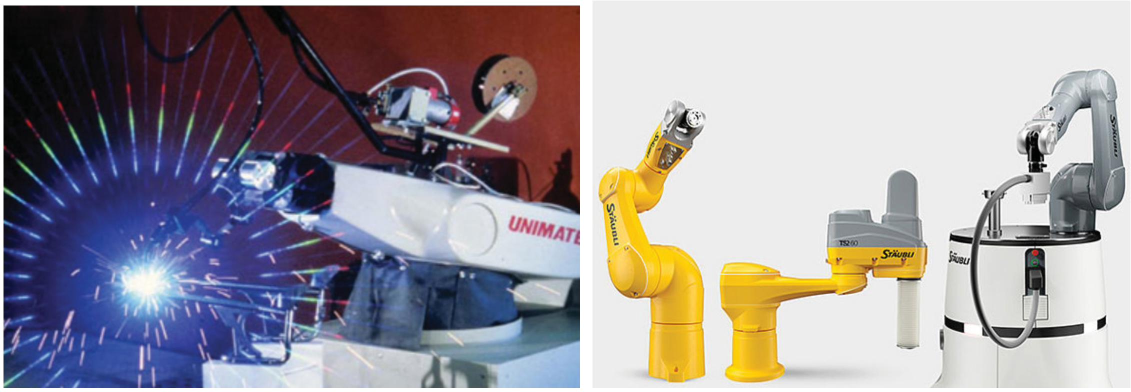
Chapter 1

Introduction

1.1 Background

Today, industrial robots have been widely used in aerospace, automotive manufacturing, special processing, electronic package, and modern logistics industries. The industrial robots play important roles in processing, welding, cutting, spraying, handling, sorting and assembly, etc. And they have become indispensable for ensuring product quality, improving production efficiency, and reducing production costs. Generally speaking, industrial robots are mainly classified into two categories according to their topological configuration: serial robots with serial topology and parallel robots with parallel topology. In addition, in recent years, there has also been a type of mixed configuration robot that integrates both serial and parallel structures.

The world's first industrial robot with the serial topology, as shown in Figure 1.1a, was invented by George C.Devol in 1961, and has been successfully used on general motors assembly lines [4]. Since then, various types of serial robots have been developed and have greatly promoted the process of industrialized production. After the 1980s, with the development of technologies such as controllers, drives, sensors, and high-level programming languages, industrial robots based on the serial topology has entered in the golden age of development. Robot manufacturers such as ABB, KUKA, FANUC, KAWASAKI, Stäbli, etc., have successively developed various types of serial robots, as shown in Figure 1.1b, which are widely used in different fields of industrial production. To this date, the serial robot technology is relatively mature.



(a) (b)

Figure 1.1: (a) The serial robot - Unimate [4], (b) Industrial serial robot [5]

After entering the 1980s, with the continuous expansion of application fields and the diversification of production environments, higher requirements are posed on the performance of industrial robots, such as speed, accuracy, stiffness, and dynamic characteristics. Due to its own structural characteristics and error accumulation effects, the applications of the serial robot in some production and processing are limited. In this case, a class of parallel topology mechanism (parallel kinematic machines) as a new type of industrial robot, a parallel robot, has attracted increasing interest in academic and industry society. Compared with serial robots, parallel robots have high stiffness, stable structure, strong load-bearing capacity, small error accumulation effect, small motion inertia and easy inverse kinematic solution etc. [11]. These characteristic allow the parallel robots to have a complementary relationship with serial robots in applications, and broad application prospects.

The theoretical structure of a parallel robot, dating back to 1943, was proposed for automatic painting by Willard L. V. Pollard. However, subject to the technical conditions at the time, the physical mechanism was not built. As shown in Figure 1.2a, a six-degree-of-freedom (6-DOF) parallel mechanism for a flight simulator, named as Stewart platform, was designed by British engineer in 1965 [6]. Since then, the parallel robot technology has been greatly promoted. Its application range covers the fields of motion simulators, machining process, medical tools, aerospace docking devices and micro-motion mechanisms, etc [12]. One typical parallel robot in Figure 1.2b is the well-known Delta parallel robot, which claims to be the fastest pick-and-place robot in the world. Delta parallel robots are often used in high-speed sorting and packaging applications [7].


(a)
(b)

Figure 1.2: (a) The Stewart robot [6], (b) The Delta parallel robot [7]

1.2 Motivation

Recently, the increasing need for high performance composite structures in various industries has greatly driven the development of the composite manufacturing technologies [13]. The traditional manual production cannot meet the high efficiency, accuracy and quality requirement of emerging industry. This aspect has encouraged the development of new production technologies such as automated fiber placement (AFP) systems. The AFP technology automates the production of composite material structures using prepregs, which are present in the form of strips composed of impregnated fiber tapes (glass, carbon, etc.) of semi-polymerized resin. In the AFP system, a deposition head with the ability of heating and compacting the resin prepregs is mounted on a fiber placement machine or an industrial robot. The AFP system in Concordia University is shown in Figure 1.3. The Kawasaki robot carrying the deposition head lays up the prepregs traversing the surface of the tooling mandrels. In the process of manufacturing, the tooling mandrel is used as a mold to be wound around by the prepregs to form a certain structure of composite part. The mandrel will be removed after the part is cured. The current AFP systems can significantly improve the efficiency and quality of the production of composite materials. However, they are limited to the production of the open surfaces presenting a flat or contoured surface, or simple revolution parts such as cylinders or cones due to the insufficient degree-of-freedom (DOF) of the system and the difficulties in

Figure 1.3: The AFP machine in Concordia University

(a) (b)

Figure 1.4: (a) The Y-shape tube, (b) The bicycle frame

generating trajectories. Especially, the aerospace industry and the production industries of sports equipment are now exploring to use this technique for the production of structures with more complex geometries, like ”Y” tubes or the structures forming closed-loops such as bicycle frames, as shown in Figure 1.4.

To be able to manufacture the structures with complex geometries, the flexibility of the AFP system should be improved. The collaborative robotic system consisting of two robots is a promising solution to increase the dexterity by employing one robot to hold the fiber placement head at the end-effector and another robot to hold the mandrel. The two-serial-robot collaborative system may not solve the fiber placement problem, since the serial robots tend to deform and lose the accuracy due to its cantilever structure, considering the weight of the mandrel and the compaction force.

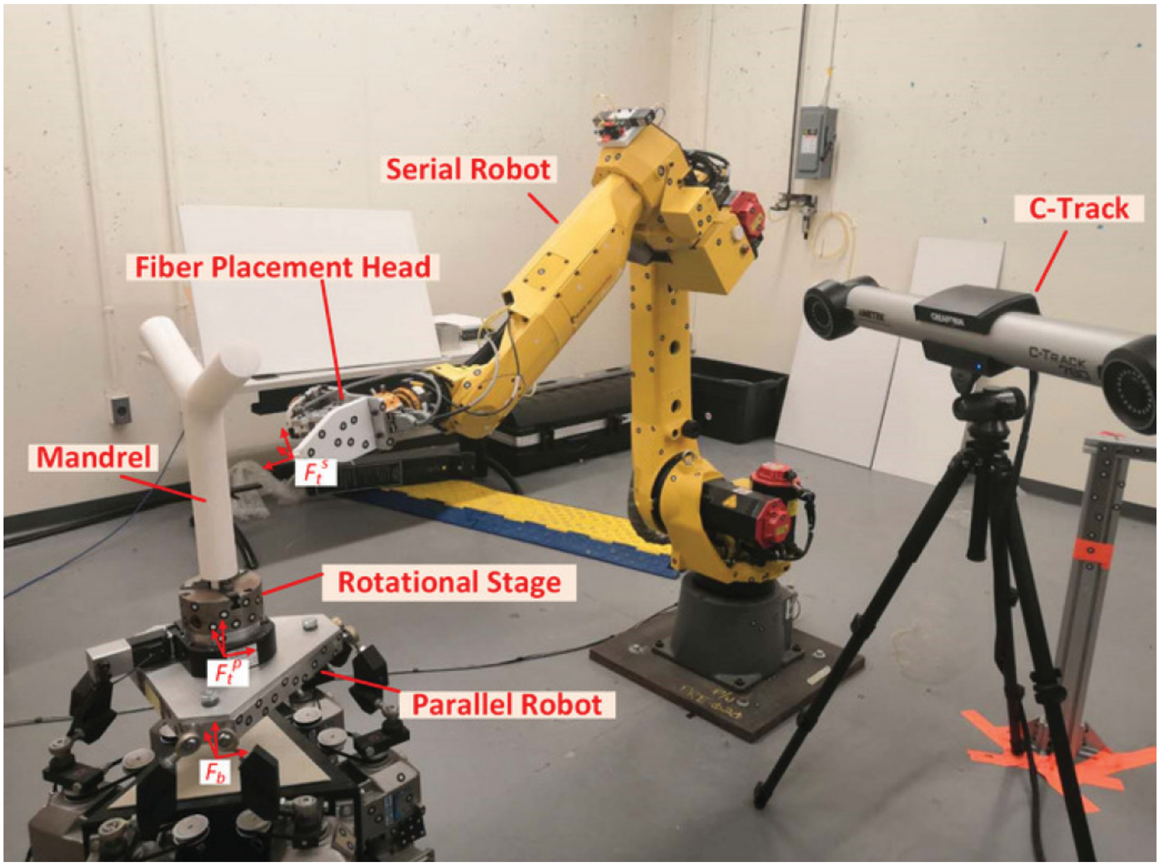


Figure 1.5: The collaborative AFP machine in Concordia University

Compared with the serial robots, 6-DOF parallel robots enjoy better stiffness. Therefore, a parallel-serial collaborative robot system, in additional to a rotational stage mounted on the platform of the parallel robot, is built for handling the complex structures manufacturing in Concordia University as shown in Figure 1.5.

The positioning and tracking accuracy is one of the most important performance indicators of a parallel robot, which directly affects the final manufacturing results of the collaborative AFP system. However, the accuracy of the 6-DOF parallel robot is relatively low compared with the serial robot, which highly restricts the further development of the collaborative AFP system. With the gradual expansion of application fields, the requirements for the positioning and tracking accuracy of parallel robots are also increasing, especially in the fields of aviation, industrial finishing, medical assistance and micro motion. In practice, the accuracy of the 6-DOF parallel robots is relative low due to the existence of many factors affecting the motion error. Therefore, the research on improving the positioning and tracking accuracy of parallel robots has important theoretical and practical significance, and thus is the main motivation of this thesis research work.

1.3 Problems and Solutions

The parallel robot motion error is referred to as the 6 dimension (6D) pose errors between the actual end-effector frame's trajectory of the parallel robot and the ideal motion trajectory. There are many factors affecting the parallel robot motion error, and the source of the errors can be attributed to the mechanism errors and environment errors. The mechanism errors include geometric errors, flexible deformation, thermal deformation, force deformation and friction, etc. The environment errors mainly are introduced by the temperature, humidity, and operation process of the surrounding environment. According to the dynamic characteristics of errors, they can be classified into static errors and dynamic errors. Static errors remain constant during movement of the parallel robot, including structural errors, environmental factors, control systems, and transmission system error, etc. Dynamic errors change over time during the movement of a parallel robot, including flexible deformation of components caused by forces, inertial forces, and weight, etc [1]. The main errors of the 6-DOF parallel robot in this research are analyzed as following:

- Mechanism error. The errors of the basic components of the parallel robot during manufacturing and assembly are unavoidable, which result in errors between the actual parameters and ideal parameters of the components. Because the motion control of the parallel robot is based on the ideal structural parameters, it is not completely consistent with the actual structural parameters, which causes a mechanical error in the parallel robot and in turn leads to errors in the trajectory of the parallel robot. There are hundreds of error sources for structural errors of parallel robots, which are slightly different depending on the configuration. Generally, to simplify the calculation, only the length error of the links and the position error of the joints are considered [6]. The mechanism errors account for more than 60% of the total parallel robot motion error [1].

- Control system error. Joint position closed-loop controller is commonly used in the parallel robot control systems instead of workspace pose closed-loop controller, due to the lack of the proper pose measurement device [12]. This control strategy is easy to be implemented, due to the fact that inverse kinematics of parallel robots can be analytically solved, and the measured joint angles can be used as feedback signal in the joint space control loop. However, the convergence of the joint positions cannot guarantee that of the 6D pose, according to the mechanism errors. Another issue of the current controllers of the parallel robots is that the dynamics of the parallel robots are ignored. The dynamic model of the parallel robot plays an important role in the model-based controller designs, especially in applications where high positioning and tracking accuracy is needed.

Kinematic calibration is a promising solution for removing the negative influence of mechanism errors and improving the positioning accuracy of end-effector output in a robot control system. Kinematic calibration is the process of determining the actual values of kinematic parameters of the robots which describe the relative position and orientation of links and joints in the robot. Basic steps of kinematic calibration are kinematic modeling, measurement and implementation [6]. Although a lot of methods for modeling and calibrating serial robots have been proposed, these methods are not always suitable for parallel robots. The existence of closed-chain mechanism and more moving parts yields difficulties on dynamic analysis of parallel robots. For example, a 6-DOF Steward platform has thirteen moving bodies (twelve legs and one end-effector), which are highly nonlinear in dynamic modeling. Since the dynamics parameters are normally unknown or approximately derived from CAD model for the industrial robots, the dynamic identification is necessary to be carried out.

Vision is a good choice for providing non-contact measurement of the end-effector's pose with respect to the camera frame and has been utilized for kinematic calibration, dynamic identification and control of the robots [14, 15]. Vision system can observe and estimate the complete end-effector pose in real time with simplicity in operation and low cost compared to other measurement methods, e.g. the laser tracker, and therefore, has a great potential for the calibration, identification and control of parallel robots. Vision can also be incorporated into the feedback control loop of robotic systems to increase the flexibility and adaptability. The pose of the end-effector can be acquired on-line provided that the image processing is fast enough. Shirai and Inoue [16] proposed an open-loop control method, so called ``look-then-move scheme˝, in 1973. With the development of high speed computer, visual seroving, ``look-and-move˝control scheme, enjoys a fast progress. The most important advantage of visual servoing is to relieve control scheme from forward kinematics calculation. There is no analytical expression for the forward kinematic model of 6-DOF parallel robots [17]. Therefore visual servoing is an effective method for improving the tracking accuracy of parallel robots. Compared with the research work on visual servoing of serial robots [18, 19], there are limited research work that has been dedicated to 6-DOF parallel robots.

The optical Coordinate Measurement Mechine (CMM) sensor is a dual camera based vision sensor. The coordinates of target reflectors in the field of view can be directly measured by the sensor. By attaching four non-collinear reflectors on the platform, the pose of the end-effector can be derived. The optical CMM sensor have been applied to the path tracking controller design [20] for the serial robots. In this research, the optical CMM sensor is utilized to measure the end-effector's pose.

1.4 Scope and Objectives

The main goal of this research is to improve the positioning and trajectory tracking accuracy of the 6-DOF parallel robot by developing efficient kinematic calibration, dynamic identification, and control algorithm based on the optical CMM.

The research work in this research is carried out in four main phases. Firstly, the theoritical models related to the accuracy of the parallel robot, including inverse kinematics, forward kinematics, velocity relations and the dynamic model of the 6-RSS parallel robots are needed to be built for developing visual calibration, identification and controller purpose. In the second phase, a calibration algorithm by using the optical CMM and several target reflectors attached on the end-effector is presented for the 6-RSS parallel robot. The detected feature points of the reflectors can be used to estimate the poses of the end-effector. Correspondingly, the methods of constructing the objective function, finding the updating algorithm, selecting candidate configuration set and determining the proper working range are developed for the visual kinematic calibration purpose.

The third phase is the development of the dynamic parameter identification method for 6-RSS parallel robot based on the optical CMM. Due to the fact that it is normally nontrival to measure the torque or current of the actuators of the industrial robots. The closed-loop output-error identification method based on the optical CMM is developed for the 6-RSS parallel robots, which is to find the dynamic model parameters by minimizing the output deviation between the actual and simulated systems subjecting to the same input.

Finally, in order to improve the tracking performance of the 6-RSS parallel robot, the dynamic model based visual servoing controller should be studied. Based on the calibrated kinematic and identified dynamic parameters, the computed torque visual servoing controller can be developed to derive good tracking performance. However the un-modeled dynamics and disturbance cannot be avoided, hence a sliding mode-based visual servoing controller can be developed to compensate the uncertainties and further improve the tracking performance. Due to the lack of the velocity measurement, the state variables estimation should be studied based on the pose measurement. At last, the stability analysis of the visual servoing controller needs to be given to guarantee the stability of the designed controller.

1.5 Contributions

In this Ph.D. project, a series of solutions for improving the positioning and tracking accuracy of the 6-RSS parallel robot are proposed based on a dual camera based optical CMM sensor. The main contributions of this project are summarized as following:

- The inverse kinematics and numerical solution of the forward kinematics for the 6-RSS parallel robot are derived. The velocity analysis of the 6-RSS parallel robot is given. The explicit dynamic model of the parallel robot is built based on the virtual work theory for the visual identification and servoing purpose.

- A relative posture-based algorithm is proposed to solve the kinematic calibration problem of a 6-RSS parallel robot by using the optical CMM sensor. This method applies both the position and orientation variations and does not need the accurate location information of the detection sensor. The simulation results validate the effectiveness of the algorithm under different circumstances. And the experimental results demonstrate that the calibrated kinematic parameters can be used to improve the positioning accuracy of the parallel robot.

- A closed-loop output-error identification method based on the optical CMM sensor is proposed for the 6-RSS parallel robot. The end-effector pose is measured by the optical CMM and served as the output of the real plant. The forward kinematics of parallel robots, which is usually solved by using time-consuming numerical algorithm, can be avoided. The exact knowledge of the built-in controller and the joint torque are not needed. The dynamic model parameters are identified by using nonlinear optimization technique. The experimental tests validate the identification results.

- A dynamic sliding mode control (DSMC) scheme is proposed to improve the tracking accuracy of the 6-RSS parallel robot. The proposed control scheme adopts the optical CMM sensor to obtain the real time pose information of the end-effector of parallel robot and to use it as the feedback signal. The DSMC scheme is robust to the modeling errors and uncertainties. With the benefit of the position-to-torque converter, the proposed DSMC scheme can be implemented in the industrial parallel robot. The stability of the proposed scheme has been proved by using the Lyapunov function. The experimental tests of the proposed control scheme have been carried out on the 6-RSS parallel robot. The comparison with the kinematic level controllers demonstrates the superiority of the proposed dynamic level visual servoing.

1.6 Publications

The achieved research results of this Ph.D. project have been submitted and published in the following journal and international conference papers.

Journal papers:

(1) Pengcheng Li, Xiaoming Zhang, Wen-Fang Xie, and Suong Van Hoa. Operation of the collaborative composite manufacturing (CCM) system. *JoVE (Journal of Visualized Experiments)*, (152):e59969, 2019.

(2) Pengcheng Li, Ahmad Ghasemi, Wen-Fang Xie, and Wei Tian. Visual closed-loop dynamic model identification of parallel robots based on optical CMM sensor. *Electronics*, 8(8):836, 2019.

(3) Ahmad Ghasemi, Pengcheng Li, and Wen-Fang Xie. Adaptive switch image-based visual servoing for industrial robots. *International Journal of Control, Automation and Systems*, 18(5):1324-1334, 2020.

(4) Ahmad Ghasemi, Pengcheng Li, Wen-Fang Xie, and Wei Tian. Enhanced switch image-based visual servoing dealing with featuresloss. *Electronics*, 8(8):903, 2019.

(5) Pengcheng Li, Rui Zeng, Wen-Fang Xie, and Xiaoming Zhang. Relative posture-based kinematic calibration of a 6-rss parallel robot by optical coordinate measurement machine. *International Journal of Advanced Robotic Systems*, 15(2):1729881418765861, 2018.

(6) Dongdong Zheng, Pengcheng Li, Wen-Fang Xie, and Dan Li, Identification and control of flexible joint robot using multitime-scale neural network, *Journal of Shanghai Jiao Tong University (Science)*, 25(4/5), 2020.

Submitted Journal paper:

(1) Pengcheng Li, Wen-Fang Xie, and Wei Tian. Dynamic visual servoing of a 6-RSS parallel robot based on optical CMM. *IEEE/ASME Transactions on Mechatronics*. Under review.

Conference papers:

(1) Pengcheng Li, Ahmad Ghasemi, Wen-Fang Xie, and Wei Tian. Visual Sensor-Based Dynamic Identification of a 6-RSS Parallel Robot. In *Proceedings of Eighth International Conference on Control, Automation & Information Sciences (ICCAIS 2019)*, Chengdu, China, October 23–26, 2019.

(2) Pengcheng Li, Wen-Fang Xie, Xiaoming Zhang, and Rui Zeng. Relative posture-based kinematic calibration of a 6-rss parallel robot by using a monocular vision system. In *Proceedings of 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Macao, China, December 5–8, 2017.

1.7 Thesis Organization

The rest of this thesis is organized as the followings. In Chapter 2, a literature review of the main topics including the 6-RSS parallel robot, kinematic problems, dynamic modeling, kinematic calibration, dynamic identification and visual seroving of the parallel robot are given. The inverse and forward kinematic models of the 6-RSS parallel robot are built and the velocity analysis is presented in Chapter 3. Then the explicit dynamic model of the parallel robot is derived based on the virtual work theory. In Chapter 4, a relative pose based calibration method based on the optical CMM sensor is proposed. The visual closed-loop output-error identification method for the 6-RSS parallel robot is developed in Chapter 5. Then a dynamic model based visual servoing method for the 6-RSS parallel robot is introduced in Chapter 6. At last, the conclusion and further works are summarized in Chapter 7.

Chapter 2

Literature Review

2.1 Introduction

The research content of this thesis involves the kinematic and dynamic modeling, kinematic calibration, dynamic identification and visual servoing control of the 6-RSS parallel robot based on the visual sensor. In terms of strategies and other aspects, the following is a review of the current research status in related fields.

2.2 6-RSS Parallel Robot

In the field of 6-DOF parallel mechanisms, most researchers have focused on six-spherical-prismatic-spherical (6-SPS) or six-universal-prismatic-spherical (6-UPS) parallel mechanisms, which are so-called Stewart platforms. Both the moving platform and the fixed base platform of Stewart parallel robot are hexagons connected by six SPS or UPS branch chains. Six moving prismatic pairs are used as inputs, and the two ends of the moving pair are connected to the moving and base platforms with spherical pairs respectively. The branch chains are symmetrical, and hence, Stewart platform is also described as 6-SPS or 6-UPS parallel robot. The number 6 represents the number of branches, and SPS represents the branch kinematic chain, which consists of a spherical pair (S), a prismatic pair (P) and a spherical pair (S) connected in series. In fact, almost all existing 6-DOF parallel robot, including those used for entertainment or motion (especially flight) simulators, are based on the 6-SPS or 6-UPS mechanism. However, few people have studied on the 6-RSS type parallel robot which has almost the same characteristic as the Stewart platforms. In the 6-RSS parallel robot, the revolute joint (R) is used as the driving pair, and the spherical pair in each branch can be interchanged with the universal joint (U), which does not change the nature of the mechanism. Therefore, the 6-RSS parallel robot has no difference from the 6-RSU and 6-RSU parallel robot in terms of kinematic analysis. As shown in Figure 2.1a, a 6-RUS parallel machine tool has been developed by Prof. Yukio Takeda of Tokyo Institute of Technology [8]. The most widely used one of this type of parallel robot is the Hexa robot proposed by Pierrot et al. Its variants, such as the Delta robot, have been applied in high speed pick-and-place fields [21]. For commercial applications, Servos & Simulation Inc. has designed an industrial 6-RSS parallel mechanisms which is used in this project. The 6-DOF mobile seat in Figure 2.1b is an application of the robot. The low-cost flight simulator based on the 6-RSS parallel robot is developed by Fidelity Flight Simulation Inc., as shown in Figure 2.1d [10].

Compared with the traditional Stewart platform, the 6-RSS parallel robot has the following main advantages:

(1) Simple mechanism, short transmission chain and fast response;

(2) The heavy motors are installed on the base, so that the weight of the moving parts is reduced;

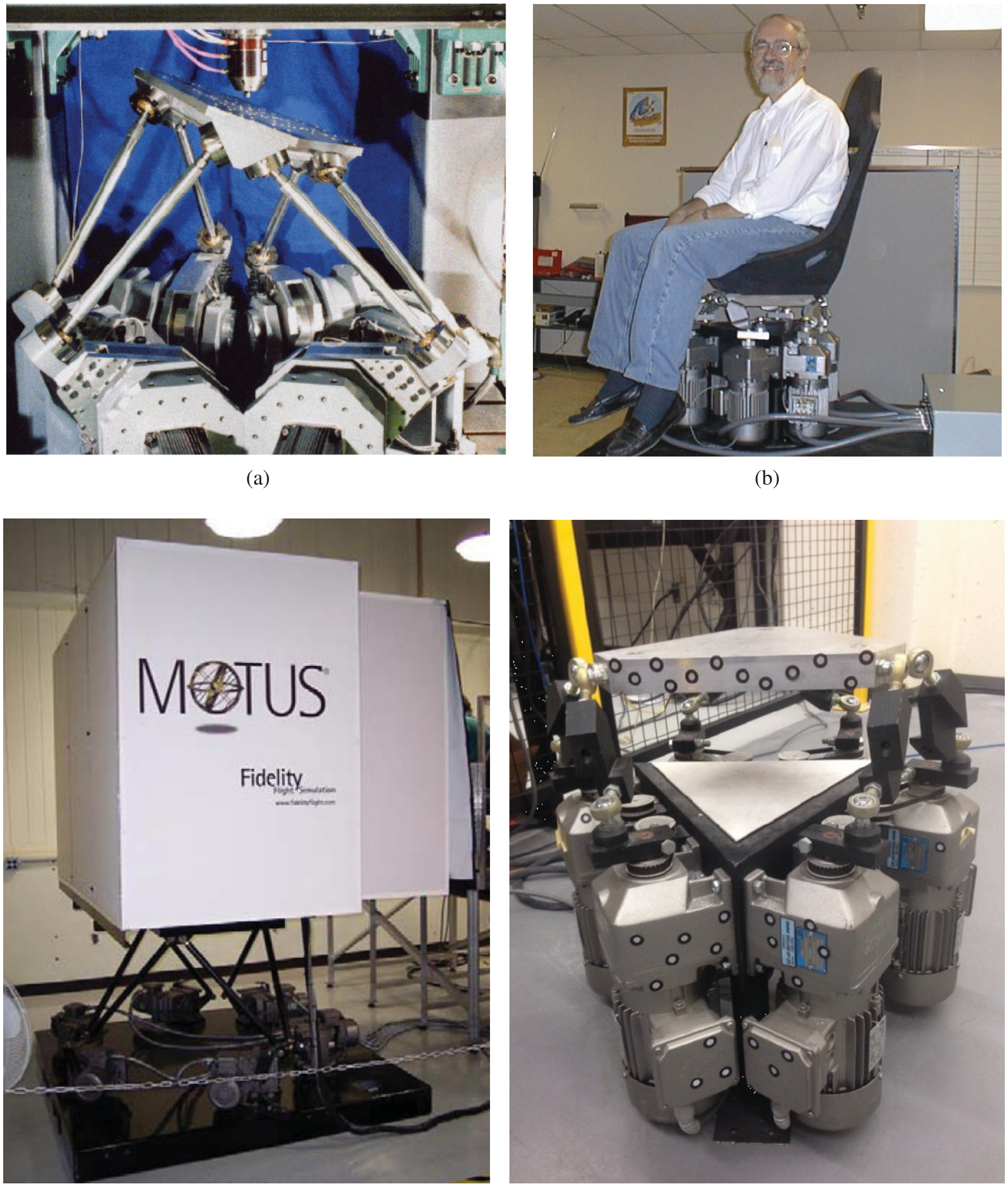
(3) Low cost electric drivers can be used.

The disadvantages associated with 6-RSS robot are listed as following:

(1) Since the legs of the serial chains are more complicated in the 6-RSS parallel robot, the kinematic analysis is complex;

(2) There are more factors affecting the positioning error of the moving platform. And thus it introduces more difficult error analysis.

The 6-RSS parallel robot in Concordia University, as shown in Figure 2.1d, is taken as the case study in this research.



(a) (b)

(c) (d)

Figure 2.1: (a) The 6-RUS robot [8], (b) The 6-DOF mobile seat [9], (c) The 6-DOF flight simulator [10], (d) The 6-RSS parallel robot in Concordia University

2.3 Kinematics of the Parallel Robot

The kinematic study of a 6-DOF parallel robot considers the relationship between the six active joint inputs and the pose of the moving platform. The research content includes two parts: inverse kinematic solution and forward kinematic analysis. The inverse kinematic analysis aims to determine the input of the motion by giving the pose of the platform. The forward kinematic analysis aims to calculate the pose of the platform by taking the active joint inputs as known parameters. The forward and inverse kinematic analysis is the core content of kinematic research of parallel robots, and it is also the theoretical basis for developing algorithms for the accuracy improvement.

The axes of translation and rotation of passive joints in the kinematic chain of a parallel robot are often designed to be intersective, so that the rotation angle or displacement of the passive joints can be eliminated when solving the kinematics equations. At the same time, each kinematic chain is structurally independent of each other, which makes the inverse kinematics problem very simple to obtain analytical solutions. In contrast, due to the combination of the position and attitude of the moving platform, the forward kinematics problem is to solve highly nonlinear equations to obtain the pose parameters, which is quite complicated and has not yet been completely solved [6].

There are two types of methods for solving forward kinematics: analytical methods and numerical methods. In terms of analytical methods, algebraic elimination method, continuous method, interval analysis, etc. are used to transform the kinematics equations into a higher-order polynomial equation, which are committed to finding all possible solutions to the equation [6]. However, so far the forward kinematic solution cannot be expressed as explicit forms of the pose variables [6, 17]. Moreover, finding all possible solutions does not completely solve the problem of forward kinematics, and it is still necessary to further determine the only actual pose in these solutions, which is necessary in practical applications. In some cases, a univariate higher-order algebraic equation or a series of nonlinear equations can be solved by analytical methods, and additional sensors must be used to obtain the unique solution [22, 23]. However, there are limitations in practical applications, due to the expensive measurement device and measurement noises. In terms of numerical methods, the Newton-Raphson method is widely used. This method aims to linearize a series of nonlinear algebraic equations to linear equations. Its convergence domain depends on the nature of the nonlinear equations and if the initial pose of the iteration is in the convergence domain, the exact solution can be derived [6, 24]. Some scholars use neural network algorithms to obtain the initial values required by Newton-Raphson algorithm to ensure the stability of the algorithm [25]. The optimization algorithms such as genetic algorithms and neural network algorithms have been developed to solve kinematic equations to obtain unique solutions [26, 27]. However, those methods can derive the exact solution with a satisfied accuracy only when the initial pose and the target pose are inside a small workspace. And if the target pose is close to the singularity configurations, the solution may converge to a local minimum. In this research, an effective and robust method for solving the forward kinematic solution of the 6-RSS parallel robot needs to be developed for the kinematic calibration purpose.

2.4 Dynamics of the Parallel Robot

The main purpose of dynamic modeling is to study the relationship between the driving force of the robot's motion platform and the motion states (displacement or angle, translation speed or angular velocity, translation acceleration or angular acceleration) of the platform, and to obtain the dynamic equation of the driving forces or torques and motion states, which describes the dynamic characteristics of the robot. The dynamic model is the basis for the design of the robot control system. Similar to the dynamics of a serial robot, the dynamic model of a parallel robot also includes two most basic problems, namely the inverse and forward dynamic solution [6]. The forward dynamic solution is to determine the motion of the parallel robot by knowing the main force or torque acting on the driving joint. On the other hand, if the motion of the reference frame of the end-effector platform is given, the main force or torque of the driving joint is calculated in the inverse dynamic problem. For both problems, the mathematical model of robot dynamics must be built.

The parallel mechanism is a complex spatial closed-chain mechanism. Compared with the serial robot, the number of moving components is doubled, and there exists coupling among the components, which makes the dynamic equation quite complicated. It is necessary for the researchers to establish the dynamic model to carry out the dynamic performance evaluation, dynamic optimization design and real-time control of the parallel robot [11]. The parallel robot theoretically has the advantages of high rigidity, small inertia, large bearing capacity, high speed, high precision, etc. [28]. However, the existence of non-linear relations in parallel structures leads to complex forward position solutions, rich singular configurations, small working space, difficult control implementation, and hence, it is difficult to achieve precise control. This is a bottleneck in developing algorithms to improve the accuracy of parallel robots. The high-precision control of parallel robots requires the introduction of dynamics control, and the establishment of accurate dynamic model is the primary issue for dynamics control. Since the parallel robot is a complex multi-degree of freedom, multi-variable, highly coupling and nonlinear system, its mathematical dynamic model is very complicated and the identification of the parameters of the model poses a great challenge to the researchers.

There are many methods for modeling the dynamics of mechanical systems based on analytical mechanics. Typical methods are Newton-Euler equation method, Lagrange equation method and virtual work method [11]. One important indicator to measure the pros and cons of various dynamics modelings is the amount of computational load to solve the dynamic modeling problem. In general, the Lagrange method has the largest calculation amount, followed by the Newton-Euler method, and the virtual work method has the smallest calculation load with the highest efficiency.

The Lagrange method mainly uses the basic principles of energy conservation law and mathematical tools of Lagrange equations to solve the dynamic modeling problems in the mechanism from the perspective of system energy. The Newton-Euler method mainly uses the Newton's second law and momentum theorem from the perspective of force, which lists the Newton-Euler equation to solve the dynamic equations in the mechanism. Combining the principle of virtual displacement with the principle of D' Alembert, the virtual work method can solve the dynamic modeling problems of systems with ideal constraints. The equation established by this method does not contain the constraint reaction force, and the constraint reaction force can be eliminated.

Each method has its advantages and disadvantages. The Lagrangian method needs to solve the derivative of the generalized coordinate, which increases the difficulty and the amount of calculation. The Newton-Euler method avoids the problem of solving the derivative of the generalized coordinate derivative, and can provide the internal forces for each individual body of the parallel robot, which can benefit the mechanical design process of the parallel robot. However, the computational load is high due to the large amount of equations. The virtual work method does not consider the ideal constraint reaction forces, and only performs the dot product and cross product operation of the vector without requiring derivative operations, which is beneficial to computer-aided operations. Moreover, in the commonly used Newton-Euler method and Lagrange method, it is more difficult to deal with the dynamics of a robot with a closed-chain mechanism. This is because in both methods, the closed-chain mechanism need to be cut into several open chains at certain joints, and to obtain the binding force and torque there. In contrast, the virtual work methods are more efficient and suitable for the control design purpose since the reaction forces between the bodies of the parallel robot are not considered.

The dynamic models of 6-DOF Gough-Stewart parallel robot have been built by several approaches such as Newton-Euler [28], Lagrangian formulation [29] and the principle of virtual work [30]. The published research work on 6-DOF RSS parallel robots is considerably scarcer compared with that on Stewart parallel robot. The dynamic models of one type of 6-DOF RSS parallel robot, in which the active rotation axes are coplanar, are built based on Newton-Euler equations [31] or Lagrangian formulation [32] for dynamic analysis and tracking control purpose respectively. In this project, the dynamic model of a 6-DOF RSS parallel robot, where the active rotation axes are parallel to each other, is built based on the virtual work principle, and the explicit form of the dynamic model is derived for the identification and dynamic model-based visual servoing design purposes.

2.5 Visual Kinematic Calibration of the Parallel Robot

Normally, the positioning accuracy of un-calibrated parallel robots is significantly affected by the manufacturing and robot installation errors. Kinematic calibration can be implemented to remove the negative influence of these errors and to improve the accuracy of end-effector output in a robot control system.

In most applications, the kinematic calibration is known as an optimization problem with redundant non-linear constraint equations. The methods such as classical non-linear algorithms [33, 34], bundle adjustment approach [35] and interval approach [36] are applied to solve it. The basic calibration principle is to construct constraint equations, which reflect the relationship between measured values (could be passive joint values or pose of the end-effector) and the input joint values at various poses, i.e. calibration candidate configurations, until the number of constraints is large enough to determine the errors of geometry parameters contained in constraint equations [6].

For a n DOF robot, at every calibration candidate configuration a set of k constraint equations must be formed as functions of s errors of geometry parameters, of w immeasurable pose parameters and of the measurements at the current configuration. The minimum calibration configuration number m should satisfy: $m \times k \geq s + m \times w$, in order that the constraint equation's number is larger than the number of the unknown parameters. Since there are some noises in the data obtained from sensors and deviations in the robot modeling, the geometry parameters which exactly match the constraint equations cannot be determined. Therefore the numerical analysis method which minimizes the sum of the squares of constraint equations can be employed to determine an optimal solution of the errors of geometry parameters. The principles for the configuration selection of parallel robot calibration have been given in some literature [37, 38], in which the error-parameter Jacobian matrix is utilized to minimize the influence of measurement noise in all candidate configurations.

Although some researchers perform the calibration without using a robot's kinematic model [39], most kinematic calibration methods construct the constraint equations based on the kinematic model [40, 41]. To derive the error parameters in the robot kinematic model, the model-based calibration is conducted in three steps: modeling, measurement and optimization [42].

Based on the geometric analysis, a kinematic error model can be constructed by considering the residual errors in kinematic parameters. Model-based kinematic calibration tries to rebuild a more accurate mapping between robot actuator outputs and the end-effector pose by determining those kinematic parameters. Precise parallel robot error model is built by using D-H method [33]. However, most researchers [43, 44] choose a reduced model in the calibration considering that the contribution of joint manufacturing tolerances has a minor effect on the platform's pose error. In other words, the manufacturing tolerances of the joints are neglectable. Nevertheless, the positional errors of the joint centers and the deviation of the active joint angles are the main reasons for the kinematic calibration.

The measurement sensors play an important role in the parallel robot calibration. It tries to collect enough information for the calibration. The sensors usually fall into two categories: contact measurement and contactless measurement. For the contact measurement type sensors such as translation detector [45], CMM [2], inclinometer [46] and double ball bar device [47], they collect various pose information of the robot end-effector directly for the kinematic calibration. However, they have to meet the strict installation requirements. And the installation errors affect the contact measurement results in different directions and with different magnitudes when the measurand is moving. While for the contactless sensors like camera [43], laser tracker [3] and optical CMM [48], it is more flexible to obtain the pose information of the end-effector. The contactless measurement can eliminate the sensor errors with the help of pre-calibration. Alternatively the sensor location uncertainties can also be viewed as external parameters in terms of the kinematic error model [43], which may increase the complexity of the error model and computation cost. Ideally, the detection of relative pose (the variation of robot position and orientation) is independent of the sensors' location.

The classic implicit calibration method proposed in [49] utilizes the closure relation of the kinematic chains to form implicit constraint equations instead of pursuing the analytical solutions of the closure equations such as the inverse kinematic model. The implicit calibration method emphasizes that the errors are involved in the kinematic loop equations implicitly, rather than being explicit outputs of a conventional input-output formulation. By removing the requirement to express errors explicitly, the formulation allows the analyst to concentrate on all sources of error [49]. And the implicit calibration method has been effectively applied to H4 mechanism [43] and 6-UPS robot.[49] In those implicit calibration methods, the absolute pose of the end-effector w.r.t. the base frame are obtained with the employment of the contactless sensor.

Most researchers assume the sensor location is exactly known in the kinematic calibration experiment [2, 36]. Hence both the absolute and relative poses for the calibration algorithm can be easily determined. However, to derive the pose of the base frame of robots w.r.t. sensor frame is usually a tedious and time consuming work due to the following reasons: (1) the manufacturers usually do not provide enough nominal dimension information of the robots; (2) the self-occlusion of the close structure of parallel robots results in measurement difficulties. If the sensor location is not known exactly, the existing absolute pose-based algorithms cannot be used for the kinematic calibration directly. Ideally the relative pose information, i.e. pose variation, can be utilized in the calibration to avoid the tedious measurement of the relationship between the base frame and the sensor frame. A relative position-based calibration algorithm [45] is carried out for parallel robots, where a simple measurement system with three distance gauges and a ball mounted on the end-effector is employed to measure the relative position movement. However the orientation accuracy cannot be evaluated. Since the gauges should be re-intalled with strict rules in every candidate configuration, the installation errors cannot be removed during the calibration.

In this research, to realize a flexible installation and to avoid the tedious measurement procedure of the sensor location, an optical CMM sensor –C-track 780 from Creaform Inc. is adopted to detect the relative pose of the 6-RSS parallel robot.

2.6 Dynamic Identification of the Parallel Robot

The dynamic parameters are normally unknown or approximately derived from manufacturer specifications, which are not accurate enough for the dynamic model-based controller design. System identification is an effective method to perceive the uncertain parameters in the dynamic model of the system, and has been applied to many engineering practices [50]. As a highly coupled multi-input/multi-output (MIMO) nonlinear system, industrial robots aroused great interest in the identification method. The literatures on the state-of-the-art identification methods can be found in [51, 52, 53]. For the industrial robots, the dynamic identification is normally performed in closed-loop, since the robotic system is open loop unstable. In [54], a MIMO closed-loop identification based on weighted least square estimation has been applied to an industrial serial robot used in a planar configuration. In addition, other closed-loop identification methods with maximum likelihood, instrumental variable and related implementation issues on industrial serial robots are addressed in [55, 56]. A new closed-loop output-error identification scheme has been adopted for the serial robots [57]. The output-error identification method aims at finding the dynamic model parameters by minimizing the output deviation between the actual and simulated systems subjecting to the same input [58]. In [57], the identification procedure is implemented in a closed-loop control structure and the joint torque is the measured output, which avoids the estimation of the velocity and acceleration from the measured joint position.

One potential issue of the above-mentioned identification methods is that joint torque measurement or a related control signal is needed for identification, which is not always available for the industrial robots, since the built-in controllers of many industrial robots are inaccessible and do not provide the torque actuation mode [59]. The input of built-in controllers is the position or velocity command, and the output is the joint torque which is inaccessible to the users. Hence the torque and current of the motors cannot be derived directly and it is not easy to install additional torque sensors to get the direct measurement. The unknown controller can be identified along with the dynamic parameters as introduced in [60]. However, joint torque measurement is still needed for identification purposes.

In [21, 51, 61, 62], identification issues on the dynamic model of parallel robots have been discussed. Most research works on dynamic identification of parallel robots are based on a simplified dynamic model, such as in [63, 64, 65]. Nevertheless, the systematic derivation of the full inverse dynamic model is proposed based on Jourdain's principle in [62]. In addition, the identification procedure is carried out in two steps: 1. identifying inertial parameters and 2. estimating the viscous coefficients. In [61], the full parameters of robots' dynamic model and the joint drive gains are identified based on the total least square method, and the method is tested on a 3-DOF Orthoglide parallel robot. Many identification methods for 6-DOF parallel robots in [21, 61, 62] come directly from those of serial robots. By rewriting the inverse dynamic model and friction model into a linear form with respect to the dynamic and friction parameters, the identification of the unknown parameters can be done through least squares technique. However for 6-DOF parallel robots, to avoid solving forward kinematics, the regression matrix of the dynamic model is constituted of the pose state variables in the Cartesian space. In [21], only joint space state variables are measured and estimated. Therefore, the state variables in Cartesian space should be calculated through numerical computation of the forward kinematics, which is quite time-consuming.

One way to avoid computing the forward kinematics in the identification process is to directly measure the pose of the end-effector in the Cartesian space based on vision sensors. In [65], a camera-based dynamic identification procedure is given for a 4-DOF parallel robot with a heavily simplified dynamic model by considering only the inertia of the end-effector. Also, a vision sensor based dynamic identification has been carried out on serial robots [66] and cable-driven robots [67], but seldom on parallel robots. The optical CMM sensor is a dual camera based vision sensor, which can provide the real-time pose information of the targets. The optical CMM sensor has been applied to the kinematics calibration [68] and the path tracking controller design [20] for the robots. To increase the flexibility and tracking accuracy, vision can also be incorporated into the feedback control loop of the parallel robot systems to form the so-called visual servoing control. The visual servoing controller for parallel robot is superior to the joint space controller due to the fact that the kinematic errors introduced from transforming the desired trajectory in the Cartesian space into the one in the joint space in the joint controller can be avoided. The measured pose together with the visual servoing controller allows using the closed-loop output-error identification method [58] to identify the dynamic model of the parallel robot, and hence will be conducted in this research.

2.7 Visual Servoing Control for the Parallel Robot

How to design effective controllers to improve the positioning and tracking accuracy of the parallel robots poses a challenge to the control community. The existence of closed-chain mechanism yields difficulties on the controller design, related to the fact that the dynamic models of parallel robots are normally more complicated than those of their serial counterparts, since the moving parts are highly coupled and more moving parts are needed to be considered. Moreover there is almost no analytical expression for the forward kinematic model of 6-DOF parallel robots [6, 17]. The joint space controller similar to that of the serial robots is designed for parallel robots by translating the desired pose trajectory of the end-effector frame in Cartesian space into the one in the joint space through the inverse kinematics [21]. This control strategy is easy to be implemented, due to the fact that inverse kinematics of parallel robots can be analytically solved, and the measured joint angles can be used as feedback signal in the joint space control loop. However, it is more desirable to directly control the pose of platform of parallel robot in Cartesian space (i.e. Cartesian space controller) for the following reasons: 1. The pose of the end-effector can be directly controlled to follow the desired trajectory in the Cartesian space control strategy which leads to a better tracking accuracy than controlling the joint space states in the joint space controller, since the desired joint trajectories are transformed from the Cartesian space to the joint space. Such trajectory transformation will introduce deviation caused by inverse kinematic model errors and hence affect the tracking accuracy. 2. The pose control scheme in Cartesian space can also avoid computing the forward kinematics which normally has no analytic solution, provided that the pose can be properly measured or estimated. Then the main challenge lies in how to measure the the pose of the end-effector as the feedback of states and incorporate the measured pose in the feedback control loop.

Visual servoing provides an effective way to control parallel robots' pose with visual sensors. Two main visual servoing control structures, namely image-based visual servoing (IBVS) and position-based visual servoing (PBVS), have been developed for parallel robots [69]. In IBVS, the control signals are directly derived from the coordinates of a set of visual features and computed in the image space [14, 70]. Since the 3D information reconstruction and image interpolation are not employed in this method, the camera calibration is not required and the errors caused by sensor and object modeling are eliminated [71, 72]. However, the singularities, local minima and lost of the image features are the main problems of IBVS due to the lack of the control strategy in Cartesian space. The problems arise especially when the difference between the initial and target views is large [20]. On the other hand, the PBVS control loop directly controls the pose in Cartesian space and allows the direct path planning in the Cartesian space [73]. This feature meets the requirement of industry where the accurate Cartesian path tracking is essential, such as welding, machining, and fiber placement. However, the main obstacle of PBVS lies in estimating the pose information by using the vision information, which usually needs the object model and well-calibrated cameras. Thanks to the optical CMM, the pose information of the target can be measured accurately in real-time with simple calibration steps [20, 68, 74]. Hence, PBVS scheme is a promising method to achieve high accuracy tracking performance of parallel robots for industrial applications.

Most recent visual servoing research work on parallel robots is dedicated to the kinematic level visual servoing design [12, 70, 73, 75, 76, 77]. The dynamics of parallel robots are usually ignored or treated as uncertainties. In [76], a visual servoing controller is designed for the end-point positioning of 3-DOF parallel robot via linear camera-space manipulation. A hidden robot concept is proposed in designing a visual servoing controller for parallel robots based on the leg observations [77]. Although the kinematic level visual servoing of parallel robot can achieve the position tracking to some extent, the tracking performance will be sacrificed greatly without considering the dynamic model of the parallel robot. To include the dynamic model in the visual servoing, the modeling and identification of the dynamics pose a challenge to the control designers. The other issue is the implementation of designed controller in the commercial industrial parallel robots. In most cases, the industrial parallel robots are joint space closed-loop architecture control systems and the built-in controllers of the robots do not provide torque control mode or joint-torque sensors [59, 78]. The built-in controller is inherently position or velocity controlled system. Therefore, the torque command generated by the dynamic level visual servoing controller cannot be directly fed to the built-in controller. To solve this problem, the torque-to-position converter is to be designed. In [79], a P-type torque-to-position converter combined with the backstepping sliding mode control method is designed to apply dynamic model-based controller to the robot manipulators. The proposed torque to position converter in [80] is developed based on the identification of the linear actuator model and is applied to the humanoid robot without using the vision information. And in [59], a similar converter combined with robot dynamics and friction feedforward compensation is utilized to control the humanoid robot. The above-mentioned research work illustrates that the torque-to-position converter has great potential to implement the dynamic level visual servoing control scheme on the position command industrial parallel robots.

For the parallel robots with the open-loop architecture control systems, a lot of dynamic model-based controllers have been developed such as iterative learning controller [21], adaptive controller [81] and robust controller [82]. In [21], the dynamic model identification and the dynamic model-based iterative learning controller are investigated to enhance the tracking performance of a 6-DOF parallel robot. In [81], an adaptive control scheme based on the robust integral of the sign of the error control theory is proposed and tested on a 3-DOF parallel robot. The robust controller for Stewart platform compensates the dynamic model represented in the joint space by transforming the Cartesian space states into the joint space and computing the bounding function in [82]. The robust control such as the sliding mode control (SMC) is known to deal with the uncertainties such as the modeling errors, frictions of the passive joints, sensor noise and time delay. SMC has been widely applied to the joint space position control of the parallel robots [83, 84]. However, the literatures on the SMC based visual servoing of parallel robots are relative scare. In [85], the trajectory tracking scheme of a 4-DOF parallel robot is proposed by utilizing a fuzzy logical SMC scheme and assuming the pose of the end-effector is available in their simulation test. The dynamic model-based controllers are easy to be implemented on the open-loop architecture robotic systems, but it is nontrivial to be implemented on the closed-loop architecture robotic systems. This research aims at developing a dynamic model-based visual servoing controller for the closed-loop architecture industrial parallel robot based on the optical CMM sensor.

2.8 Summary

In this chapter, a brief introduction of the 6-RSS parallel robot regarding its applications, advantages and weakness compared with standard Steward platform is given at first. Then the kinematic problems of the parallel robot including the inverse and the forward solutions are presented. The dynamic modeling methods of the parallel robot are summarized in Section 2.4. The virtual work principle method is chosen to build the dynamic model of the 6-RSS parallel robot for the visual identification and control purpose. Then, the relevant studies on the kinematic calibration method to improve the positioning accuracy of the parallel robot are introduced. With the help of the optical CMM sensor, the relative pose-based calibration method can be developed to derive accurate kinematic parameters, since the method can avoid the tedious measurement of the relationship between the base frame and the sensor frame. In Section 2.6, the literature survey on the dynamic identification methods of the parallel robots are introduced. One of potential issues of the current methods is that the torque measurement of the robot is needed and it is not always available for the industrial robot. At last the visual servoing methods for the parallel robot are discussed. It is found out that most current methods are kinematic level controllers and more advanced dynamic model based visual servoing methods are needed to be further studied to improve the tracking accuracy of the parallel robot.

Chapter 3

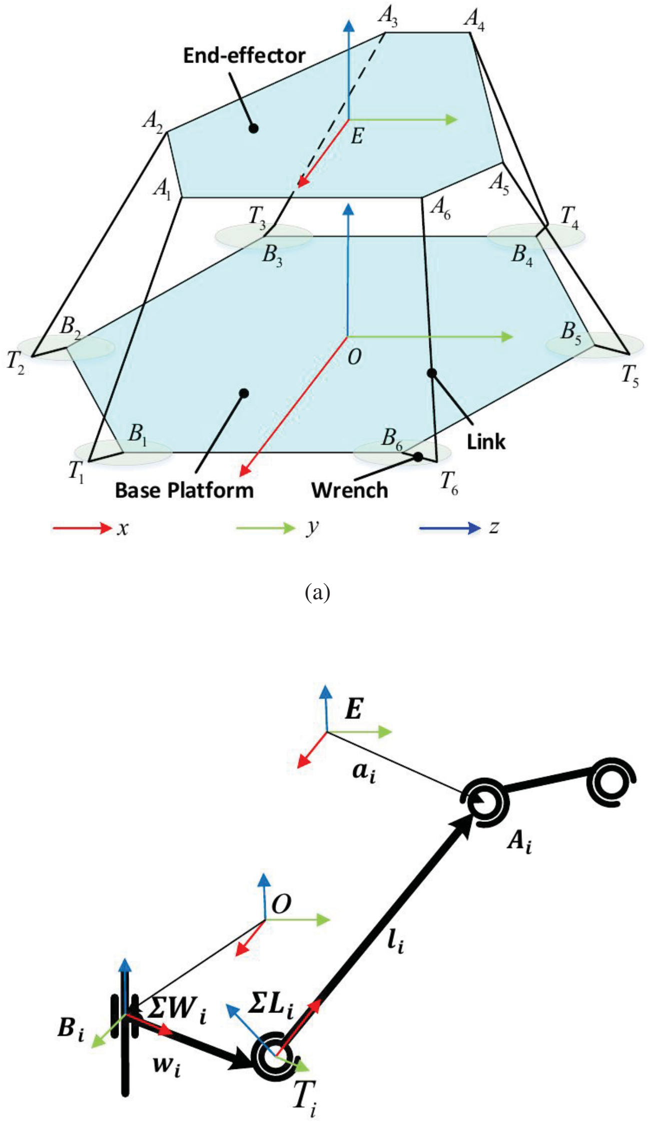
Kinematic Analysis and Dynamic Modeling of the 6-RSS Parallel Robot

3.1 Introduction

The kinematic analysis and dynamic modeling are the foundation of the kinematic calibration, dynamic identification and control of the parallel robot. The inverse kinematic problem of the 6-RSS parallel robot will be solved analytically. A quasi-Stewart method is developed to solve the forward kinematic problem of the 6-RSS parallel robot numerically. By performing the velocity analysis, the Jacobian matrices are derived, which will be used for building the dynamic model and determining the singularity-free and bifurcation-free actuator motion range. The explicit form of the dynamic model will be built based on the virtual work principle for the visual identification and servoing of the 6-RSS parallel robot.

The rest of this chapter is organized as followings. In Section 3.2, a description of the 6-RSS parallel robot is given. Then the inverse and forward kinematic solutions are presented in Section 3.3 and 3.4 respectively. The velocity analysis of the 6-RSS parallel robot is performed in Section 3.5. And the dynamic model of the parallel robot is developed in in Section 3.6. Lastly, a brief summary is given in 3.7.

3.2 Description of the 6-RSS Parallel Robot



(a)

(b)

Figure 3.1: (**a**) The sketch of 6-RSS parallel robot, (**b**) Single serial branch.

The literature survey shows that not much research has been carried out on the actuator-coplanar 6-RSS parallel robot kinematic modeling. In this research, the kinematic model of the parallel robot is built based on the geometrical analysis. As shown in Figure 3.1a, the 6-RSS parallel robot consists of six identical serial branch chains. Each serial branch, illustrated in Figure 3.1b, consists of a wrench, a link, an active revolute joint (R) and two passive spherical joints (S). One spherical joint is used to connect the wrench and the link. The revolute joint is driven by actuators and connects the wrench and the base platform, while the spherical joint is employed between the link and the end-effector. The base frame ΣO is assigned at the symmetric center of the base platform and the end-effector frame ΣE is also attached at the symmetric center, while $\boldsymbol{E} \in \mathbb{R}^3$ denotes the coordinate vector of the frame origin w.r.t. the base frame. The coordinate vectors of revolute joint centers w.r.t. the base frame are marked by $\boldsymbol{B_i} \in \mathbb{R}^3$ $(i = 1,\ 2,\ \dots,\ 6)$, while those of the rotation centers of spherical joints are represented as $\boldsymbol{T_i} \in \mathbb{R}^3$ and $\boldsymbol{A_i} \in \mathbb{R}^3$ respectively. In the subsequent kinematic and dynamic analysis, the coordinates of the parts of parallel robot are defined w.r.t. the base frame by default.

The vector $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_6]^T \in \mathbb{R}^6$ represents the rotation angles of the actuators. The coordinate vector from $\boldsymbol{E}$ pointing to $\boldsymbol{A_i}$ w.r.t. the end-effector frame is denoted as $\boldsymbol{a_i} \in \mathbb{R}^3$. And the coordinate vectors $\boldsymbol{w_i} \in \mathbb{R}^3$ and $\boldsymbol{l_i} \in \mathbb{R}^3$ represent the directions and length of the wrench and link:

$$\boldsymbol{a_i} = \boldsymbol{E} - \boldsymbol{A}_i, \quad \boldsymbol{w_i} \ = \boldsymbol{T}_i - \boldsymbol{B}_i, \quad \boldsymbol{l_i} = \boldsymbol{A}_i - \boldsymbol{T}_i. \tag{3.1}$$

The pose vector of the end-effector frame is expressed as $\boldsymbol{\chi_E} = \left[\boldsymbol{E}^T, \boldsymbol{\phi}^T\right]^T \in \mathbb{R}^6$, where $\boldsymbol{\phi} = [\alpha, \beta, \gamma]^T \in \mathbb{R}^3$ represents the Euler-angle rotation of the frame. The rotation matrix, R $\in SO(3)$, is given by

$$\begin{aligned} R &= R_x(\alpha) R_y(\beta) R_z(\gamma) \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & -s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix} \begin{bmatrix} c\beta & 0 & s\beta \\ 0 & 1 & 0 \\ -s\beta & 0 & c\beta \end{bmatrix} \begin{bmatrix} c\gamma & -s\gamma & 0 \\ s\gamma & c\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} c\beta c\gamma & -c\beta s\gamma & s\beta \\ c\alpha s\gamma + c\gamma s\beta s\alpha & -s\beta s\alpha s\gamma + c\alpha c\gamma & -c\beta s\alpha \\ s\alpha s\gamma - c\alpha c\gamma s\beta & c\alpha s\beta s\gamma + c\gamma s\alpha & c\beta c\alpha \end{bmatrix}, \end{aligned} \tag{3.2}$$

where, s and c stand for sin and cos respectively.

3.3 Inverse Kinematic Modeling of the 6-RSS Robot

The inverse kinematic model reveals the mapping relation from the pose of ΣE to joint angles which can be directly controlled by the actuators. The mapping relation $\mathbb{H}_{inv}$ can be written as

$$\boldsymbol{\theta} = \mathbb{H}_{inv}(\boldsymbol{\chi}_E). \tag{3.3}$$

Therefore, the purpose of inverse kinematic model is to calculate the joint space angles $\boldsymbol{\theta}$ which is the function of the workspace variables $\boldsymbol{\chi}_E$ as known. The inverse kinematic model can be solved analytically as follows.

The coordinate vectors of sphere joint centers with respect to ΣE are written as ${}^{E}\boldsymbol{A}_i \in \mathbb{R}^3$, which are constant and known a priori. Then the coordinate vectors of sphere joint centers with respect to ΣO, $\boldsymbol{A}_i$, can be calculated by

$$\boldsymbol{A}_i = R \cdot {}^{E}\boldsymbol{A}_i + \boldsymbol{E} = [x_{A_i}, y_{A_i}, z_{A_i}]^T \tag{3.4}$$

According to the geometry feature of parallel robot, the lengths of the wrenches and links are constant. Therefore, the following equations can be derived.

$$(x_{T_i} - x_{A_i})^2 + (y_{T_i} - y_{A_i})^2 + (z_{T_i} - z_{A_i})^2 = \parallel \boldsymbol{l}_i \parallel^2 \tag{3.5}$$

$$(x_{T_i} - x_{B_i})^2 + (y_{T_i} - y_{B_i})^2 + (z_{T_i} - z_{B_i})^2 = \parallel \boldsymbol{w}_i \parallel^2 \tag{3.6}$$

where, $\boldsymbol{B}_i = [x_{B_i}, y_{B_i}, z_{B_i}]^T$, $\boldsymbol{T}_i = [x_{T_i}, y_{T_i}, z_{T_i}]^T$, and the operator $\parallel \cdot \parallel$ represents the Euclidean norm. Since the centers of the revolute joints ($\boldsymbol{B}_i$) and sphere joints ($\boldsymbol{T}_i$) locates in the plane $O-xy$, the following constraint exists:

$$z_{T_i} = z_{B_i} = 0. \tag{3.7}$$

By combining Eq.3.5, Eq.3.6 and Eq.3.7, the following relation can be derived as:

$$\begin{aligned} &2x_{T_i}(x_{B_i} - x_{A_i}) + 2y_{T_i}(y_{B_i} - y_{A_i}) + z_{A_i}^2 + x_{A_i}^2 \\ &+y_{A_i}^2 - x_{B_i}^2 - y_{B_i}^2 = \parallel \boldsymbol{l}_i \parallel^2 - \parallel \boldsymbol{w}_i \parallel^2 . \end{aligned} \tag{3.8}$$

Then the coordinate, x_{T_i}, can be solved by

$$x_{T_i} = \frac{N_1}{2(x_{A_i} - x_{B_i})} - \frac{y_{A_i} - y_{B_i}}{x_{A_i} - x_{B_i}} y_{T_i}, \tag{3.9}$$

where,

$$N_1 = - \parallel \boldsymbol{l}_i \parallel^2 + \parallel \boldsymbol{w}_i \parallel^2 + (x_{A_i}^2 + y_{A_i}^2) - (x_{B_i}^2 + y_{B_i}^2) + z_{A_i}^2 \quad . \tag{3.10}$$

With the substitution of x_{T_i} by using Eq.3.9, Eq.3.6 can be written as follows:

$$N_2 y_{T_i}^2 + 2N_3 y_{T_i} + N_4 = 0 \tag{3.11}$$

Then y_{T_i} is obtained:

$$y_{T_i} = \frac{-N_3 \pm \sqrt{N_3^2 - 4N_2 N_4}}{2N_2}, \tag{3.12}$$

where, N_2, N_3 and N_4 are given by:

$$N_2 = \frac{(y_{A_i} - y_{B_i})^2}{(x_{A_i} - x_{B_i})^2} + 1 \tag{3.13}$$

$$N_3 = -2\frac{(y_{A_i} - y_{B_i})}{(x_{A_i} - x_{B_i})}[\frac{N_1}{2(x_{A_i} - x_{B_i})} - x_{B_i}] - 2y_{B_i} \tag{3.14}$$

$$N_4 = (\frac{N_1}{2(x_{A_i} - x_{B_i})} - x_{B_i})^2. \tag{3.15}$$

Then the joint space value $\boldsymbol{\theta}$ can be obtained as Eq.3.16

$$\theta_i = \arctan(\frac{y_{T_i}}{x_{T_i}}) - \theta_{0,i}, \tag{3.16}$$

where $\theta_{0,i}$ denotes the initial position of the $i-th$ revolute joint angle. When $\sqrt{N_3^2 - 4N_2N_4} \geq 0$, the workspace pose is reachable. Furthermore, when the range of θ_i is between $(-\pi/2, \pi/2]$ the solution for y_{T_i} is unique.

3.4 Forward Kinematic Solution of the 6-RSS Paralllel Robot

As discussed in Section 2.3, the forward kinematic solution of the parallel robot is an indispensable part in the kinematic error analysis. In contrast to the serial robot, the forward kinematic solution of the parallel robot is very complicated and has multiple solutions. In most applications, the convergence speed of the forward kinematic algorithm is one of the most important indicators to determine the quality of the algorithm. The forward kinematic problem can be solved through numerical methods like Newton-Raphson method [86], Jacobi method [87] and Powell method [88]. Those methods can derive a relative high accurate forward kinematic solution in a small workspace, assuming that the initial pose of the end-effector is close enough to its target pose. However, when the target pose is close to the singularity configuration of the parallel robot, the updating iteration will converge to a local minimal solution. Even the methods without using Jacobian matrix (Powell method) may be affected by the bifurcation or singularities.

The forward kinematic mapping relation $\mathbb{H}_{fwd}$ can be written as

$$\boldsymbol{\chi}_E = \mathbb{H}_{fwd}(\boldsymbol{\theta}). \tag{3.17}$$

Therefore, the purpose of the forward kinematic solution is to calculate the workspace variables $\boldsymbol{\chi}_E$ as the function of the known input variable $\boldsymbol{\theta}$. A quasi-Stewart method is proposed to solve the forward kinematic solution numerically in this project. By giving the joint space variables $\boldsymbol{\theta}$, the coordinates of $\boldsymbol{T_i}$ can be derived by

$$\begin{aligned} x_{T_i} &= l_{T_iB_i} cos(\theta_i + \theta_{0,i}), \\ y_{T_i} &= l_{T_iB_i} sin(\theta_i + \theta_{0,i}), \\ z_{T_i} &= 0, \end{aligned} \tag{3.18}$$

where $l_{T_iB_i}$ denotes the length of the i^{th} wrench. By assuming the position of $\boldsymbol{T_i}$ as known and fixed, the forward kinematic solution of the 6-RSS parallel robot can be converted to that of a new Stewart platform. In the new Stewart platform, the base terminal is $\boldsymbol{T_i}$ and the end-effector terminal is $\boldsymbol{A_i}$. The target pose to be determined should satisfy that the i^{th} leg length of the new Stewart platform are equal to $l_{A_iT_i}$ which represents the i^{th} link length of the 6-RSS parallel robot.

The current forward kinematic methods use the increment of the $\boldsymbol{\theta}$ to search the correct solution of the pose. Therefore, all the temporary $\boldsymbol{\theta}$ and $\boldsymbol{T_i}$ should be calculated in each iteration, which will increase the computational time. Instead, in the quasi-Stewart method, the increment of the leg length of the new Stewart platform is utilized to search the right solution and the number of the iteration will be reduced. The quasi-Stewart forward kinematic method is implemented according to the following steps:

(1) Set the stop criteria coefficient ε_c for the leg length convergence accuracy and coefficient ε_p for the pose updating condition.

(2) Calculate the coordinates of $\boldsymbol{T_i}$ using the known $\boldsymbol{\theta}$ by Eq. 3.18. Choose the initial pose of the 6-RSS parallel robot.

(3) Determine the pose increment $\Delta\boldsymbol{\chi}_E^t$ of the t^{th} iteration as the followings: Since the $\boldsymbol{T_i}$ and $\boldsymbol{\chi}_E^t$ is known, the coordinate of $\boldsymbol{A}_{\boldsymbol{i}}^t$ can be derived by Eq. 3.4. Then the i^{th} leg direction vector $\boldsymbol{l}_{\boldsymbol{i}}^t = \boldsymbol{A}_{\boldsymbol{i}}^t - \boldsymbol{T_i}$ and the length vector of the new Stewart platform $\boldsymbol{L}^t = [\| \boldsymbol{l}_{\boldsymbol{1}}^t \|, \| \boldsymbol{l}_{\boldsymbol{2}}^t \|, ..., \| \boldsymbol{l}_{\boldsymbol{6}}^t \|]^T$ are obtained.

The Jacobian matrix of the new Stewart platform is given as Eq. 3.19 (t is omitted).

$$J' = \begin{bmatrix} \frac{1}{L_1}\boldsymbol{l_1}^T & \left[\boldsymbol{a_1} \times (\frac{1}{L_1}\boldsymbol{l_1})\right]^T \\ \vdots & \vdots \\ \frac{1}{L_6}\boldsymbol{l_6}^T & \left[\boldsymbol{a_6} \times (\frac{1}{L_6}\boldsymbol{l_6})\right]^T \end{bmatrix}. \tag{3.19}$$

Then $\Delta\boldsymbol{\chi}_E^t$ can be derived by

$$\Delta\boldsymbol{\chi}_E^t = J'^{\,-1}\Delta\boldsymbol{l}^t \tag{3.20}$$

where $\Delta\boldsymbol{l}^t = \boldsymbol{l_{AT}} - \boldsymbol{L}^t$, and $\boldsymbol{l_{AT}} = [l_{A_1T_1}, l_{A_2T_2}, ..., l_{A_6T_6}]^T$.

(4) If $|\Delta \boldsymbol{l}^t| \geqslant \varepsilon_c$ and $|\Delta \boldsymbol{\chi}_E^t| \geqslant \varepsilon_p$, $\boldsymbol{\chi}_E^{t+1} = \boldsymbol{\chi}_E^t + \Delta \boldsymbol{\chi}_E^t$, and then return to step (3) for the next iteration calculation. Otherwise, $\boldsymbol{\chi}_E^t$ is the solution.

The normal forward kinematic method and the quasi-Stewart method are compared in the simulation and the results are given in Figure 3.2 and Figure 3.3. The blue bold solid lines show the initial configuration of the 6-RSS parallel robot, the black dotted lines are the final configurations

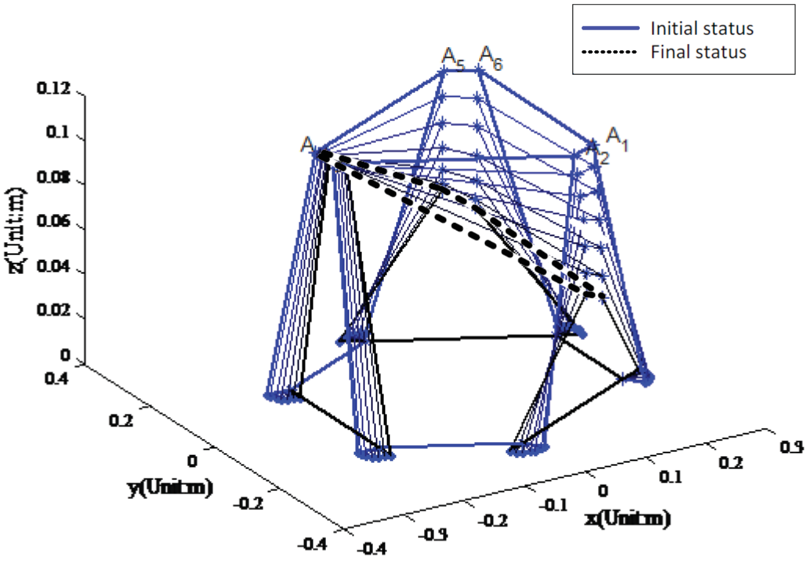


Figure 3.2: The process of normal forward kinematic method

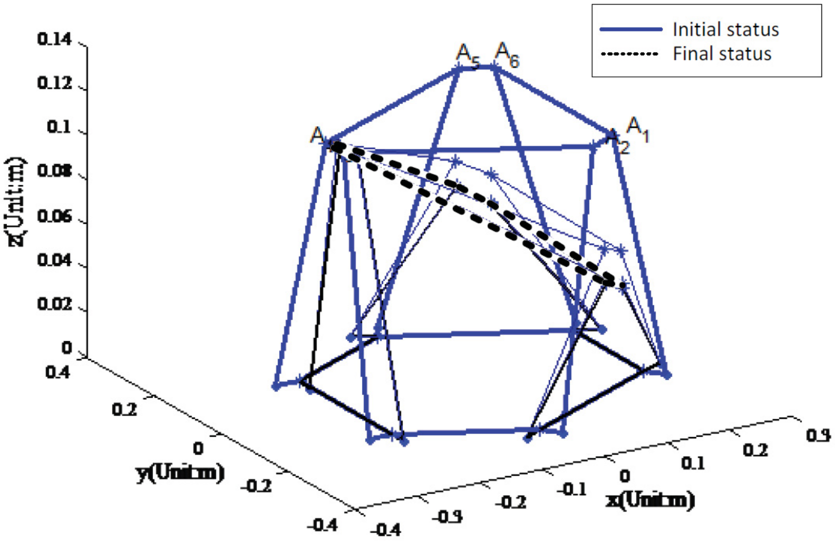


Figure 3.3: The process of quasi-Stewart method

and other lines give the temporary iteration configurations. Although all the temporary iteration configurations can be achieved by the 6-RSS parallel robot, the number of the iteration steps is much more than that in the quasi-Stewart method.

3.5 Velocity Analysis of the 6-RSS Parallel Robot

The inverse and forward kinematic models reveal the static positioning relationship between the end-effector pose and the revolute joint angles. In addition, velocity (differential kinematic) analysis of the parallel robot plays a critical role in the singularity analysis, dynamic model identification and controller design for the parallel robot. The motivation of velocity analysis is to derive the mapping relationships of the end-effector pose velocity with respect to the velocity of the revolute joints, wrenches and links.

The moving wrench and link frames $\varSigma W_i$ and $\varSigma L_i$ are attached to the wrenches and links respectively as depicted in Figure 3.1b. The coordinate vectors of the centers of mass of the wrenches and links are denoted as $\boldsymbol{c_{w_i}}$ and $\boldsymbol{c_{l_i}}$ respectively. The vector $\dot{\boldsymbol{\chi}}_{\boldsymbol{E}} = \left[\dot{\boldsymbol{E}}^T, \dot{\boldsymbol{\phi}}^T\right]^T$ and $\ddot{\boldsymbol{\chi}}_{\boldsymbol{E}} = \left[\ddot{\boldsymbol{E}}^T, \ddot{\boldsymbol{\phi}}^T\right]^T$ are the first and second order time derivatives of $\boldsymbol{\chi}_{\boldsymbol{E}}$. The vector $\boldsymbol{v}_{\boldsymbol{E}} = \left[\dot{\boldsymbol{E}}^T, \boldsymbol{\omega}^T\right]^T \in R^{6\times 1}$ denotes the linear and angular velocities of the end-effector frame. Then $\dot{\boldsymbol{v}}_{\boldsymbol{E}}$ is the acceleration vector. The relationship between the Euler angle rate $\dot{\boldsymbol{\phi}}$ and angular velocity $\boldsymbol{\omega}$ is expressed as follows

$$\boldsymbol{\omega} = J_e\, \dot{\boldsymbol{\phi}}, \tag{3.21}$$

where

$$J_e = \begin{bmatrix} 1 & 0 & s\beta \\ 0 & c\alpha & -c\beta\, s\alpha \\ 0 & s\alpha & c\beta\, c\alpha \end{bmatrix} \tag{3.22}$$

is the analytical Jacobian matrix. Moreover, the transformation relation from $\dot{\boldsymbol{\chi}}_{\boldsymbol{E}}$ to $\boldsymbol{v}_{\boldsymbol{E}}$ can be derived by

$$\boldsymbol{v}_{\boldsymbol{E}} = \begin{bmatrix} E_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & J_e \end{bmatrix} \dot{\boldsymbol{\chi}}_{\boldsymbol{E}} = J_s\, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}}, \tag{3.23}$$

The following assumptions are considered for velocity and dynamic analysis of the 6-RSS parallel robot.

Assumption 1. *The end-effector platform, wrenches and links are symmetric with respect to their axes.*

Assumption 2. *The links ($\overrightarrow{T_iA_i}$) do not rotate about its symmetric axes, and the frictions of the spherical joints (T_i and A_i) are ignored.*

As shown in Figure 3.1a and 3.1b, the closure loop position relationship between the end-effector frame and the base frame can be expressed as the following:

$$\boldsymbol{E} + \boldsymbol{a_i} - \boldsymbol{A_i} = 0 \quad and \tag{3.24}$$

$$\boldsymbol{B_i} + \boldsymbol{w_i} + \boldsymbol{l_i} - \boldsymbol{A_i} = 0, i = 1, 2, \ldots, 6. \tag{3.25}$$

Then the linear velocity can be derived by combining Eq. 3.24 and 3.25 and taking the time derivative:

$$\dot{\boldsymbol{E}} + \boldsymbol{\omega} \times \boldsymbol{a_i} = \boldsymbol{\omega_1} \times \boldsymbol{w_i} + \boldsymbol{\omega_2} \times \boldsymbol{l_i}, i = 1, 2, \ldots, 6, \tag{3.26}$$

where $\boldsymbol{\omega_1}$ and $\boldsymbol{\omega_2}$ are the angular velocity of frame ΣW_i and ΣL_i regarding ΣO respectively.

Then, dot multiplying $\boldsymbol{l_i}$ on both sides of Eq. 3.26 yields:

$$\boldsymbol{l_i}\dot{\boldsymbol{E}} + (\boldsymbol{a_i} \times \boldsymbol{l_i})\boldsymbol{\omega} = (\boldsymbol{w_i} \times \boldsymbol{l_i}) \cdot \boldsymbol{\omega_1}, i = 1, 2, \ldots, 6. \tag{3.27}$$

Since the wrench rotates around a fixed axis which is denoted as $\hat{\boldsymbol{s}} = [0, 0, 1]^T$, the Jacobian matrix mapping from the end-effector Cartesian velocity to joint velocity can be derived by the following:

$$J_{a1}\dot{\boldsymbol{\theta}} = J_{a2}\boldsymbol{v_E}, \tag{3.28}$$

where

$$\begin{aligned} J_{a1} =& diag((\boldsymbol{w_1} \times \boldsymbol{l_1}) \cdot \hat{\boldsymbol{s}}, (\boldsymbol{w_2} \times \boldsymbol{l_2}) \cdot \hat{\boldsymbol{s}}, \ldots, (\boldsymbol{w_6} \times \boldsymbol{l_6}) \cdot \hat{\boldsymbol{s}}), \\ J_{a2} =& \begin{bmatrix} \boldsymbol{l_1}^T & (\boldsymbol{a_1} \times \boldsymbol{l_1})^T \\ \vdots & \vdots \\ \boldsymbol{l_6}^T & (\boldsymbol{a_6} \times \boldsymbol{l_6})^T \end{bmatrix}. \end{aligned} \tag{3.29}$$

When the robot works in the singularity-free operational space, the Jacobian matrix J_{ad} can be derived as follows:

$$\dot{\boldsymbol{\theta}} = J_{a1}^{-1} J_{a2} \boldsymbol{v_E} = J_{ad} \boldsymbol{v_E}. \tag{3.30}$$

Then the translational velocity of the center of mass of the wrench $\dot{\boldsymbol{c_{w_i}}}$ can be obtained from Equation 3.31.

$$\dot{\boldsymbol{c_{w_i}}} = \dot{\theta}_i \hat{\boldsymbol{s}} \times \boldsymbol{c_{w_i}} = J_{au} \boldsymbol{v_E}, \tag{3.31}$$

where

$$J_{au} = (\hat{\boldsymbol{s}} \times \boldsymbol{c_{w_i}}) J_{adi} \in \mathbb{R}^{3\times 6}, \tag{3.32}$$

and J_{adi} is the i^{th} row of J_{ad}.

Then the link Jacobian J_a mapping $\boldsymbol{v_E}$ to the velocity of the center of mass of the ith wrench $\boldsymbol{v_1} = [\dot{\boldsymbol{c_{w_i}}}^T, \boldsymbol{\omega}_{1i}^T]^T$ can be derived as

$$\boldsymbol{v_1} = \begin{bmatrix} \dot{\boldsymbol{c}}_{w_i} \\ \boldsymbol{\omega_{1i}} \end{bmatrix} = \begin{bmatrix} J_{au} \\ \hat{\boldsymbol{s}} J_{adi} \end{bmatrix} \boldsymbol{v_E} = J_a \boldsymbol{v_E}. \tag{3.33}$$

Eq. 3.34 can be deduced by right cross multiplying $\boldsymbol{l_i}$ on the both sides of Eq. 3.26 and using Lagrange's rule.

$$(\boldsymbol{\omega_2} \cdot \boldsymbol{l_i}) \cdot \boldsymbol{l_i} - (\boldsymbol{l_i} \cdot \boldsymbol{l_i}) \cdot \boldsymbol{\omega_2} = \dot{\boldsymbol{E}} \times \boldsymbol{l_i} + (\boldsymbol{\omega} \times \boldsymbol{a_i}) \times \boldsymbol{l_i} - (\boldsymbol{\omega_1} \times \boldsymbol{w_i}) \times \boldsymbol{l_i}. \tag{3.34}$$

Since the link does not rotate about its longitudinal axis, it is inferred that

$$\boldsymbol{\omega_2} \cdot \boldsymbol{l_i} = 0 \;\; holds. \tag{3.35}$$

Rearranging Eq. 3.34, the following equation can be derived:

$$\parallel \boldsymbol{l_i} \parallel^2 \boldsymbol{\omega_2} = [\boldsymbol{l_i}]_X \dot{\boldsymbol{E}} - [\boldsymbol{l_i}]_X \, [\boldsymbol{a_i}]_X \, \boldsymbol{\omega} + [\boldsymbol{l_i}]_X \, [\boldsymbol{w_i}]_X \, \boldsymbol{\omega_1}, \tag{3.36}$$

where the operator $[\cdot]_X$ and $\parallel \cdot \parallel$ represents the cross product operation and Euclidean norm respectively.

By combining Eq. 3.30 and 3.36, the following Jacobian matrix J_{bd} mapping from the end-effector Cartesian velocity to the angular velocity of the link frame can be deduced:

$$\begin{aligned} \boldsymbol{\omega_2} &= J_{bd}\boldsymbol{v_E}, \\ J_{bd} &= \frac{1}{\parallel \boldsymbol{l_i} \parallel^2}\{\begin{bmatrix} [\boldsymbol{l_i}]_X & -[\boldsymbol{l_i}]_X[\boldsymbol{a_i}]_X \end{bmatrix} + [\boldsymbol{l_i}]_X[\boldsymbol{w_i}]_X \hat{\boldsymbol{s}} J_{adi}\}. \end{aligned} \tag{3.37}$$

The translational velocity of the center of mass of the link $\dot{\boldsymbol{c_{l_i}}}$ can be obtained as follows:

$$\dot{\boldsymbol{c_{l_i}}} = -[\boldsymbol{w_i}]_X\boldsymbol{\omega_1} - [\boldsymbol{c_{l_i}}]_X\boldsymbol{\omega_2} \tag{3.38}$$

By substituting Eq. 3.33 and 3.37 into Eq. 3.38, the following velocity relationship can be derived:

$$\dot{\boldsymbol{c_{l_i}}} = (-[\boldsymbol{w_i}]_X \hat{\boldsymbol{s}} J_{adi} - [\boldsymbol{c_{l_i}}]_X J_{bd})\boldsymbol{v_E} = J_{bu}\boldsymbol{v_E} \tag{3.39}$$

Hence the link Jacobian J_b mapping $\boldsymbol{v_E}$ to the velocity of the center of mass of the i^{th} link $\boldsymbol{v_2}$ can be derived.

$$\boldsymbol{v_2} = \begin{bmatrix} \dot{\boldsymbol{c_{l_i}}} \\ \boldsymbol{\omega_{2i}} \end{bmatrix} = \begin{bmatrix} J_{bu} \\ J_{bd} \end{bmatrix} \boldsymbol{v_E} = J_b\boldsymbol{v_E} \tag{3.40}$$

3.6 Dynamic Modeling of the 6-RSS Parallel Robot

In contrast to serial robots, the dynamic modeling of parallel robots is more complicated due to its closure geometrical structure and difficulties in deriving the forward kinematics. The principle of virtual work is employed to derive the explicit form of the dynamic model in terms of the workspace coordinates and their time derivatives as shown in Eq. 3.41 for 6-DOF RSS robot, which is useful for dynamic model-based controller design.

$$\boldsymbol{\tau_g} = M(\boldsymbol{\chi_E})\dot{\boldsymbol{v}}_{\boldsymbol{E}} + C(\boldsymbol{\chi_E}, \boldsymbol{v_E})\boldsymbol{v_E} + G(\boldsymbol{\chi_E}) + \boldsymbol{\tau_f}, \tag{3.41}$$

where $\boldsymbol{\tau_g}$ denotes the general force acting on the end-effector frame, $\boldsymbol{\tau_f}$ is the friction, $M(\boldsymbol{\chi_E})$ is the mass matrix, $C(\boldsymbol{\chi_E}, \boldsymbol{v_E})$ is Coriolis and centrifugal matrix, and $G(\boldsymbol{\chi_E})$ denotes the gravity. In order to avoid solving the forward kinematics of the parallel robot which may not have analytical solutions, the pose in the coordinates of the end-effector and its time derivatives are employed in Eq. 3.41.

The balance equation of virtual work principle for one moving rigid body of the end-effector platform, wrenches and links, $*$, can be expressed as follows:

$$\bar{\boldsymbol{F}}_* \cdot \delta\boldsymbol{\chi}_* = (\boldsymbol{F_{ext_*}} + \tilde{\boldsymbol{F}}_*) \cdot \delta\boldsymbol{\chi}_* = 0, \tag{3.42}$$

in which $\bar{\boldsymbol{F}}_*$ contains the static balance force and torque, $\boldsymbol{F_{ext_*}} = [\boldsymbol{f}_{\boldsymbol{ext_*}}^T, \boldsymbol{\tau}_{\boldsymbol{ext_*}}^T]^T$ is the external force ($\boldsymbol{f_{ext_*}}$) and torque ($\boldsymbol{\tau_{ext_*}}$) exerted on the center of mass of the body respectively, $\delta\boldsymbol{\chi}_*$ denotes the virtual displacement, and the fictitious force and torque are:

$$\tilde{\boldsymbol{F}}_* = \begin{bmatrix} m_*\boldsymbol{g} - m_*\ddot{\boldsymbol{h}}_* \\ -(I_*\dot{\boldsymbol{\omega}}_* + \boldsymbol{\omega}_* \times I_*\boldsymbol{\omega}_*) \end{bmatrix}, \tag{3.43}$$

where m_* is the mass of the body, $\boldsymbol{g}$ is the gravity vector, $\ddot{\boldsymbol{h}}_*$ is the linear acceleration of center of mass of the body, I_* is the moment of inertia, $\boldsymbol{\omega}_*$ and $\dot{\boldsymbol{\omega}}_*$ denote the angular velocity and acceleration of the moving body frame. Further, $\bar{\boldsymbol{F}}_*$ can be represented in the form similar to Eq. 3.41:

$$\begin{aligned} &\bar{\boldsymbol{F}}_* = \boldsymbol{F_{ext_*}} + M_*\dot{\boldsymbol{v}}_* + C_*\boldsymbol{v}_* + G_*,\ where \\ &M_* = \begin{bmatrix} -m_*E_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & -I_* \end{bmatrix}, C_* = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & -\boldsymbol{\omega}_* \times I_* \end{bmatrix}, \\ &G_* = \begin{bmatrix} m_*\boldsymbol{g} \\ \mathbf{0}_{3\times1} \end{bmatrix}, \dot{\boldsymbol{v}}_* = \begin{bmatrix} \ddot{\boldsymbol{h}}_* \\ \dot{\boldsymbol{\omega}}_* \end{bmatrix}, \boldsymbol{v}_* = \begin{bmatrix} \dot{\boldsymbol{h}}_* \\ \boldsymbol{\omega}_* \end{bmatrix} \end{aligned} \tag{3.44}$$

in which $E_{3\times3} \in R^{3\times3}$ denotes the identity matrix. For a 6-DOF RSS parallel robot, there are 13 moving bodies including the end-effector, 6 wrenches and 6 links. Therefore the balance equation of the 6-DOF parallel robot can be rewritten as Eq. 3.45:

$$\bar{\boldsymbol{F}}_p \cdot \delta\boldsymbol{\chi_e} + \sum_{i=1}^{6} \bar{\boldsymbol{F}}_{l_i} \cdot \delta\boldsymbol{\chi_{l_i}} + \sum_{i=1}^{6} \bar{\boldsymbol{F}}_{w_i} \cdot \delta\boldsymbol{\chi_{w_i}} = 0, \tag{3.45}$$

where $\bar{\boldsymbol{F}}_p$, $\bar{\boldsymbol{F}}_{l_i}$ and $\bar{\boldsymbol{F}}_{w_i}$ contain the static balance force and torque exerted on the centers of mass of the platform, links and wrenches respectively and can be represented in the same form as Eq. 3.44, $\delta\boldsymbol{\chi_e}$, $\delta\boldsymbol{\chi_{l_i}}$, and $\delta\boldsymbol{\chi_{w_i}}$ are the virtual displacements accordingly.

In addition, the following relations hold for the velocity analysis:

$$\delta\boldsymbol{\theta} = J_{ad}\delta\boldsymbol{\chi_e}, \;\; \delta\boldsymbol{\chi_{w_i}} = J_a\delta\boldsymbol{\chi_e}, \;\; \delta\boldsymbol{\chi_{l_i}} = J_b\delta\boldsymbol{\chi_e}. \tag{3.46}$$

Substituting Eq. 3.46 into Eq. 3.45, the terms in Eq. 3.41 can be derived as the following:

$$\begin{aligned}
M(\boldsymbol{\chi_E}) =& M_p + \sum_{i=1}^{6}(J_{a_i}^T M_{w_i} J_{a_i} + J_{b_i}^T M_{l_i} J_{b_i}),\\
C(\boldsymbol{\chi_E}, \boldsymbol{v_E}) =& C_p + \sum_{i=1}^{6}(J_{a_i}^T C_{w_i} J_{a_i} + J_{a_i}^T M_{w_i} \dot{J}_{a_i} + J_{b_i}^T C_{l_i} J_{b_i} + J_{b_i}^T M_{l_i} \dot{J}_{b_i}),\\
G(\boldsymbol{\chi_E}) =& G_p + \sum_{i=1}^{6}(J_{a_i}^T G_{w_i} + J_{b_i}^T G_{l_i}),\\
\boldsymbol{\tau_g} =& J_{ad}^T \boldsymbol{\tau_a},
\end{aligned} \tag{3.47}$$

where $\boldsymbol{\tau_a} = [\tau_{a_1}, \tau_{a_2} ... \tau_{a_6}]^T$ is the actuator torque vector applying on the revolute joints, and the details of Eq. 3.47 are given in Appendix 7.2. The joint friction is described by Coulomb model [89] that has been used in the modelings of both parallel robots [62] and serial robots. Based on this friction model, the friction $\boldsymbol{\tau_f}$ in Eq. 3.41 can be represented as:

$$\boldsymbol{\tau_f} = J_{ad}^T \begin{bmatrix} f_{c_1} sign(J_{ad_1} \boldsymbol{v_E}) + f_{v_1} J_{ad_1} \boldsymbol{v_E} \\ f_{c_2} sign(J_{ad_2} \boldsymbol{v_E}) + f_{v_2} J_{ad_2} \boldsymbol{v_E} \\ \dots \\ f_{c_6} sign(J_{ad_6} \boldsymbol{v_E}) + f_{v_6} J_{ad_6} \boldsymbol{v_E} \end{bmatrix}, \tag{3.48}$$

where f_{c_i} and f_{v_i} are the Coulomb and viscous friction parameters of the i^{th} revolute joint.

3.7 Summary

In this chapter, the inverse kinematic problem of the 6-RSS parallel robot is solved analytically. A quasi-Stewart forward kinematic method is developed for the 6-RSS parallel robot. The superiority of the method compared with the normal forward kinematic method is presented by the simulation. The kinematic solutions will be used to determine the proper workspace and to develop the kinematic calibration method. The Jacobian matrices reveling the velocity mappings from the joint space, wrench and link frame velocity to the workspace velocity of the 6-RSS parallel robot are developed for building the dynamic model. The explicit form of the dynamic model of the 6-RSS parallel robot is obtained based on the virtual work principle. The built dynamic model will be utilized to design the identification method and advanced dynamic model based controller for the 6-RSS parallel robot.

Chapter 4

Visual Kinematic Calibration Method for the 6-RSS Parallel Robot

4.1 Introduction

Kinematic calibration is necessary to be carried out to remove the kinematic parameter errors which are caused by the manufacturing and installation process of the robot, and to improve the positioning accuracy of end-effector output in a robot control system. Based on the geometric analysis, a kinematic error model can be constructed by considering joint residual errors in kinematic parameters. Instead of using a full robot error model built by D-H method in [33], a reduced error model will be considered in this chapter, since the contribution of joint manufacturing tolerances has a minor effect on the platform pose error. The reduced error model will emphasize the positional errors of the joint centers and the deviation of the active joint angles. The external sensors will be utilized to collect enough redundant information for the calibration which is posed as a nonlinear optimization problem. In this thesis research, the optical CMM sensor, C-track, will be used to measure the end-effector pose of the parallel robot. By attaching several target reflectors on the end-effector platform, the 6D pose of the end-effector frame can be derived by the optical CMM sensor which is more flexible compared with traditional contact measurement sensors such as the translation detector, CMM and inclinometer. The calibration algorithms based on end-effector absolute pose are generally used to determine the optimal robot kinematic model [42]. This kind of algorithm needs accurate absolute location of the robot base frame in the sensor frame through a tedious pre-calibration procedure due to the lack of a well-defined and mechanically accessible base coordinate frame for the robot [90]. The location of the robot base frame w.r.t. the sensor frame is not needed in the relative pose (or pose variation) based calibration. And based on the relative pose measurement, the constraint equations for deriving the entire considered kinematic parameters of the robot can be constructed for the calibration. Therefore in this research, we focus on the kinematic model calibration based on robot relative pose.

In this chapter, a relative pose-based calibration algorithm (RPBA) based on the optical CMM sensor is developed for a 6-RSS parallel robot. The detected feature points of the reflectors can be used to estimate the relative poses. The obtained relative poses are then used to construct an objective function, and the updating algorithm is determined by minimizing this objective function following the least square norm principle. The actuation stroke range and optimal set of candidate configurations are determined for the kinematic calibration of the 6-RSS parallel robot. Simulation has been carried out to prove the superiority of relative pose based calibration method comparing with the implicit calibration method based on absolute pose measurement. The experimental tests show that the following advantages of the proposed algorithm comparing with the other relative position-based algorithm [45]: both the position and orientation variations can be utilized and no accurate location information for the detection sensor is needed. Both simulation and experimental results demonstrate that the proposal RPBA using optical CMM is an implementable and effective method for the parallel robot calibration.

This chapter is organized as follows. The kinematic error model and the visual detection system are introduced in section 4.2. Section 4.3 presents the classic implicit calibration method and the RPBA based on the optical CMM. In section 4.4, simulation results on optimal actuator stroke and calibration are presented. Finally, experimental case studies for the 6-RSS parallel robot kinematic calibration are given in section 4.5, and a brief summary is drawn in section 4.6.

4.2 Kinematic Error Model and Pose Estimation

In this section, the kinematic error model is built based on the 6-RSS kinematic analysis. And the optical CMM system for the robot pose detection is introduced.

4.2.1 Kinematic Error Analysis

Fig.4.1 shows the 6-RSS parallel robot with six coplanar rotary actuators. There is only one actuator in each parallel robot chain, and the kinematic model can be described by 6 equations given as follows:

$$l_{A_iT_i} = \left\| \mathbf{T}_i(\theta_i, l_{T_iB_i}, \mathbf{B}_i) - \mathbf{A}_i(\boldsymbol{\chi}_E, \mathbf{A}_i') \right\| \quad (i = 1, ..., 6), \tag{4.1}$$

where $l_{A_iT_i}$, $l_{T_iB_i}$ are constant lengths of wrenches and links in the i^{th} chain respectively, and $\mathbf{A}_i' \in \mathbb{R}^3$ denotes the coordinate of the sphere joint center (A_i) with respect to ΣE. $\mathbf{T}_i(\theta_i, l_{T_iB_i}, \mathbf{B}_i)$ represents the calculation process of $\mathbf{T}_i$ through θ_i, $l_{T_iB_i}$, and $\mathbf{B}_i$, similarly, $\mathbf{A}_i(\boldsymbol{\chi}_E, \mathbf{A}_i')$ is the calculation process of $\mathbf{A}_i$ through $\boldsymbol{\chi}_E$ and $\mathbf{A}_i'$.

For a complete error modeling of 6-UPS robots, 132 geometric errors parameters are identified by Masory [33]. The geometric parameters can be reduced to 42 assuming a good manufacturing quality is applied to the joints. The reduced error model is introduced in Wang's result [1], which shows that the position accuracy of Stewart platform is insensitive to the joint errors. The reduced kinematic error model is considered in this chapter. Notice that the revolute joint values θ_i are measured by built-in potentiometers of the 6-RSS parallel robot. The linear relationship, Eq. 4.2 ,

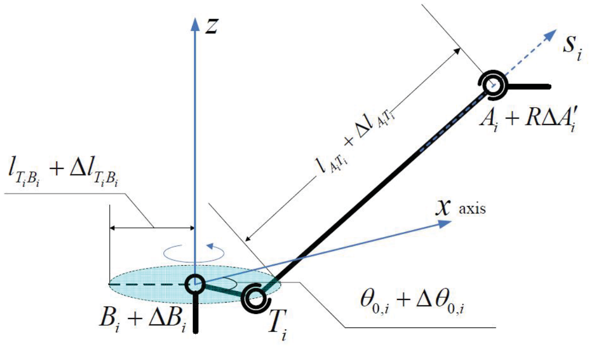


Figure 4.1: Error parameters considered in the model

can be used to compute θ_i [91].

$$\theta_i = \eta_i \kappa_i + \mathit{\Delta}\theta_i \quad (i = 1, ..., 6), \tag{4.2}$$

where, $\mathit{\Delta}\theta_i$ are the angle offsets, κ_i are the sensor outputs, and η_i are the angle conversion coefficients. Based on the kinematic analysis, the considered parameters include: initial terminal coordinate errors $\mathit{\Delta}\mathbf{A}'_i = [\mathit{\Delta}x'_{ai}, \mathit{\Delta}y'_{ai}, \mathit{\Delta}z'_{ai}]^T$, $\Delta\mathbf{B}_i = [\mathit{\Delta}x_{bi}, \mathit{\Delta}y_{bi}, \mathit{\Delta}z_{bi}]^T$, length error $\mathit{\Delta}l_{A_iT_i}$, $\mathit{\Delta}l_{T_iB_i}$, angle conversion coefficients η_i, and angle offset $\mathit{\Delta}\theta_i$, where $i = 1, 2, ..., 6$. Fig.4.1 shows the kinematic structure of the i^{th} chain for the parallel robot with errors marked. Hence, 60 unknown parameters are considered in the kinematic model. And those parameters are denoted as the column vector

$$\begin{aligned}\mathbf{b}_k =&[\mathit{\Delta}x'_{a1}; \mathit{\Delta}y'_{a1}; \mathit{\Delta}z'_{a1}; \ldots; \mathit{\Delta}x'_{a6}; \mathit{\Delta}y'_{a6}; \mathit{\Delta}z'_{a6}; \mathit{\Delta}x_{b1}; \mathit{\Delta}y_{b1}; \mathit{\Delta}z_{b1}; \ldots; \mathit{\Delta}x_{b6}; \mathit{\Delta}y_{b6}; \mathit{\Delta}z_{b6};\\ &\mathit{\Delta}l_{A_1T_1}; \ldots; \mathit{\Delta}l_{A_6T_6}; \mathit{\Delta}l_{T_1B_1}; \ldots; \mathit{\Delta}l_{T_6B_6}; \eta_1; \ldots; \eta_6; \mathit{\Delta}\theta_1; \ldots; \mathit{\Delta}\theta_6] = [b_1, b_2, ...b_{60}]^T.\end{aligned}$$

The kinematic error model is then given as follows:

$$\begin{aligned} l_{A_iT_i} + \mathit{\Delta}l_{A_iT_i} =& \| \mathbf{T}_i(\eta_i\kappa_i + \mathit{\Delta}\theta_i, l_{T_iB_i} + \mathit{\Delta}l_{T_iB_i}, \mathbf{B}_i + \mathit{\Delta}\mathbf{B}_i) \\ &- \mathbf{A}_i\left(\boldsymbol{\chi}_E, \mathbf{A}'_i + \mathit{\Delta}\mathbf{A}'_i\right) \| \quad (i = 1, ..., 6). \end{aligned} \tag{4.3}$$

4.2.2 Pose Estimation Using the Optical CMM

The kinematic calibration of the parallel robot can be formulated as an optimization problem with redundant non-linear constraint equations. As shown in Fig. 4.2, a dual-camera optical CMM (C-track 780) is employed to estimate the pose of end-effector as redundant data for the optimization problem in this research. The pose estimation principle of binocular vision is presented in this section. n reflectors ($n > 3$) placed on the robot are chosen as target feature points to form the target frame $\mathit{\Sigma}t$. Given a group of non-collinear feature points p_i $(i = 1, 2, ...n)$, whose homogeneous coordinates values in the sensor frame is denoted by ${}^C\mathbf{P}_i = [x_{pi}, y_{pi}, z_{pi}, 1]^T$, the relative pose ${}^C\boldsymbol{\chi}_t$ between the target frame and the sensor frame $\mathit{\Sigma}C$ can be estimated.

The projection coordinates of ${}^C\mathbf{P}_i$ on the image frame of each camera can be written as ${}^I\mathbf{P}_{ij} = (u_{ij}, v_{ij}, 1)$, $i = 1 \cdots n$ and $j = 1, 2$, where j is the number of dual cameras. ${}^IW_j$, is the projection matrix of each camera. The perspective projection can be given as below [92]:

$$ {}^{I}\mathbf{P}_{ij} = {}^{I}W_j \, {}^{C}\mathbf{P}_i, \quad {}^{I}W_j = B_j \ \ {}^{I}M_{Cj}, \tag{4.4} $$

where B_j is the camera matrix, including the intrinsic parameters of the j^{th} camera; ${}^{I}M_{Cj}$ is the homogeneous transformation matrix from the sensor frame ΣC to the j^{th} image frame, B_j and ${}^{I}M_{Cj}$ can be derived through calibration of the optical sensor. Due to the lens distortion, calibration errors and other uncertainties, the ${}^{C}\mathbf{P}_i$ derived from each camera is different. The triangulation is the main way to balance the difference in the results [93]. In order to ensure a matching pair of points, ${}^{I}\mathbf{P}_{i1}$ and ${}^{I}\mathbf{P}_{i2}$, meet in space, the following constraint should be satisfied:

$$ {}^{I}\mathbf{P}_{i1}^{T} \, G \, {}^{I}\mathbf{P}_{i2} = 0, \tag{4.5} $$

where G is the fundamental matrix that can be computed when dual camera projection matrices, ${}^{I}M_1$ and ${}^{I}M_2$, are given. Due to the uncertainty of image processing, Eq. 4.5 may not be satisfied accurately. According to optimal correction principle of Kanatani [94], the objective function is

$$ \min_{{}^{C}\hat{\mathbf{P}}_{i1}{}^{T} G \, {}^{C}\hat{\mathbf{P}}_{i2}=0} \left(d({}^{C}\mathbf{P}_{i1}, {}^{C}\hat{\mathbf{P}}_{i1}) + d({}^{C}\mathbf{P}_{i2}, {}^{C}\hat{\mathbf{P}}_{i2})\right), \tag{4.6} $$

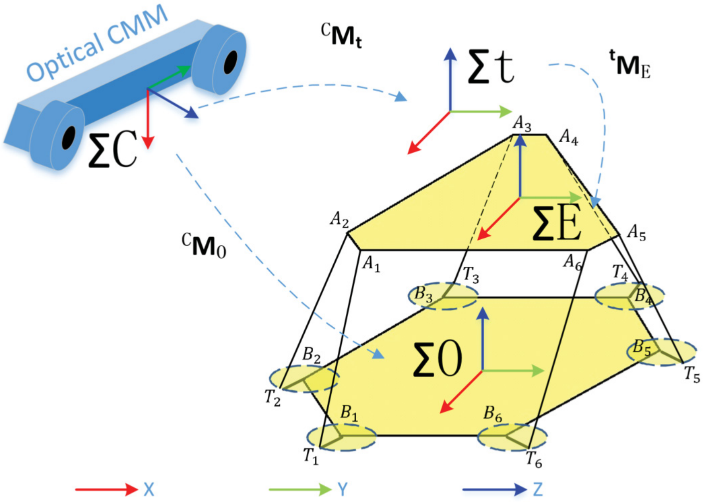


Figure 4.2: The calibration system of 6-RSS parallel robot

where $\min(\cdot)$ represents the minimization function subject to the constraint ${}^C\hat{\mathbf{P}}_{i1}^T G\ {}^C\hat{\mathbf{P}}_{i2} = 0$, $d(*,*)$ denotes Euclidean distance, ${}^C\hat{\mathbf{P}}_{i1}$ and ${}^C\hat{\mathbf{P}}_{i2}$ are the estimated points of ${}^C\mathbf{P}_{i1}$ and ${}^C\mathbf{P}_{i2}$ respectively. As a result, ${}^C\hat{\mathbf{P}}_{i1}$ and ${}^C\hat{\mathbf{P}}_{i2}$ can be derived. Then, by Eq. 4.4, the coordinate of i^{th} feature point in sensor frame, ${}^C\mathbf{P}_i$ can be obtained.

After the position information of all the feature points on the end-effector is prepared, the pose estimation of the end-effector can be developed. Suppose n feature points on the rigid end-effector are fixed and known from the definition of the tool frame $\mathit{\Sigma} t$, whose homogeneous coordinates are denoted as ${}^t\mathbf{P}_i = ({}^t x_{pi}, {}^t y_{pi}, {}^t z_{pi}, 1)$. It is assumed the current pose of $\mathit{\Sigma} t$ in the sensor frame $\mathit{\Sigma} C$ is denoted as ${}^t\boldsymbol{\chi}_C$. Correspondingly, the transformation equation of i^{th} feature point can be written as:

$$ {}^C\mathbf{P}_i = {}^C M_t\, {}^t\mathbf{P}_i, \tag{4.7} $$

where ${}^C M_t$ is the homogeneous transformation matrix from $\mathit{\Sigma} t$ to $\mathit{\Sigma} C$. In order to derive ${}^C M_t$, at least three non-collinear feature points are required [95]. However, as indicated in [96], at least four coplanar feature points are needed for finding an unique solution while additional non-coplanar feature points can be used to improve the estimation accuracy with measurement noise. By employing Eq. 4.8, the homogeneous transformation matrix ${}^C M_E$ from $\mathit{\Sigma} E$ to $\mathit{\Sigma} C$ can be derived.

$$ {}^C M_E = {}^C M_t\, {}^t M_E, \tag{4.8} $$

where ${}^t M_E$ is the homogeneous transformation matrix from $\mathit{\Sigma} E$ to $\mathit{\Sigma} t$ and is known for the definition of the end-effector frame $\mathit{\Sigma} E$ and $\mathit{\Sigma} t$. Similarly, the homogeneous transformation matrix mapping from $\mathit{\Sigma} O$ to $\mathit{\Sigma} C$, ${}^C M_O$, can be obtained. Then, the homogeneous transformation matrix ${}^O M_E$ can be calculated by Eq.4.9.

$$ {}^O M_E = {}^C M_O^{-1}\, {}^C M_E. \tag{4.9} $$

Accordingly, the pose vector $\boldsymbol{\chi_E}$ can be analytical solved from ${}^O M_E$.

By using the proprietary software VXelements provided by Creaform Inc., the target frame can be defined based on the selected reflectors on the surface of the end-effector. The positional and rotational information of the target frame w.r.t. sensor frame can be acquired, recorded or displayed simultaneously. Therefore, the computation to obtain the pose of the target frame is carried out by VXelements.

4.3 Calibration Algorithm Based on the Optical CMM

In this section, we start with the classical implicit calibration method based on optical CMM for parallel robots. Since optical CMM device is involved, external parameters which describe the relationships between base frame and sensor frame, end-effector frame and target frame should be considered during this method implement. The kinematic parameter $\mathbf{b}_k$ can be seen as the internal parameters. Then we propose a RPBA for parallel robot calibration based on optical devices. It can be seen that less external parameters are needed in RPBA. Then the constraints for candidate configurations selection are determined. Finally the identifiability and observability analysis of the calibration are given for both calibration methods.

4.3.1 Implicit Calibration Method Based on the Optical CMM

The implicit kinematic model which depicts the closed structure of parallel robots is commonly employed in parallel robots kinematic calibration since it can avoid solving inverse kinematics and forward kinematics [49]. The choice of implicit kinematic model can be various. Normally it should be composed of equations that reflect the relationships between joint values, kinematic parameters and the pose of end-effector. For the 6-RSS parallel robot, the constraint equations $\varPhi_I$ of implicit kinematic model for calibration can be derived from Eq. 4.3.

$$\begin{aligned}\varPhi_{Ii}(\mathbf{b}_k)=&\|\mathbf{T}_i(\eta_i\kappa_i^l+\varDelta\theta_i, l_{T_iB_i}+\varDelta l_{T_iB_i}, \mathbf{B}_i+\varDelta\mathbf{B}_i) - \mathbf{A}_i\left({}^{O}M_E^l, \mathbf{A}'_i+\varDelta\mathbf{A}'_i\right)\| \\ &- (l_{A_iT_i}+\varDelta l_{A_iT_i}) = 0 \quad (i=1,...,6),\end{aligned} \tag{4.10}$$

where $\varPhi_{Ii}$ is the i^{th} element of $\varPhi_I$. Then the kinematic calibration can be derived by solving an optimization problem with the measurement of joint values κ_i^l and transformation matrix ${}^{O}M_E^l$ at the l^{th} candidate configuration, and the total number m of candidate configurations should satisfy

$6m > 60$. The cost functions for calibration can be written as following:

$$\min \sum_{l=1}^{m} \sum_{i=1}^{6} \parallel \mathit{\Phi}_{Ii}(\mathbf{b}_k) \parallel^{2} \tag{4.11}$$

However, due to the employment of an external optical device, ${}^{O}M_{E}^{l}$ cannot be achieved directly, since one shall take the pose of the sensor frame ${}^{O}\boldsymbol{\chi}_{C}$ w.r.t. the base frame of the parallel robot and the pose of the target frame ${}^{E}\boldsymbol{\chi}_{t}$ w.r.t. the end-effector frame into account. The homogeneous transformation matrices corresponding to ${}^{O}\boldsymbol{\chi}_{C}$ and ${}^{E}\boldsymbol{\chi}_{t}$ is ${}^{O}M_{C}$ and ${}^{E}M_{t}$. Then the ${}^{O}M_{E}^{l}$ at the l^{th} candidate configuration can be derived from the following equation [43]:

$${}^{O}M_{E}^{l} = {}^{O}M_{C}\, {}^{C}M_{t}^{l}\, {}^{t}M_{E}, \tag{4.12}$$

where ${}^{C}M_{t}^{l}$ represents the measurement of target frame pose with respect to the sensor frame at the l^{th} calibration configuration, and ${}^{t}M_{E} = {}^{E}M_{t}^{-1}$.

The 12 parameters representing ${}^{O}\boldsymbol{\chi}_{C} = \left[{}^{o}x_{c}, {}^{o}y_{c}, {}^{o}z_{c}, {}^{o}\alpha_{c}, {}^{o}\beta_{c}, {}^{b}\gamma_{c}\right]^{T}$ and ${}^{t}\boldsymbol{\chi}_{E} = [{}^{t}x_{E}, {}^{t}y_{E}, {}^{t}z_{E}, {}^{t}\alpha_{E}, {}^{t}\beta_{E}, {}^{t}\gamma_{E}]^{T}$ can be viewed as the external parameters for the implicit calibration of parallel robots based on the optical CMM sensor, and can be written as a column vector

$$\mathbf{b}_{Ie} = \left[{}^{o}x_{c}, {}^{o}y_{c}, {}^{o}z_{c}, {}^{o}\beta_{c}, {}^{o}\gamma_{c}, {}^{o}\alpha_{c}, {}^{t}x_{e}, {}^{t}y_{e}, {}^{t}z_{e}, {}^{t}\beta_{e}, {}^{t}\gamma_{e}, {}^{t}\alpha_{e}\right]^{T}.$$

Then substituting Eq. 4.12 into Eq. 4.10, the cost function becomes:

$$\min \sum_{l=1}^{m} \sum_{i=1}^{6} \parallel \mathit{\Phi}_{Ii}(\mathbf{b}_k,\ \mathbf{b}_{Ie}) \parallel^{2} \tag{4.13}$$

Then the updating formula for $\mathbf{b}_I = [\mathbf{b}_k; \mathbf{b}_{Ie}]^T$ is given as follows:

$$\mathbf{b}_{I}^{t+1} = (J_I^T J_I)^{-1} J_I^T \mathit{\Phi}_I^t + \mathbf{b}_I^t, \tag{4.14}$$

where $\mathbf{b}_I^t$, $\mathit{\Phi}_I^t$ is the value of $\mathbf{b}_I$, $\mathit{\Phi}_I$ in the t^{th} iteration respectively and J_I is the Jacobian matrix of

$\varPhi_I$ about $\mathbf{b}_I$ as below:

$$J_I = \begin{bmatrix} \frac{\partial \varPhi_I^1}{b_{I1}} & \frac{\partial \varPhi_I^1}{b_{I2}} & \cdots & \frac{\partial \varPhi_I^1}{b_{I72}} \\ \frac{\partial \varPhi_I^2}{\partial b_{I1}} & \frac{\partial \varPhi_I^2}{\partial b_{I2}} & \cdots & \frac{\partial \varPhi_I^2}{\partial b_{I72}} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial \varPhi_I^m}{\partial b_{I1}} & \frac{\partial \varPhi_I^m}{\partial b_{I2}} & \cdots & \frac{\partial \varPhi_I^m}{\partial b_{I72}} \end{bmatrix}. \tag{4.15}$$

In some researches,[97, 98] the vector $\mathbf{b}_{Ie}$ is derived from hand-eye calibration and calibrated independently assuming the calibration of kinematic parameters is well carried out. But in our case, the $\mathbf{b}_{Ie}$ appears in every constraint equation, the values of $\mathbf{b}_k$ and $\mathbf{b}_{Ie}$ should be calculated simultaneously.

To achieve the better results in solving the nonlinear optimization problem, accurate initial guesses of $\mathbf{b}_k$ and $\mathbf{b}_{Ie}$ are needed. For $\mathbf{b}_k$, the nominal values provided by manufacturer of parallel robots can be used as the initial guess. However the initial guess of $\mathbf{b_e}$ should be measured manually. Taking the advantages of optical CMM sensor, the ${}^t\boldsymbol{\chi}_E$ can be observed and computed easily. However due to the lack of a well-defined and mechanically accessible base frame for the parallel robots, the process of deriving the transformation matrix from the optical sensor frame to the base frame can be expensive and time consuming. If the ${}^C\boldsymbol{\chi}_O$ can only be roughly measured, this may lead to unhealthy results of $\mathbf{b}_k$ from the nonlinear optimization.

4.3.2 RPBA Based on the Optical CMM

Here we propose RPBA in which the pose variation of parallel robot end-effector can be used to eliminate the influence of external parameters. The transformation matrix M^l expressing the relative pose between the l^{th} end-effector configuration ${}^C\boldsymbol{\chi}_t^l$ and an arbitrarily initial pose ${}^C\boldsymbol{\chi}_t^0$ obtained from optical sensor can be derived as follows:

$$M^l = {}^C M_t^{0\,-1}\ {}^C M_t^l. \tag{4.16}$$

The inverse kinematics of the 6-RSS parallel robot can be represented by $f : SE(3) \rightarrow W, \boldsymbol{\theta} = f({}^O M_E)$, W is the actuator parameter space. Then the transformation matrix of initial pose ${}^O M_E^0$

and the l^{th} configuration ${}^{O}M_E^l$ is determined as ${}^{O}M_E^0 = f^{-1}(\boldsymbol{\theta}_0)$ and ${}^{O}M_E^l = f^{-1}(\boldsymbol{\theta}_l)$ respectively, if the forward kinematics f^{-1} exists in the pose ${}^{O}\boldsymbol{\chi}_E^0$ and ${}^{O}\boldsymbol{\chi}_E^l$. The transformation matrix M^l of the relative pose can also be obtained by

$$\begin{aligned} M^l =& ({}^{C}M_O\ {}^{O}M_E^0\ {}^{E}M_t)^{-1}\ {}^{C}M_O\ {}^{O}M_E^l\ {}^{E}M_t \\ =& {}^{E}M_t^{-1}\ {}^{O}M_E^{0\ -1}\ {}^{O}M_E^l\ {}^{E}M_t. \end{aligned} \tag{4.17}$$

The bijective mapping from homogeneous transformation matrix to the pose can be defined as: $\boldsymbol{\chi} = L(M)$, combining the Eq. 4.16 and 4.17, the constraint function $\varPhi_R$ for calibration can be derived as the following equations:

$$\begin{aligned} &\varPhi_R(\mathbf{b}_k, \mathbf{b}_{Re}) = L({}^{C}M_t^{0\ -1}\ {}^{C}M_t^l) - L({}^{E}M_t^{-1}\ {}^{O}M_E^{0\ -1}\ {}^{O}M_E^l\ {}^{E}M_t) \\ &= L({}^{C}M_t^{0\ -1}\ {}^{C}M_t^l) - L({}^{E}M_t^{-1}\ f^{-1}(\boldsymbol{\kappa}^0, \mathbf{b}_k)^{-1}\ f^{-1}(\boldsymbol{\kappa}^l, \mathbf{b}_k)\ {}^{E}M_t) = \mathbf{0}. \end{aligned} \tag{4.18}$$

where, $\varPhi_R(\mathbf{b}_k, \mathbf{b}_{Re})$ is a 6×1 vector; $\boldsymbol{\kappa}^0 = [\kappa_1^0, \kappa_2^0, \ldots, \kappa_6^0]^T$ and $\boldsymbol{\kappa}^l = [\kappa_1^l, \kappa_2^l, \ldots, \kappa_6^l]^T$ can be obtained by the potentiometers; ${}^{C}M_t^0$ and ${}^{C}M_t^l$ are measured by the optical CMM sensor. The external parameters for the relative calibration method $\mathbf{b}_{Re} = \left[{}^{t}x_E, {}^{t}y_E, {}^{t}z_E, {}^{t}\alpha_E, {}^{t}\beta_E, {}^{t}\gamma_E\right]^T$ is just 6 parameters related to ${}^{t}\boldsymbol{\chi}_E$. Based on classical nonlinear least square method and the forward kinematics, an objective function for the kinematic calibration is defined as follows:

$$\min \sum_{l=1}^{m} \parallel \varPhi_R(\mathbf{b}_R) \parallel^2, \tag{4.19}$$

where $\mathbf{b}_R = [\mathbf{b}_k; \mathbf{b}_{Re}]^T$. Then the updating formula for $\mathbf{b}_R$ is given as follows:

$$\mathbf{b}_R^{t+1} = (J_R^T J_R)^{-1} J_R^T\ \varPhi_R^t + \mathbf{b}_R^t, \tag{4.20}$$

where $\mathbf{b}_R^t$ is the value of $\mathbf{b}_R$ in the t^{th} iteration and J_R is the Jacobian matrix of $\varPhi_R$ about $\mathbf{b}_R$ given by:

$$J_R = \begin{bmatrix} \frac{\partial \Phi_R^1}{\partial \mathbf{b}_{R1}} & \frac{\partial \Phi_R^1}{\partial \mathbf{b}_{R2}} & \cdots & \frac{\partial \Phi_R^1}{\partial \mathbf{b}_{R66}} \\ \frac{\partial \Phi_R^2}{\partial \mathbf{b}_{R1}} & \frac{\partial \Phi_R^2}{\partial \mathbf{b}_{R2}} & \cdots & \frac{\partial \Phi_R^2}{\partial \mathbf{b}_{R66}} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial \Phi_R^m}{\partial \mathbf{b}_{R1}} & \frac{\partial \Phi_R^m}{\partial \mathbf{b}_{R2}} & \cdots & \frac{\partial \Phi_R^m}{\partial \mathbf{b}_{R66}} \end{bmatrix}. \tag{4.21}$$

As we can see from Eq. 4.19, the parameters representing ${}^{C}M_{O}$ which lie in $\mathbf{b}_{Ie}$ are omitted in the relative calibration method and hence we only need to calculate the parameters involved in ${}^{E}M_{t}$ which can be easily calibrated using optical sensor in this method.

4.3.3 Constraints Determination for Robot Configuration Selection

The robot configurations used in the kinematic calibration, or named as candidate configurations, will affect the performance of calibration results [38]. In this section, the constraints for the robot configurations selection based on kinematic analysis results on workspace and singularity in [99] are firstly introduced. Two constraints for the candidate configuration selection are listed as follows:

(1) The kinematic mapping should be totally singularity-free;

(2) The candidate configurations should be selected in a workspace where any robot configuration corresponds to unique actuator outputs.

To guarantee the safety of the 6-RSS parallel robot and to avoid the local minimal during the numerical calculation process of the forward kinematics, the first constraint should be satisfied.

The second constraint, known as homomorphism constraint [100], ensures that the forward and the inverse kinematic calculation will converge to the right value during the calibration. In addition, the mapping between Euler angles and the rotation matrix is a homomorphism by limiting the Euler angles in the range of $(-\pi, \pi]$, which ensures the forward and inverse kinematic mapping have the same kind of geometric characteristic.

The singularity analysis is normally based on the Jacobian matrix of the kinematic mapping which is given in Section 3.5. As given in [101], type-I singularity occurs and the 6-RSS parallel robot meets its boundary of available workspace, if $det(J_{a1}) = 0$, where $det(*)$ represents the determinant of the matrix. If all the diagonal elements of J_{a1} hold their signs in a workspace, the parallel robot stays in a type-I singularity-free range. The robot type-II singularity of 6-RSS parallel robot occurs when $det(J_{a2}) = 0$. To avoid the perturbation from the degenerated singularity surface, the Cauchy Index ∇ of J_{a2} can be used to identify the type-II singularity [100]. For any two nearby poses $\boldsymbol{\chi}_1, \boldsymbol{\chi}_2$ in the workspace:

(1) If $\left|\nabla_{\boldsymbol{\chi}_1}^{J_{a2}} - \nabla_{\boldsymbol{\chi}_2}^{J_{a2}}\right| = 0$, line $\boldsymbol{\chi}_1\boldsymbol{\chi}_2$ stays in a type-II singularity-free domain;

(2) If $\left|\nabla_{\boldsymbol{\chi}_1}^{J_{a2}} - \nabla_{\boldsymbol{\chi}_2}^{J_{a2}}\right| = 2$, line $\boldsymbol{\chi}_1\boldsymbol{\chi}_2$ crosses a non-degenerate type-II singularity surface;

(3) If $\left|\nabla_{\boldsymbol{\chi}_1}^{J_{a2}} - \nabla_{\boldsymbol{\chi}_2}^{J_{a2}}\right| > 2$, and $\min\limits_{\boldsymbol{\chi}_1\boldsymbol{\chi}_2} det(J_{a2}) = 0$, line $\boldsymbol{\chi}_1\boldsymbol{\chi}_2$ crosses a degenerated type-II singularity surface.

Singularity-free is not a sufficient condition for the uniqueness of forward kinematic solution in parallel robots [102]. The second constraint requires for a workspace in which $\mathbb{H}_{fwd}$ is a bijective mapping. The conclusion in the previous study [100] is directly used: kinematic mapping $\mathbb{H}_{fwd}$ is a homeomorphism if the following equation holds:

$$\sqrt[\frac{2}{n}]{\sum_{i=1}^{n} \sigma_i^2} \leq C \cdot \left|\prod_{i=1}^{n} \lambda_i\right|, \tag{4.22}$$

where σ_i and λ_i is the i^{th} singular value and the eigenvalue of $J_{ad_2} = J_{a2}^{-1} J_{a1}$ respectively, $C \geq 1$ is a bounded real number.

4.3.4 Identifiability and Observability Index

Since external parameters are involved in both implicit calibration and relative calibration, firstly the identifiability of external parameters and kinematic parameters should be checked to see if there is linear dependency between those parameters. According to Eq. 4.15 and 4.21, if any parameter did not lie in the kernal of the regressor J_I and J_R, the parameter to be calibrated cannot be updated in each iteration, which means the parameter is non-identifiable. Therefore the full rank of J_I and J_R should be guaranteed.

In addition to checking the identifiability of kinematic and external parameters, the observability should also be considered to minimize the estimation errors by selecting an optimal set of calibration candidate configurations. In this research, the observability index proposed in [103], which emphasizes the volume of a hyper-ellipsoid whose directions are represented by the singular values of the Jacobian matrix, denoted by $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_\varsigma$, is chosen as the criteria in this research. The index O_{in} can be expressed as (take J_I case for instance):

$$O_{in} = \frac{\sqrt[\varsigma]{\sigma_1 \sigma_2 \cdots \sigma_\varsigma}}{\sqrt{m}} = \frac{\sqrt[\varsigma]{det(\sqrt{J_I^T J_I})}}{\sqrt{m}} \tag{4.23}$$

where m is the number of calibration candidate configurations, ς is the number of error parameters.

4.4 Simulation Case Study

In this section, simulations are based on the geometry of 6-RSS parallel robot depicted in Figure 3.1 and the setup is shown in Figure 4.2. The proper actuator strokes for the calibration are determined. And the optimal set of candidate configurations is derived for both implicit and relative calibration methods. Then the calibration simulation comparison of two calibration methods is implemented to show the superiority of RPBA.

4.4.1 Actuator Stroke for Calibration Configuration Determination

Compared with the end-effector poses, it is more convenient to describe the robot configuration by the actuator joints' angles. Hence the results in this subsection are all determined in the actuator parameter space. Besides, the kinematic calculation is based on ideal kinematic model. Two kinds of parallel robot configurations should be determined for the kinematic calibration: the initial configuration and the candidate configurations.

The optimal initial actuator angles for the 6-RSS parallel robot are listed in the vector $[-60^\circ, -60^\circ, 180^\circ, 180^\circ, 60^\circ, 60^\circ]$ as shown in the previous work [99]. For the selection of candidate poses, we try to determine the maximum singularity-free actuator stroke in which the kinematic equation $\mathbb{H}_{fwd}$ is a homomorphism mapping. Considering the symmetric structure of the 6-RSS parallel robot, the optimization problem can be simplified into a single-objective planning: finding the maximum of actuator strokes under constraint conditions introduced in last section. The cost function is given by Eq. 4.24.

$$\max_{(-\pi/2,\pi/2)} \sum_{i=1}^{6} \mu_i \left| \theta_{max,i} - \theta_{min,i} \right|, \tag{4.24}$$

where $\theta_{max,i}$ and $\theta_{min,i}$ are the maximum and minimum actuator angles of the i^{th} branch respectively, and μ_i is the positive punishment coefficient. The used constraint conditions and parameters are listed as follows:

(1) Initial angles $[-60°, -60°, 180°, 180°, 60°, 60°]$;

(2) Singularity I-free condition
$sgn(J_{a1}) = [+, -, +, -, +, -]$;

(3) Singularity II-free condition $\nabla^{J_{a2}} = 0$ and $\det(J_{a2}) \neq 0$;

(4) Homomorphism condition Eq. 4.22.

The optimization procedure is shown in Figure 4.3. Assuming the initial angle is at the zero degree axis of the polar coordinate, we try to determine the upper and lower bounds of the actuator stroke.

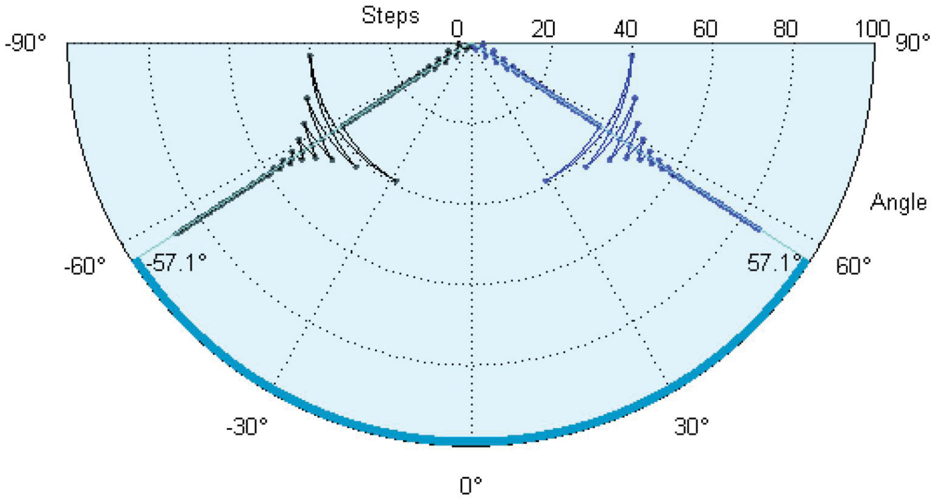


Figure 4.3: The determination of proper actuator stroke

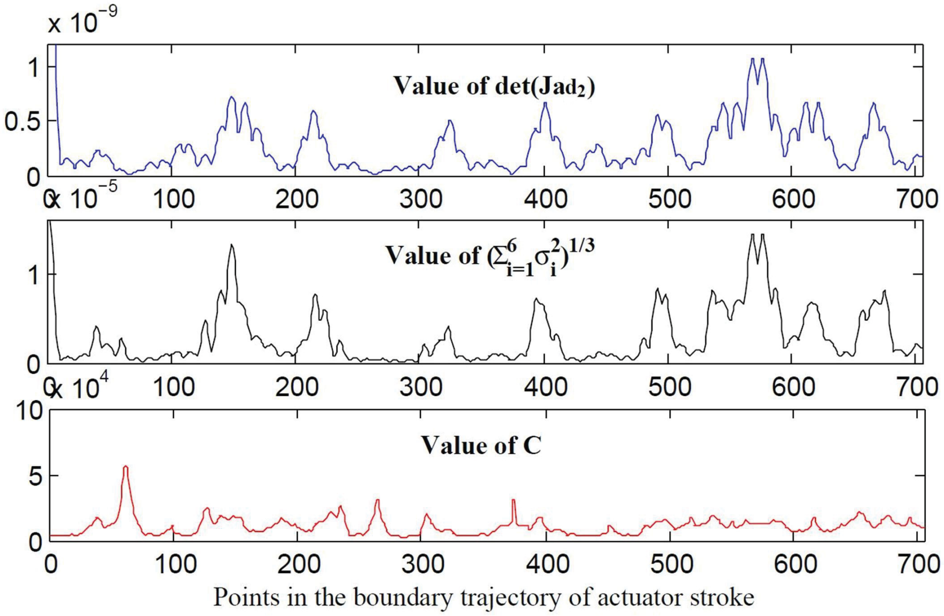


Figure 4.4: Bijective validation through the boundaries

After the 40^{th} step the two bounds start to converge. Finally, the robot configurations should be selected in the actuator stroke $(-57.1^{\circ}, 57.1^{\circ})$.

We validate the homomorphism constraint in the boundaries of the determined actuator stroke to ensure the uniqueness of inverse and forward kinematic solution. The searching trajectory starts from initial angles and is determined by link terminal points of the actuator strokes. Because the trajectory is selected from the boundary of the maximum singularity-free domain, as shown in Fig.4.4, the value of $\det(J_{ad_2})$ is almost zero in every point of the trajectory. From Eq. 4.22, as the value of C is bounded, and the kinematic mapping $\mathbb{H}_{fwd}$ is a bijective (homomorphism) mapping in the determined actuator stroke.

4.4.2 Optimal Set of Calibration Configurations Selection Simulation

In this subsection, the identifiability and the observability of the kinematic parameters for both implicit calibration and relative calibration are checked. According to Section 4.3.4, the identifiability of $\mathbf{b}_I$ and $\mathbf{b}_R$ in both calibration methods is dependent on the Jacobian matrix J_I and J_R

which are calculated numerically (take the relative method case for instant) as

$$J_R=\begin{bmatrix} \frac{\varPhi_R^1(b_{R1}+\varepsilon)-\varPhi_R^1}{\varepsilon} & \frac{\varPhi_R^1(b_{R2}+\varepsilon)-\varPhi_R^1}{\varepsilon} & \dots & \frac{\varPhi_R^1(b_{R66}+\varepsilon)-\varPhi_R^1}{\varepsilon} \\ \frac{\varPhi_R^2(b_{R1}+\varepsilon)-\varPhi_R^2}{\varepsilon} & \frac{\varPhi_R^2(b_{R2}+\varepsilon)-\varPhi_R^2}{\varepsilon} & \dots & \frac{\varPhi_R^2(b_{R66}+\varepsilon)-\varPhi_R^2}{\varepsilon} \\ \dots & \dots & \dots & \dots \\ \frac{\varPhi_R^m(b_{R1}+\varepsilon)-\varPhi_R^m}{\varepsilon} & \frac{\varPhi_R^m(b_{R2}+\varepsilon)-\varPhi_R^m}{\varepsilon} & \dots & \frac{\varPhi_R^m(b_{R66}+\varepsilon)-\varPhi_R^m}{\varepsilon}, \end{bmatrix} \tag{4.25}$$

where $\varepsilon = 10^{-6}$ is the small variation added to each kinematic parameter. For both methods the J_I and J_R are of full rank which means all the internal and external kinematic parameters are identifiable.

The searching method of the optimal set of calibration configurations proposed by Nubiola et al. is used in this chapter to maximize the index O_{in} given in Eq. 4.23 [104]. For both calibration methods, the numerical algorithm starts with a candidate set of 13 random configurations selected from the determined proper actuator stroke in the previous subsection. At each iteration, one configuration chosen from 5000 random configurations in the proper actuator stroke is added to the candidate group. If the index increases, this configuration is kept in the candidate group and any other configuration which decreases the index of the set of remaining configurations is removed. Otherwise this configuration is discarded and the process goes to next iteration.

For implicit calibration method, the algorithm stops after 618 iterations with the index reaching 0.0031 which cannot be further improved. And for relative calibration method, the maximum index goes to 0.0322 with 572 iterations. The optimal sets for implicit and relative method contain 34 and 22 configurations respectively.

4.4.3 Calibration Simulation

In this subsection, the calibration simulation is carried out on the case study of the 6-RSS parallel robot based on the pose measurement from the optical CMM system. The detection noise in the simulation is determined based on experimental analysis.

As introduced in Section 4.2.2, the reflectors attached on the end-effector can be used as the feature points for the pose estimation. And the noise of the poses and joint angle measurements should

Table 4.1: The standard deviations of the noise distribution

	Standard deviation
Joint angle measurement $\boldsymbol{\kappa}$	0.2233 bit
Pose measurement (x, y, z)	0.0096 mm
Pose measurement (β, γ, α)	0.0261 deg

Table 4.2: The standard deviations of the parameter errors distribution

	Standard deviation
$[b_{I1}, b_{I2}, \cdots, b_{I48}]$	1 mm
$[b_{I49}, b_{I50}, \cdots, b_{I54}]$	0.00005 rad/bit
$[b_{I55}, b_{I56}, \cdots, b_{I60}]$	0.05 rad
$[b_{I61}, b_{I62}, b_{I63}], [b_{I67}, b_{I68}, b_{I69}]$	0.5 mm
$[b_{I64}, b_{I65}, b_{I66}], [b_{I70}, b_{I71}, b_{I72}]$	0.017 rad

be considered in the simulations. With a 10000 times detection for the robot initial configuration, the error distribution of the feature points is obtained. Based on the results of noises analysis, the measurement noise satisfies a two-dimensional normal distribution with standard deviations shown in the Table 4.1.

In the calibration simulation, a group of parameter errors $\mathbf{b}_I^n$ is generated randomly complying with normal distribution. Note that $\mathbf{b}_R^n$ is a subset of $\mathbf{b}_I^n$. The standard deviations ρ given in Table 4.2 are roughly chosen such that $\pm 3\rho$ error would lie in $\pm 20\%$ of the nominal parameters.

In the simulation, the procedure of implementing the implicit calibration method is given as follows:

(1) Load the initial parameters $\mathbf{b}_I^0$;

(2) Generate a group of parameter errors with normal distributions of certain standard deviations as mentioned. By adding the errors to the initial parameters, the real kinematic parameters $\mathbf{b}_I^n$ can be obtained;

(3) Load the optimal set of 34 configurations derived from previous subsection for the implicit calibration method. Through Eq. 4.2, the optimal set expressing in the actuator domain is converted into joints' readings. Then the joint measurement $\boldsymbol{\kappa}^l (l = 1, ..., 34)$ is derived by adding the detection noise to the joints' readings;

Table 4.3: The error norms between nominal and calibrated values-normal for implicit calibration method

	Before calibration	After calibration
$\|\mathbf{b}_I^c - \mathbf{b}_I^n\|_2$	3.273	0.8197
$\sum_{l=1}^{34} \|\varPhi_I^c - \varPhi_I^n\|_2$	0.00109929	2.94235e-05

(4) The measurements of the poses ${}^{C}\boldsymbol{\chi}_t^l$ are derived from the forward kinematics and Eq. 4.12, employing the real kinematic parameters $\mathbf{b}_I^n$. Also the measurement noise should be added to the pose values;

(5) The updating formula Eq. 4.14 is used to determine the optimal error parameters $\mathbf{b}_I$ by minimizing Eq. 4.13.

After 11 iterations, the calibrated error parameters $\mathbf{b}_I^c$ are found. The results with the norm of 72 error-parameters and $\varPhi_I$ are shown in Table 4.3.

For the RPBA simulation, the procedure is similar to that of the implicit calibration as follows:

(1) Load the initial parameters $\mathbf{b}_R^0$;

(2) Extract the subset $\mathbf{b}_R^n$ from $\mathbf{b}_I^n$;

(3) Load the optimal set of 22 configurations derived and obtain the joint measurement $\boldsymbol{\kappa}^l (l = 1, ..., 22)$ with the measurement noise;

(4) The measurements of the relative poses M^l are derived from the forward kinematics and Eq. 4.17, employing the real kinematic parameters $\mathbf{b}_R^n$. Also the detection noise should be added to the pose values;

(5) The updating formula Eq. 4.19 is used to determine the optimal error parameters $\mathbf{b}_R$ by minimizing Eq. 4.20.

After 4 iterations, the calibrated error parameters $\mathbf{b}_R^c$ are found. The results with the norm of 66 error-parameters and $\varPhi_R$ are shown in Table 4.4.

The effectiveness of the two calibration methods are compared regarding to the positioning accuracy improvements with respect to the initial pose. The simulation procedure is given as follows:

Table 4.4: The error norms between nominal and calibrated values-normal for RPBA

	Before calibration	After calibration
$\|\mathbf{b}_R^c - \mathbf{b}_R^n\|_2$	3.1630	0.6616
$\sum_{l=1}^{22} \|\Phi_R^c - \Phi_R^n\|_2$	0.00542206	2.38881e-05

(1) Without loss of generality, 100 random test relative configurations $M_n^l (l = 1, 2 \cdots 100)$ w.r.t. the initial configuration are generated;

(2) Load the initial parameters $\mathbf{b}_I^u$, the calibrated parameters $\mathbf{b}_I^c$ and $\mathbf{b}_R^c$ derived from the two calibration methods;

(3) The test configurations expressed in actuator domain $\boldsymbol{\theta}_u^l$, $\boldsymbol{\theta}_I^l$ and $\boldsymbol{\theta}_R^l$ are derived through the inverse kinematics and Eq. 4.17, employing $\mathbf{b}_I^u$, $\mathbf{b}_I^c$ and $\mathbf{b}_R^c$ respectively.

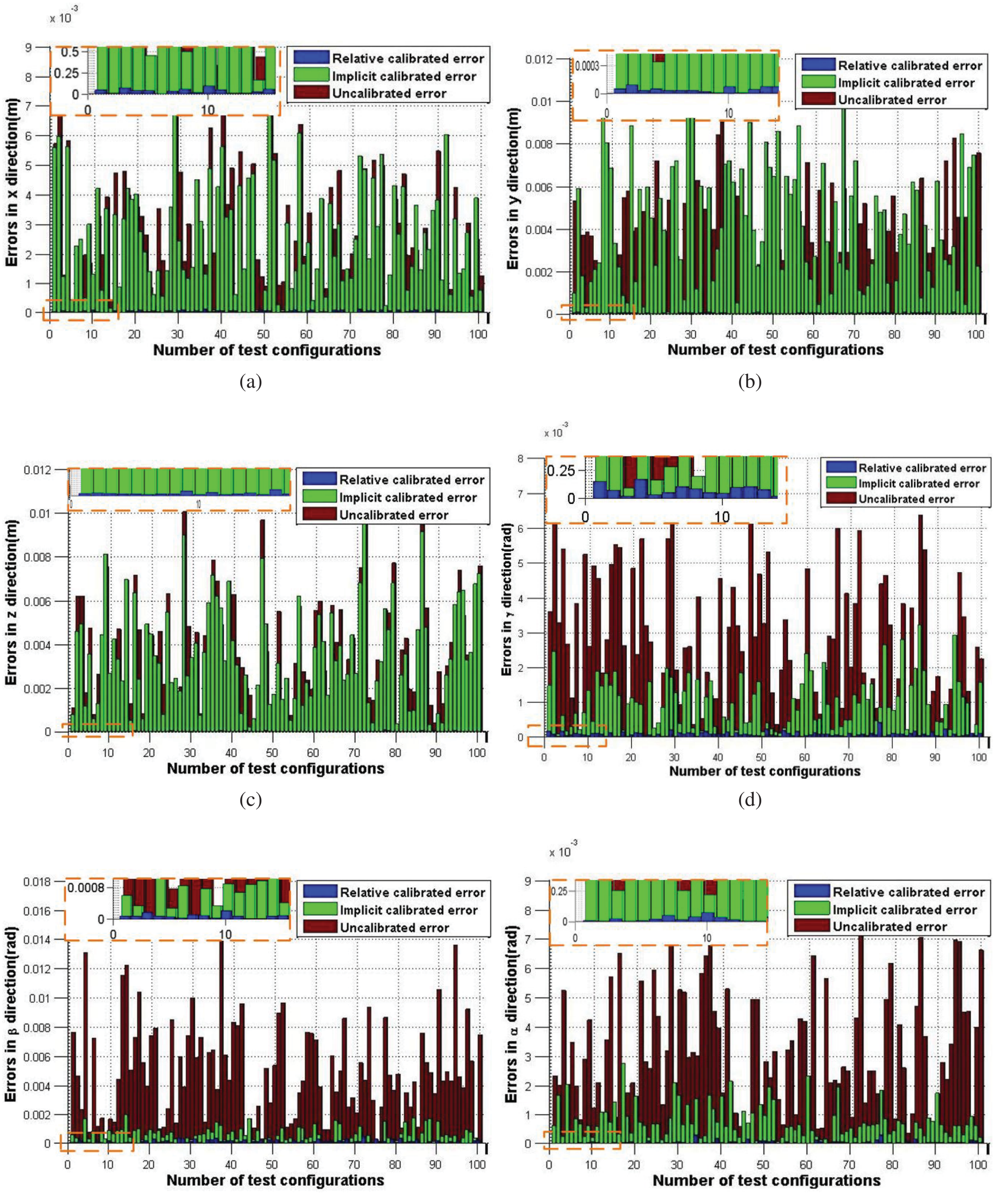
(4) Input the deriving $\boldsymbol{\theta}_u^l$, $\boldsymbol{\theta}_I^l$ and $\boldsymbol{\theta}_R^l$ into the real kinematic model of the robot separately, which employs the real parameters $\mathbf{b}_I^n$. Then the relative configurations M_u^l, M_I^l and M_R^l can be obtained respectively.

Converting M_n^l, M_u^l, M_I^l and M_R^l into $SE(3)$ space, we obtain $\boldsymbol{\chi}_n^l$ (the nominal relative poses), $\boldsymbol{\chi}_u^l$ (the relative poses deriving from uncalibrated parameters), $\boldsymbol{\chi}_I^l$ (the relative poses deriving from implicit calibration result) and $\boldsymbol{\chi}_R^l$ (the relative poses deriving from RPBA result) respectively. Then the pose errors $\boldsymbol{\chi}_{u,e}^l$, $\boldsymbol{\chi}_{I,e}^l$ and $\boldsymbol{\chi}_{R,e}^l$ can be derived from $\boldsymbol{\chi}_u^l - \boldsymbol{\chi}_n^l$, $\boldsymbol{\chi}_I^l - \boldsymbol{\chi}_n^l$ and $\boldsymbol{\chi}_R^l - \boldsymbol{\chi}_n^l$ respectively.

Table 4.5: The RMS of the relative pose errors in simulation.

	$\boldsymbol{\chi}_{u,e}^l$	$\boldsymbol{\chi}_{I,e}^l$	$\boldsymbol{\chi}_{R,e}^l$
x	3.4 mm	3.1 mm	0.053 mm
y	3.8 mm	4.9 mm	0.039 mm
z	4.5 mm	4.2 mm	0.018 mm
γ	0.0033 rad	0.0012 rad	9.7 e-5rad
β	0.0060 rad	7.9 e-4 rad	1.6 e-4 rad
α	0.0037 rad	9.6 e-4 rad	6.7 e-5rad

The results are shown in Figure 4.5 and Table 4.5. The results show that both calibration methods can improve the positioning accuracy of the parallel robot. In addition, the RPBA can improve



(a) (b)

(c) (d)

(e) (f)

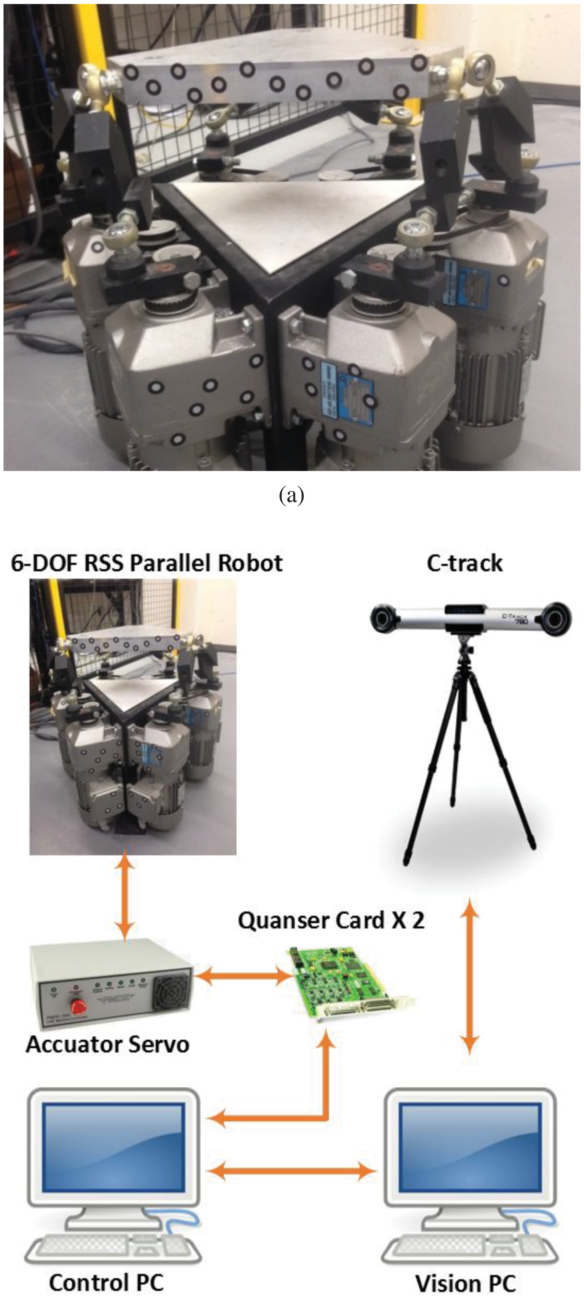
Figure 4.5: The simulation results of relative pose errors derived from implicit calibration, relative calibration and un-calibration: (a) Along X Direction; (b) Along Y Direction; (c) Along Z Direction; (d) Around α Axis; (e) Around β Axis; (f) Around γ Axis

the positioning accuracy of the parallel robot in both translational and angular portions compared with the implicit calibration method. The implicit calibration results lead to a considerable accuracy improvement on the angular portion, but only slightly improvement is gained in terms of the translational accuracy. The RPBA can improve the positioning accuracy significantly in all directions of the relative pose at the given configurations. Therefore, a conclusion can be drawn that considering the relative pose accuracy, the parallel robot can fulfill more accurate positioning tasks based on the proposed RPBA than those based on the implicit method.

4.5 Experimental Validation

The experiment setup is shown in Figure 4.6b. The 6-RSS parallel robot with 6 built-in joint controllers is provided by Servo $\&$ Simulation Inc. (Sanford, Fl, USA). as shown in Figure 4.6a. The built-in controllers communicate with the robot control computer through two Quanser MultiQ-PCI (Sensoray Model 626) data acquisition cards provided by Quanser Inc. (Markham, ON, Canada). Quanser's QUARC^{TM} software is running on the robot control computer with Windows 7.0 32-bit operating system and Intel Core Processor i7-3770 3.4 GHz. QUARC^{TM} software is capable of generating real-time application though Simulink based controllers and implementing the application in real time on the Windows target. The C-track 780 provided by Creaform Inc. (Levis, QC, Canada) is used to obtain the image data of the reflectors attached on the robot. The reflectors provided by Creaform Inc. are magnetic stickers which are easily fixed on the robots and are used as the feature points. In another Windows 7.0 64-bit computer with Intel Xeon Processor E5-1650 v3 3.5 GHz and NVIDIA Quadro K2200 (Santa Clara, CA, USA) professional graphics board, Vxelements software is used to process the image data and transmit the pose of the end-effector to the robot control computer.

As shown in Figure 4.7, the reflectors are stuck on the surface of the moving platform. At least three non-collinear points on each plane of Plane A, Plane B and Plane C are employed to build up the equations of planes based on Cramer's rule. Then the intersection lines and points of three planes can be used to define the x direction of ΣE, and z direction is aligned with the norm of Plane A. The origin point of ΣE is derived from the intersection point of l_1 and l_2. Then, the obtained



(a)

(b)

Figure 4.6: (a) The 6-RSS parallel robot, (b) Architecture of the Experiment System

ΣE in the optical CMM sensor frame is directly used as the target frame. The base frame of the parallel robot is defined by following the similar procedure.

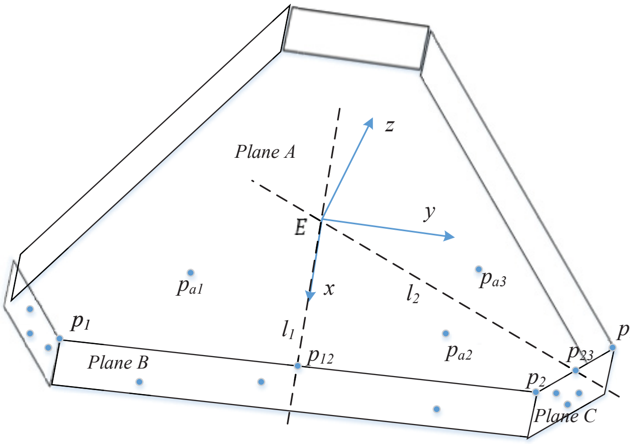
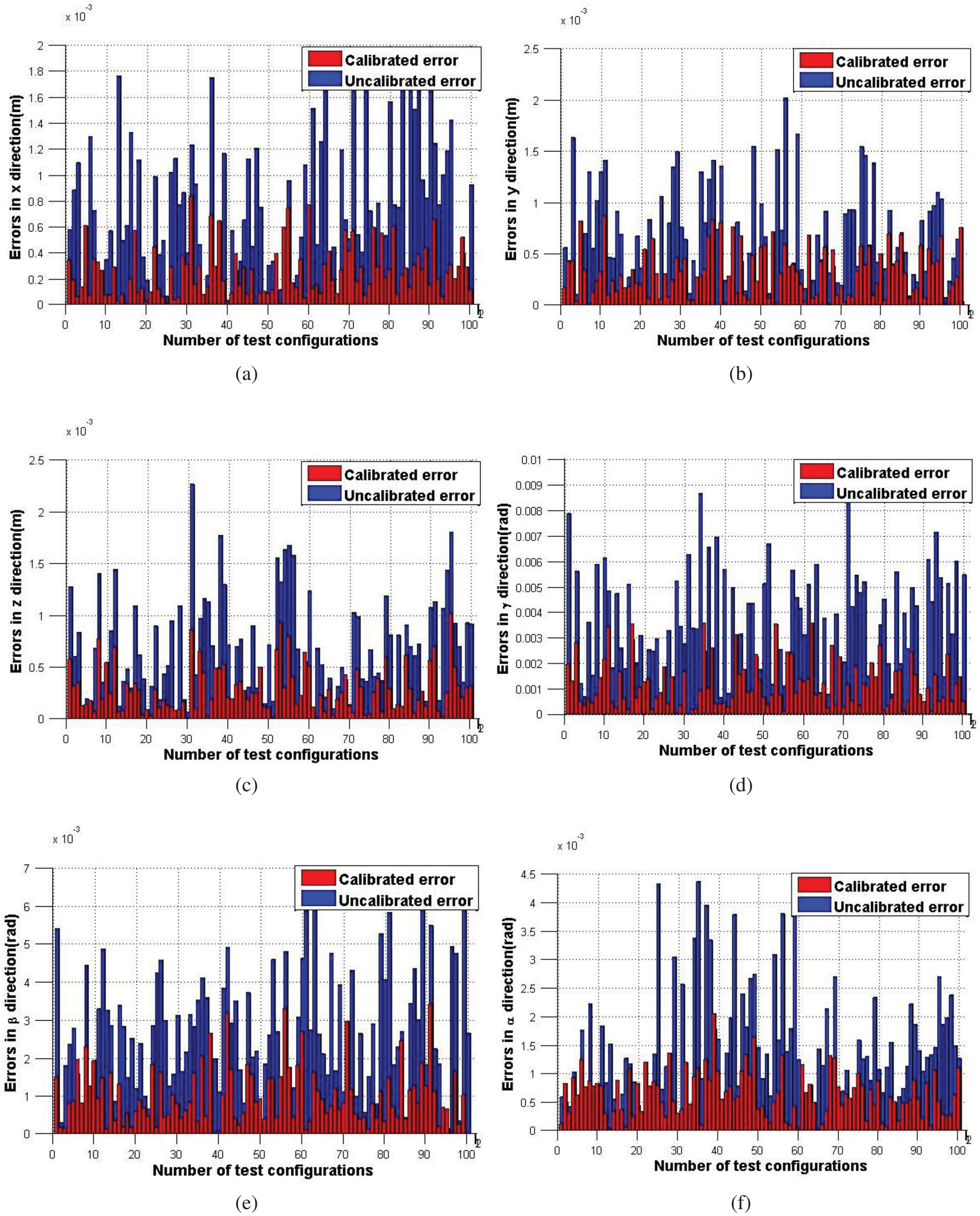


Figure 4.7: Measurement of ΣE.

For the relative calibration method, the optimal set of 22 configurations derived from previous section are selected as candidate configurations. And the actuator values are set as $[-60°, -60°, 180°, 180°, 60°, 60°]$ for the initial configuration. At each configuration the robot stops for 2 seconds and the sampling frequency of the potentiometer and C-track is 500 Hz and 29 Hz respectively. The mean values are used for each configuration. Then the deriving ${}^{C}\boldsymbol{\chi}_t^l$ and $\boldsymbol{\kappa}^l(l = 0, 1 \cdots 22)$ are utilized in the objective function Eq. 4.19. Eq. 4.20 is employed as the updating formula. After 5 steps iterations, the calibrated kinematic parameters are obtained. The objective function starts from 0.00155347 and converges to 7.07731×10^{-5}.

To validate the calibration results, 100 random relative configurations $\chi_n^l(l = 1, 2, \cdots, 100)$ w.r.t. its initial configuration inside the workspace of the parallel robot is chosen. And two sets of configurations are generated by the robot using un-calibrated and calibrated kinematic model respectively. Also the mean values of the sensor reading are utilized in the experimental test. By using Eq. 4.17, the relative pose χ_c^l based on the calibrated kinematic model and χ_u^l based on the un-calibrated model can be derived. The pose errors after calibration $\chi_{c,e}^l$ should be $\chi_c^l - \chi_n^l$ and the pose errors before calibration $\chi_{u,e}^l$ can be derived from $\chi_u^l - \chi_n^l$. Those errors show how accurately the parallel robot can position to a given configuration w.r.t. the initial configuration. The results are given in Fig. 4.8 and Table 4.6.



(a) (b)

(c) (d)

(e) (f)

Figure 4.8: The experiment result of relative pose errors derived from calibrated model and uncalibrated model: (a) Along X Direction; (b) Along Y Direction; (c) Along Z Direction; (d) Around α Axis; (e) Around β Axis; (f) Around γ Axis

Table 4.6: The RMS of the relative pose errors in experiment.

	Before calibration	After calibration
x	0.876 mm	0.319 mm
y	0.796 mm	0.393 mm
z	0.820 mm	0.360 mm
γ	0.039 rad	0.0015 rad
β	0.031 rad	0.0013rad
α	0.017 rad	7.464 e-04 rad

As it can be seen from the test results, the relative calibration can effectively improve the positioning accuracy of the parallel robot on both translational and angular directions. And this method is free from the tedious measurement of the relationship between the base frame and the sensor frame. Thus, it is concluded that the RPBA is an implementable and effective method for the parallel robot calibration.

4.6 Summary

In this Chapter, a relative pose-based kinematic calibration method is proposed for a 6-RSS parallel robot by using the optical CMM system. The developed calibration algorithm can improve the positioning accuracy through more accurate kinematic parameters. In this algorithm, the base frame pose w.r.t. the sensor frame is not needed, which leads to an effective relative calibration method for the parallel robot. Since the forward kinematic model is used in the relative pose based calibration, the optimal actuator strokes of the parallel robot are derived to ensure the homeomorphism mapping of the forward kinematic model. The simulation results show the relative pose based calibration algorithm successfully improves the relative accuracy of the parallel robot. The comparison with the implicit calibration demonstrates that the RPBA can deliver a more satisfactory accuracy. The experimental tests on an arbitrary trajectory with 100 configurations further show the proposed RPBA has improved the positional accuracy in the workspace effectively. The developed calibration algorithm can be applied to other types of parallel and serial robots. And the calibrated kinematic parameters would be used in a 6-RSS parallel robot visual servoing design in the subsequent chapters.

Chapter 5

Visual Close-loop Output-error Identification Method for the 6-RSS Parallel Robot

5.1 Introduction

Parallel robots are often considered as high-precision robots since there is no accumulation of joint errors. However, this is not always guaranteed due to the kinematic parameter errors and difficulties in the dynamic analysis. The kinematic parameter errors have been determined by the kinematic calibration method introduced in Chapter 4. The dynamic model of the parallel robot is normally more complex than that of serial robots, due to the fact that more moving components are needed to be considered, and also there exits highly coupling relations between those moving components. However, the dynamic model is the essential foundation of the dynamic model-based controller design to improve the tracking accuracy.

To obtain an accurate dynamic model, the dynamic identification needs to be carried out. The identification methods of 6-DOF parallel robot in [21, 62] are borrowed from those of serial robots. The identification is formulated as the linear optimization problem in which the dynamic model is built as linear form of the parameters to be identified including dynamics and friction parameters. The measurement of joint angle position and actuator torque data are needed for identification purpose. In [21], the time-consuming numerical method is utilized to estimate the pose of the end-effector frame from the joint measurement, since forward kinematics of 6-DOF parallel robots cannot be solved analytically [6]. Thanks to the visual sensor, the pose of the end-effector frame can be directly measured via the optical CMM sensor.

Another drawback of the current method is that the actuator torque or current measurement is a prerequisite for the dynamic identification. However, the torque related data is normally inaccessible to the users of industrial robots. The output-error identification method can be utilized to solve this problem. The dynamic parameters can be obtained bysolving the optimization problem whose objective is to minimize the output error of the real plant and the mathematical model subject to the same desired path. And it has been applied in the identification of serial robots [57].

A closed-loop output-error identification method based on a CMM sensor is proposed for parallel robots in this chapter. The end-effector pose is measured by the optical CMM and served as the output of the real plant. The same outer loop visual servoing controller and reference trajectory are employed in both actual robot and simulation model for model identification. The forward kinematics of parallel robots, which is usually solved by using time-consuming numerical algorithm, can be avoided. The exact knowledge of the built-in controller and the joint torque are not needed. The dynamic model parameters are identified by using nonlinear optimization technique. The experimental tests validate the identification results.

This chapter is organized as follows. Section 5.2 describes the linear form of the dynamic model of the 6-RSS parallel robot. The closed-loop output-error identification method is proposed in 5.3. The selection of the exciting trajectory and the procedure of the identification are presented in Section 5.4 and 5.5 respectively. The dynamic model validations based on simulation and the experiment results of the identification are given in Section 5.6. Finally, a summary is drawn in Section 5.7.

5.2 Linear Form of the Dynamic Model

Since the kinematic parameters of parallel robots can be obtained through the kinematic calibration method as introduced in Chapter 4, only inertia and friction parameters are considered for the model identification of parallel robots. Considering the heavy computation load of solving inverse dynamic model for the dynamic identification and visual servoing purpose, a reduction of the dynamic parameters with the trade-off between the computation load and accuracy should be implemented. The geometry feature of the parallel robot can be considered for simplification. For the 6-RSS parallel robot, it is assumed that the wrenches, links and end-effector are symmetric. Furthermore, the center of mass is assumed to be located in the geometric center. Therefore, the dynamic parameters for each body of the wrenches, links and end-effector can be reduced to $\boldsymbol{\xi_*} = [m_*, I_{x_*}, I_{y_*}, I_{z_*}]^T$, where $*$ represents arbitrary moving body of the parallel robot. Then Eq. 3.41 can be rewritten as the linear form w.r.t. the dynamic parameters and the friction coefficients:

$$\boldsymbol{\tau_g} = \mathit{\Gamma}(\boldsymbol{\chi_E}, \boldsymbol{v_E}, \dot{\boldsymbol{v}}_{\boldsymbol{E}})\boldsymbol{\mathit{\Xi}}, \tag{5.1}$$

where $\boldsymbol{\mathit{\Xi}} = [\boldsymbol{\xi_p}^T, \boldsymbol{\xi_{w1}}^T, \boldsymbol{\xi_{w2}}^T, \ldots, \boldsymbol{\xi_{w6}}^T, \boldsymbol{\xi_{l1}}^T, \boldsymbol{\xi_{l2}}^T, \ldots, \boldsymbol{\xi_{l6}}^T, f_{c_1}, f_{c_2}, \ldots, f_{c_6}, f_{v_1}, f_{v_2}, \ldots, f_{v_6}]^T$ is a $\mathbb{R}^{64\times1}$ vector of dynamic parameters and the friction coefficients, and $\mathit{\Gamma}(\boldsymbol{\chi_E}, \boldsymbol{v_E}, \dot{\boldsymbol{v}}_{\boldsymbol{E}})$ is the regressor matrix, which consists of the kinematic parameters, state variables and their derivatives. $\mathit{\Gamma}(\boldsymbol{\chi_E}, \boldsymbol{v_E}, \dot{\boldsymbol{v}}_{\boldsymbol{E}})$ can be derived using the Symbolic Math Toolbox of Matlab. Given an exciting trajectory as a reference input to the robot, which will be introduced in Section 5.4, Eq. 5.2 can be obtained by reorganizing Eq. 5.1.

$$\begin{bmatrix} \boldsymbol{\tau_{g_1}} \\ \boldsymbol{\tau_{g_2}} \\ \vdots \\ \boldsymbol{\tau_{g_n}} \end{bmatrix} = \begin{bmatrix} \mathit{\Gamma}(\boldsymbol{\chi_{E_1}}, \boldsymbol{v_{E_1}}, \dot{\boldsymbol{v}}_{\boldsymbol{E_1}}) \\ \mathit{\Gamma}(\boldsymbol{\chi_{E_2}}, \boldsymbol{v_{E_2}}, \dot{\boldsymbol{v}}_{\boldsymbol{E_2}}) \\ \vdots \\ \mathit{\Gamma}(\boldsymbol{\chi_{E_n}}, \boldsymbol{v_{E_n}}, \dot{\boldsymbol{v}}_{\boldsymbol{E_n}}) \end{bmatrix} \boldsymbol{\mathit{\Xi}} = H\boldsymbol{\mathit{\Xi}}, \tag{5.2}$$

where n is the number of the sampled poses from the given trajectory. By feeding various testing trajectories to the robot, the regression matrix H is of full rank, which means all elements of $\boldsymbol{\mathit{\Xi}}$ can be identified.

5.3 Closed-loop Output-error Identification Method

The basic idea of output error identification is to use nonlinear optimization technique to minimize the squared error between the outputs of the real plant and the simulated model. Since the motor torque is inaccessible, the pose of the end-effector, $\boldsymbol{\chi_E}$, measured by optical CMM is used as the output of the system. Hence, the closed-loop output-error identification method is adopted. In conventional closed-loop output-error identification method, the controllers should be exactly known and applied to both the real plant and the simulated model. However for an industrial robot, the built-in controllers are usually unknown and need to be identified.

The closed-loop output-error identification approach for vision-based robotic system, as depicted in Figure 5.1, is proposed in this chapter. For the built-in controller of the 6-RSS parallel robot, a PID controller is used to control the joint angle of each revolute joint. However, the three gains of PID controller are unknown and needed to be identified. During the process of identification, the gains and dynamic parameters are updated in each iteration of nonlinear optimization. An outer loop visual servoing controller is added to stabilize both real robot and simulated model. The visual servoing controller and the built-in controller form a cascade PID controller. As stated in [105, 106], the cascade controller yields better dynamic performance in terms of stability and working frequency compared with single loop controller. With a well-tuned outer loop visual servoing controller, both the real and simulation systems can have a better performance and stability.

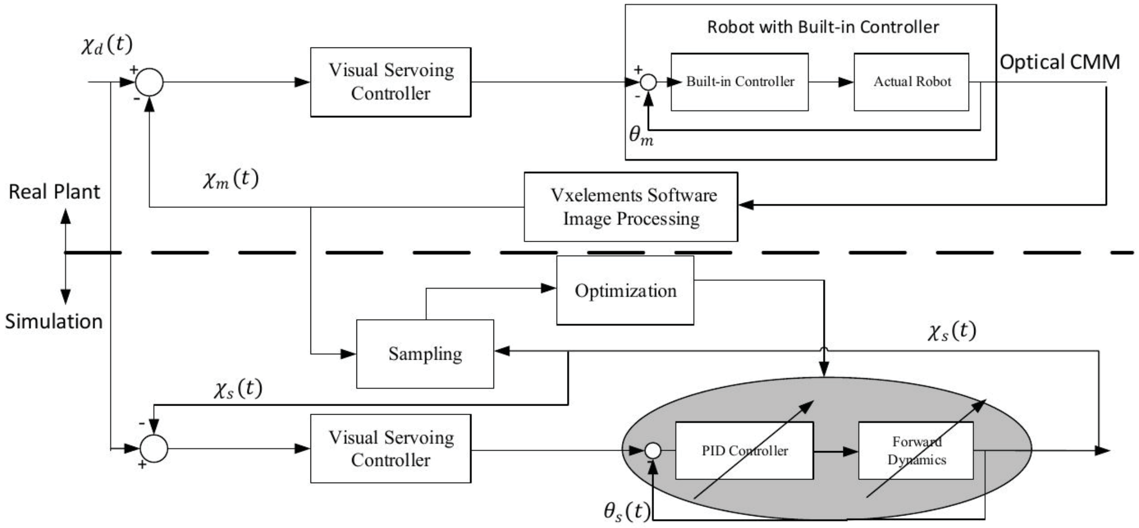


Figure 5.1: The block diagram of closed-loop control system for model identification of 6-RSS parallel robot.

In Figure 5.1, for both the real plant and simulated model, the same exciting trajectory of the end-effector pose w.r.t. the base frame is given as the input $\boldsymbol{\chi_d(t)}$. Then, the outer loop visual servoing controller is designed as Eq. 5.4. The inverse kinematic Jacobian J_θ is used to transform the velocity in the workspace to that in the joint space. By combining Eq. 3.23 and 3.30, Jacobian matrix J_θ is derived as shown in Eq. 5.3 and the joint position control signal $\boldsymbol{U_\theta(t)}$ is generated as shown in Eq. 5.4.

$$\dot{\boldsymbol{\theta}} = J_{ad}\begin{bmatrix} E_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & J_e \end{bmatrix}\dot{\boldsymbol{\chi}}_{\boldsymbol{E}} = J_\theta\dot{\boldsymbol{\chi}}_{\boldsymbol{E}}. \tag{5.3}$$

$$\boldsymbol{U_\theta(t)} = J_\theta[k_p(\boldsymbol{\chi_d(t)} - \boldsymbol{\chi_m(t)}) + k_d(\dot{\boldsymbol{\chi}}_{\boldsymbol{d}}\boldsymbol{(t)} - \dot{\boldsymbol{\chi}}_{\boldsymbol{m}}\boldsymbol{(t)}) + k_i\int_0^t(\boldsymbol{\chi_d(t)} - \boldsymbol{\chi_m(t)})dt], \tag{5.4}$$

where k_p, k_i, k_d are constants, and $\boldsymbol{\chi_m(t)}$ is the visual measurement of the end-effector pose w.r.t. the base frame obtained from VXelements software, while in the simulation $\boldsymbol{\chi_m(t)}$ is replaced by $\boldsymbol{\chi_s(t)}$ which is the pose calculated by using the forward dynamic model.

In addition, the PID controller, given in Eq. 5.5, is used to describe the built-in joint controller in each joint and is employed in the simulated model:

$$\boldsymbol{\tau_a}(t) = l_p(\boldsymbol{U_\theta(t)} - \boldsymbol{\theta_s(t)}) + l_d(\boldsymbol{\dot{U}_\theta(t)} - \boldsymbol{\dot{\theta}_s(t)}) + l_i \int_0^t (\boldsymbol{U_\theta(t)} - \boldsymbol{\theta_s(t)})dt, \tag{5.5}$$

where l_i, l_p, l_d are the PID parameters to be identified, and $\boldsymbol{\theta_s(t)}$ is the joint angles, which can be obtained by analytically solving the inverse kinematics.

The real plant output $\boldsymbol{Y_m} = [\boldsymbol{\chi_m(1)}, \boldsymbol{\chi_m(2)}, \cdots, \boldsymbol{\chi_m(k)},]^T$ and the simulation output $\boldsymbol{Y_s} = [\boldsymbol{\chi_s(1)}, \boldsymbol{\chi_s(2)}, \cdots, \boldsymbol{\chi_s(k)}]^T$ are the variables in the optimization problem. The parameters to be identified, $\boldsymbol{\mathit{\Lambda}}$, can be denoted as $\boldsymbol{\mathit{\Lambda}} = [\boldsymbol{\mathit{\Xi}}^T, l_p, l_i, l_d]^T$. Accordingly the identification of $\boldsymbol{\mathit{\Lambda}}$ can be converted to solving the following nonlinear optimization problem:

$$min\ \mathit{\Phi}(\boldsymbol{\mathit{\Lambda}}) = \parallel \boldsymbol{Y_m} - \boldsymbol{Y_s} \parallel^2. \tag{5.6}$$

Then the updating formula for $\boldsymbol{\mathit{\Lambda}}$ is given as follows:

$$\boldsymbol{\mathit{\Lambda}}^{r+1} = (J_{\mathit{\Phi}}^T J_{\mathit{\Phi}})^{-1} J_{\mathit{\Phi}}^T \mathit{\Phi}(\boldsymbol{\mathit{\Lambda}}^r) + \boldsymbol{\mathit{\Lambda}}^r, \tag{5.7}$$

where $\boldsymbol{\mathit{\Lambda}}^r$ is the value of $\boldsymbol{\mathit{\Lambda}}$ in the rth iteration and $J_{\mathit{\Phi}}$ is the Jacobian matrix of $\mathit{\Phi}(\boldsymbol{\mathit{\Lambda}})$ w.r.t. $\boldsymbol{\mathit{\Lambda}}$ given as:

$$J_{\mathit{\Phi}} = \begin{bmatrix} \frac{\partial \mathit{\Phi}(\boldsymbol{\mathit{\Lambda}})}{\partial \boldsymbol{\mathit{\Lambda}_1}} & \frac{\partial \mathit{\Phi}(\boldsymbol{\mathit{\Lambda}})}{\partial \boldsymbol{\mathit{\Lambda}_2}} & \cdots & \frac{\partial \mathit{\Phi}(\boldsymbol{\mathit{\Lambda}})}{\partial \boldsymbol{\mathit{\Lambda}_{67}}} \end{bmatrix}, \tag{5.8}$$

where $\boldsymbol{\mathit{\Lambda}_i}$ denotes the i^{th} column of $\boldsymbol{\mathit{\Lambda}}$. The terminate criteria is given as:

$$\begin{aligned} &\frac{\parallel \mathit{\Phi}(\boldsymbol{\mathit{\Lambda}}^{r+1}) - \mathit{\Phi}(\boldsymbol{\mathit{\Lambda}}^r) \parallel}{\parallel \mathit{\Phi}(\boldsymbol{\mathit{\Lambda}}^r) \parallel} \leq tol_1, \\ &\max \left| \frac{\boldsymbol{\mathit{\Lambda}}_i^{r+1} - \boldsymbol{\mathit{\Lambda}}_i^r}{\boldsymbol{\mathit{\Lambda}}_i^r} \right| \leq tol_2, \quad i = 1, \cdots, n, \end{aligned} \tag{5.9}$$

where $|\cdot|$ denotes the absolute-value norm operation, tol_1 and tol_2 are the thresholds to be chosen for tuning the accuracy. A compromise should be made between the convergence speed and accuracy when choosing thresholds. To achieve good results in solving the nonlinear optimization problem, a proper initial guess of $\boldsymbol{\mathit{\Lambda}}$ is needed. For the dynamic parameters of the parallel robots, the initial guess can be calculated from manufacturer specifications. Then a priori PID parameters are obtained based on the simulation model.

5.4 Modified Exciting Trajectory

The exciting trajectories for the dynamic model identification of the serial robots are usually determined based on Finite Fourier series [107]. The Finite Fourier series-based exciting trajectory has been tested in a large amount of research works for identification purpose [108]. For serial robots, it can be represented by the following:

$$\begin{aligned}
\theta_i(t) &= \sum_{l=1}^{n}[\frac{sin(2\pi f_0 lt)}{2\pi f_0 l}s_i^l - \frac{cos(2\pi f_0 lt)}{2\pi f_0 l}c_i^l] + \theta_{0_i}, \\
\dot{\theta}_i(t) &= \sum_{l=1}^{n}[cos(2\pi f_0 lt)s_i^l + sin(2\pi f_0 lt)c_i^l], \\
\ddot{\theta}_i(t) &= \sum_{l=1}^{n}[-2\pi f_0 l sin(2\pi f_0 lt)s_i^l + 2\pi f_0 lt cos(2\pi f_0 lt)c_i^l],
\end{aligned} \tag{5.10}$$

where $\theta_i(t)$ is the ith joint angle trajectory of serial robots, n is the harmonics number, f_0 is the fundamental frequency, and s_i^l, c_i^l, θ_{0_i} are the trajectory parameters to be optimized. Instead of choosing the joint space states $(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}})$ for serial robots, the pose in the workspace is used for the dynamic identification of parallel robots. A modified Finite Fourier series-based exciting trajectory for parallel robots is proposed as:

$$\chi_i(t) = \sum_{l=1}^{n}[\frac{sin(2\pi\omega_0 lt)}{2\pi\omega_0 l}s_i^l - \frac{cos(2\pi\omega_0 lt)}{2\pi\omega_0 l}c_i^l] + \chi_{0_i}, \tag{5.11}$$

where $\chi_i(t)$ is the i^{th} column of the pose trajectory.

Therefore, $2n+1$ parameters $\boldsymbol{\delta_i} = [s_i^1, c_i^1, \cdots, s_i^n, c_i^n, \chi_{0_i}]^T$ can be estimated by solving a nonlinear optimization problem. The maximum wrench rotation range inside the singularity-free domain of the 6-RSS parallel robot is $(-57.1°, 57.1°)$. The inverse kinematic model is used to map the poses into the joint space and to check if the joint angles stay inside the singularity-free domain. To obtain the workspace states in the regression matrix H, the time derivatives of the Euler-angle should be converted to the angular velocity and acceleration, as shown in Eq. 3.21 and 5.12.

$$\dot{\boldsymbol{\omega}} = \begin{bmatrix} \ddot{\alpha} + \ddot{\gamma} s\beta + \dot{\beta}\dot{\gamma} c\beta \\ c\alpha(\ddot{\beta} - \dot{\alpha}\dot{\gamma} c\beta) + s\alpha(\dot{\beta}\dot{\gamma} s\beta - \dot{\beta}\dot{\alpha} - \ddot{\gamma} c\beta) \\ s\alpha(\ddot{\beta} - \dot{\alpha}\dot{\gamma} c\beta) + c\alpha(\dot{\beta}\dot{\alpha} + \ddot{\gamma} c\beta - \dot{\beta}\dot{\gamma} s\beta) \end{bmatrix}. \tag{5.12}$$

As shown in Eq. 5.2, the observability index of H should be maximized to achieve a good identification result for given the exciting trajectories. The observability index O_{in} used in Section 4.3 is chosen as the criteria. Therefore, the optimal exciting trajectory can be obtained by solving the following nonlinear optimization problem:

$$max \;\; O_{in}(\delta) = \frac{\sqrt[\varsigma]{\sigma_1 \sigma_2 \cdots \sigma_\varsigma}}{\sqrt{m}} = \frac{\sqrt[\varsigma]{det(\sqrt{H^T H})}}{\sqrt{m}}, \tag{5.13}$$

where the singular values of H are denoted by $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_\varsigma$, m is the number of sampled poses of the trajectory, ς is the number of dynamic parameters to be identified.

5.5 The Procedure of Identification

The whole procedure of the proposed closed-loop output-error identification method is given in Figure 5.2. Firstly, the dynamic model of the parallel robot is derived as Eq. 3.41 and 5.1. Then the optimized exciting trajectory, $\boldsymbol{\chi_d}(t)$, can be generated by using the method mentioned in previous section. By using $\chi_d(t)$ as reference input signal to the outer loop visual servoing control systems of both the real plant and the simulated model, the measured output pose $\boldsymbol{\chi_m}(t)$ of the parallel robot by the optical CMM sensor is compared with that of the simulated model. The identification of the parameters $\boldsymbol{\Lambda}$ is carried out by solving the nonlinear optimization problem. Lastly, the identified model can be validated by feeding several testing trajectories to the systems.

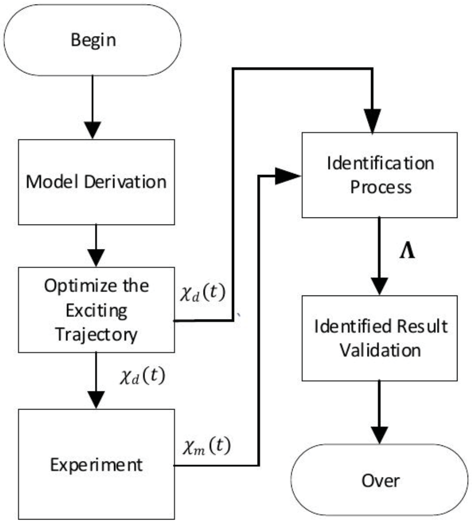


Figure 5.2: Sketch of the identification procedure.

5.6 Simulation and Experiment Results

In this section, the dynamic model is validated by the simulation using Matlab/SimMechanics. In addition, the closed-loop output-error identification is carried out on a 6-RSS parallel robot. An outer loop visual servoing controller is implemented on the real plant and the simulated model individually. The C-track 780 from Creaform Inc. is adopted to measure the pose of the end-effector of parallel robot.

5.6.1 Model Validation

The analytical dynamic model is rather complex and it is a non-trival task to ensure that the code of dynamic model is mistake free. A simulation validation method is used to validate the built mathematical dynamic model. A mechanical model of the 6-RSS parallel robot is built by using the Multibody SimMechanics Toolbox of Matlab/Simulink. The SimMechanics model is constructed by choosing the parts from SimMechanics library as shown in Figure 5.3. The coordinates frames and physical parameters of rigid body blocks can be specified in the setting option. The revolute and spherical joints are used to connect the rigid bodies. The body sensor part can provide the position and orientation of the coordinate frames. The entire model is directly actuated by the torque of the motors, and 3D animation shown in Figure 5.4 can be provided by SimMechanics. It should be noted that SimMechanics model can only simulate the forward dynamics and cannot be used for controller design. However, it is relatively easy and intuitive to build SimMechanics model with high fidelity [109]. To validate the mathematical dynamic model Eq. 3.41, the explicit form of the forward dynamic model obtained by Eq. 5.14 is employed to compare with the SimMechanics model.

$$\dot{\boldsymbol{v}}_{\boldsymbol{E}} = M(\boldsymbol{\chi}_{\boldsymbol{E}})^{-1}(\boldsymbol{\tau}_{\boldsymbol{g}} - C(\boldsymbol{\chi}_{\boldsymbol{E}}, \boldsymbol{v}_{\boldsymbol{E}})\boldsymbol{v}_{\boldsymbol{E}} - G(\boldsymbol{\chi}_{\boldsymbol{E}}) - \boldsymbol{\tau}_{\boldsymbol{f}}), \tag{5.14}$$

The mathematical dynamic model is built by using S-function of Simulink. The initial dynamic model parameters of both SimMechanics and mathematical models are derived from manufacturer specifications and are given in Table 5.1.

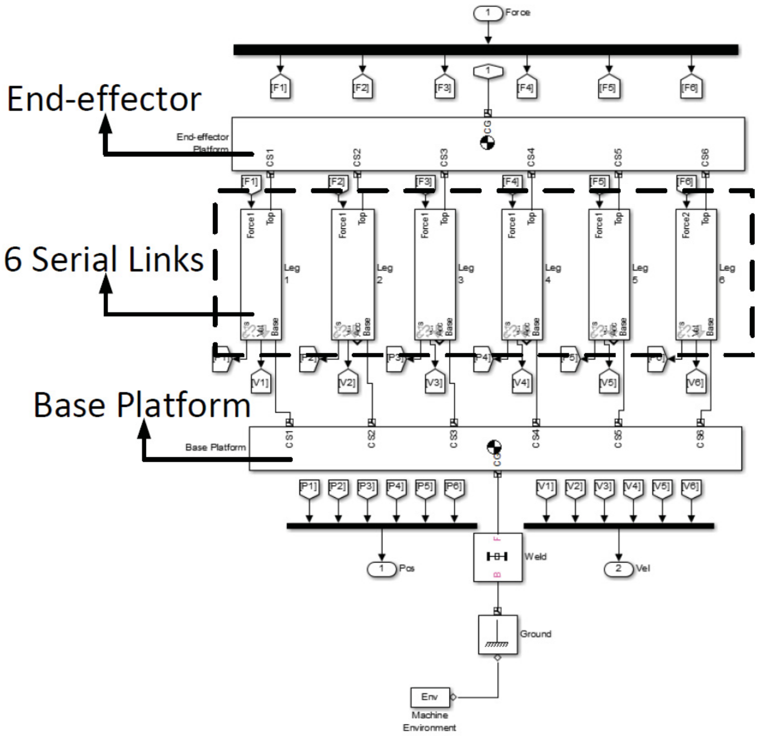


Figure 5.3: Mechanical model of 6-RSS parallel robot built by SimMechanics.

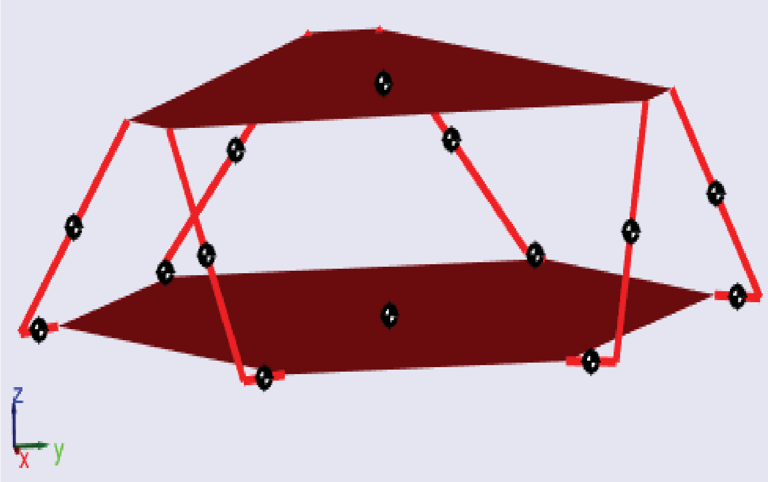


Figure 5.4: 3D animation of 6-RSS parallel robot. `https://youtu.be/HXtCvgkn2jw`.

Table 5.1: Initial dynamic model parameters of 6-RSS parallel robot.

Dynamic Model Parameters	**Initial Value**
m_p (kg)	24.0
I_{x_p} 10^{-2} (kg · m^2)	17.8
I_{y_p} 10^{-2} (kg · m^2)	17.8
I_{z_p} 10^{-2} (kg · m^2)	35.0
m_{wi} 10^{-2} (kg)	68.5
$I_{x_{w_i}}$ 10^{-5} (kg · m^2)	22.9
$I_{y_{w_i}}$ 10^{-5} (kg · m^2)	22.9
$I_{z_{w_i}}$ 10^{-5} (kg · m^2)	60.5
m_{li} (kg)	1.31
$I_{x_{l_i}}$ 10^{-5} (kg · m^2)	52.3
$I_{y_{l_i}}$ 10^{-5} (kg · m^2)	52.3
$I_{z_{l_i}}$ 10^{-4} (kg · m^2)	21.3

As shown in Figure 5.5, a simple PID controller is used to stabilize both mathematical and SimMechanic models with the same PID gains. The exciting trajectory $\chi_d(t)$ derived from Equation (5.10) is used as the reference input signal. The outputs of the SimMechanics and mathematical

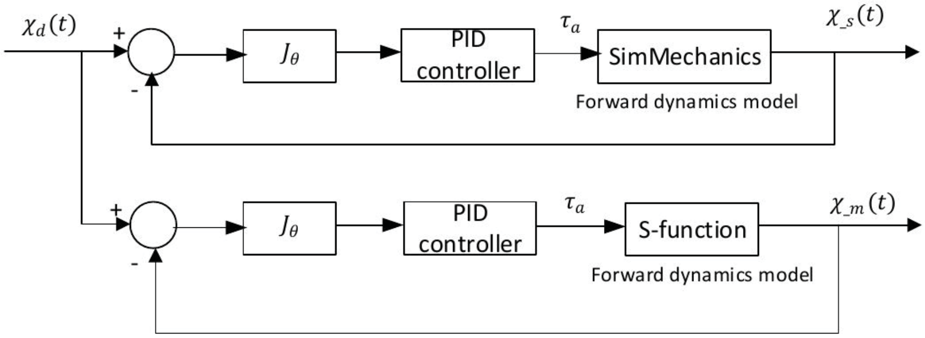


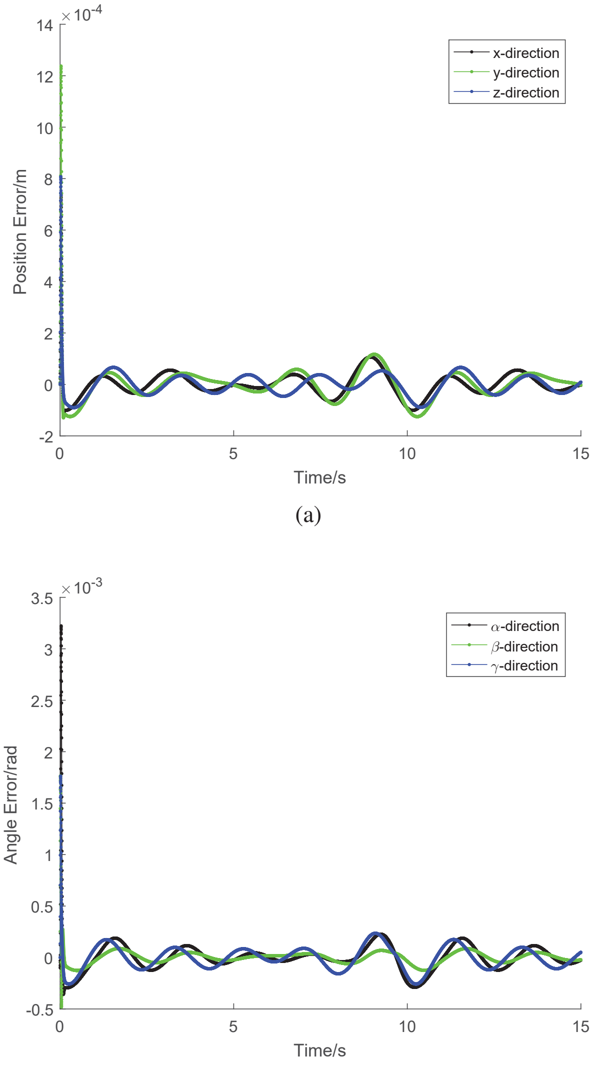
Figure 5.5: Dynamic model validation block diagram of 6-RSS parallel robot.

dynamic model (Eq. 5.14) are $\boldsymbol{\chi_{\_s}(t)}$ and $\boldsymbol{\chi_{\_m}(t)}$ respectively. The difference between $\boldsymbol{\chi_{\_s}(t)}$ and $\boldsymbol{\chi_{\_m}(t)}$ is shown in Figure 5.6. The maximum position and angle errors are around 1.25 mm and 3.25×10^{-3} rad, which occur at the beginning of the simulation, and are often caused by the kinematic error. The largest steady-state errors are about 0.1 mm in the position and 0.25×10^{-3} rad in the angle, which can prove the correctness of the mathematical model. The validated mathematical model can be used in the simulation part for the subsequent identification.

5.6.2 Identification Experiment

The experiment setup described in Section 4.5 is used for the identification experiment. To eliminate the high frequency noise of the pose measurement from the optical CMM, measurement data is filtered by the zero-phase forward and reverse 8th order Butterworth filter with the cut-off frequency 60 Hz. The filtering process is carried out by the Zero-phase digital filtering function of Maltab, $filtfilt$.

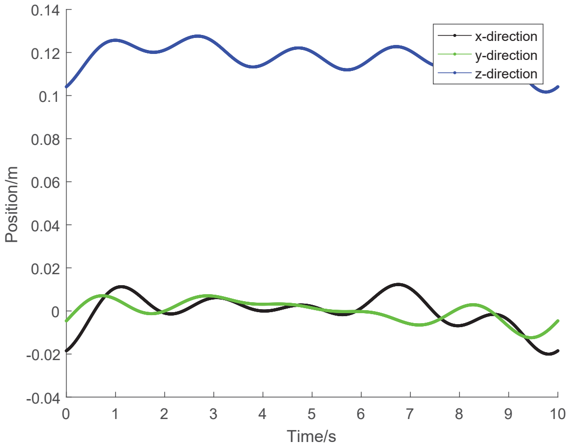
The optimization of the exciting trajectory is carried out by using the GA function of the Optimization Toolbox of Matlab. The GA algorithm uses binary code to represent the harmonic parameters. The fundamental frequency f_0 is selected as 0.1 Hz and the harmonics number n is chosen as 5. By taking the observation index as the fitness function, the binary code is updated to maximize the fitness value in each step. The starting value of the observation index O_{in} is 0.38 and the stop criteria is set as 10^{-10}. The algorithm stops after 324 iterations with the maximum O_{in} 1.345. The derived optimal exciting trajectory is given in Figure 5.7.



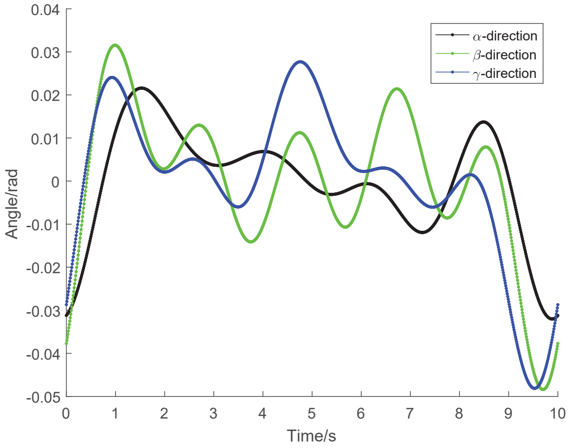
(a)

(b)

Figure 5.6: Simulation results of SimMechanics model. (**a**) Position error; (**b**) Angle error.



(a)



(b)

Figure 5.7: Optimal exciting trajectory. (**a**) Positional trajectory; (**b**) Angular trajectory.

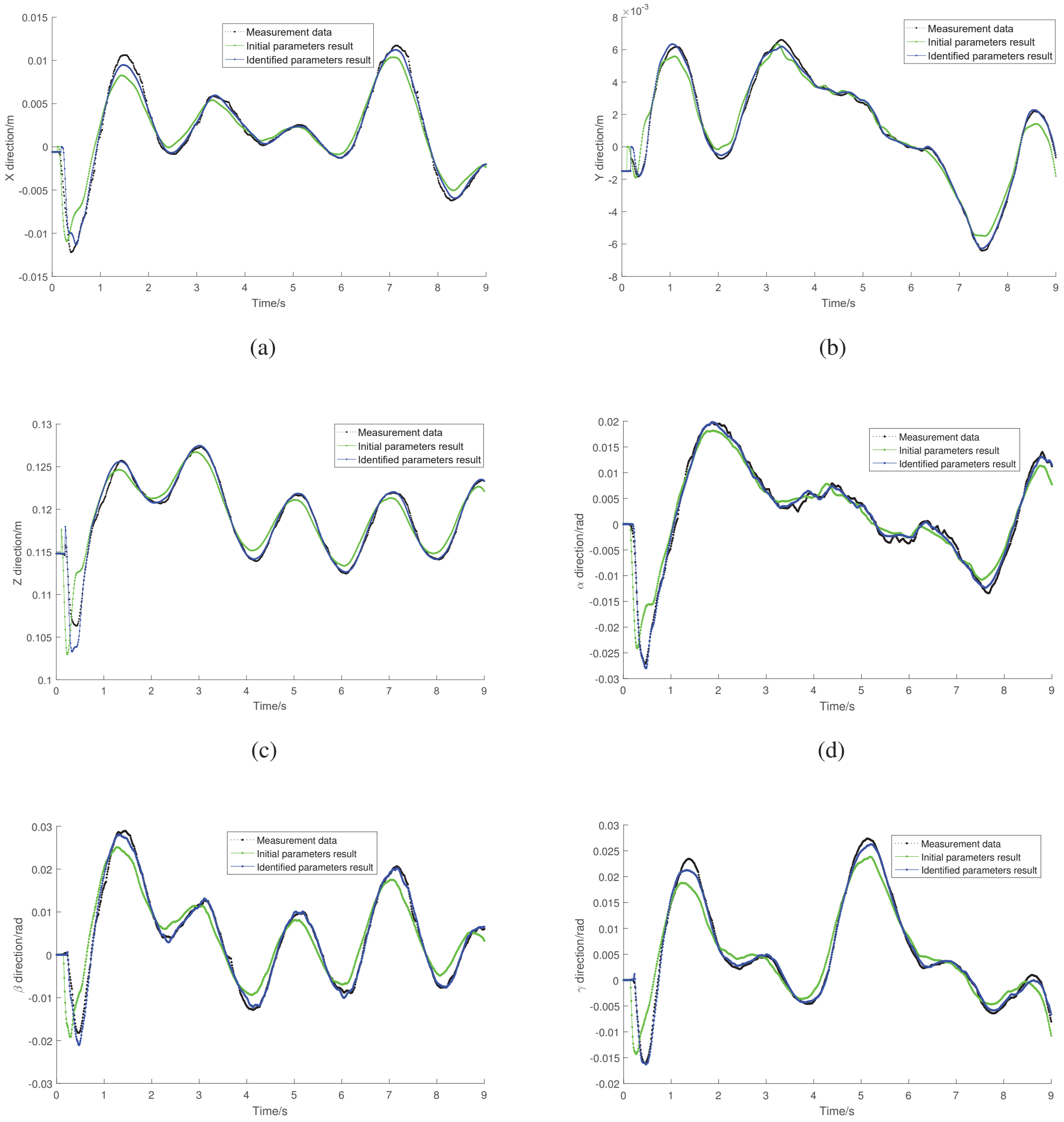
Table 5.2: Identified parameters of 6-RSS parallel robot.

Parameters	**Initial Value**	**Identified Value**	**Parameters**	**Initial Value**	**Identified Value**
m_p (kg)	24.0	23.5	m_{l4} (kg)	1.31	1.24
I_{x_p} 10^{-2} $(\text{kg}\cdot\text{m}^2)$	17.8	16.1	$I_{x_{l_4}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	12.9
I_{y_p} 10^{-2} $(\text{kg}\cdot\text{m}^2)$	17.8	14.4	$I_{y_{l_4}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	39.2
I_{z_p} 10^{-2} $(\text{kg}\cdot\text{m}^2)$	35.0	33.7	m_{l5} (kg)	1.31	1.33
m_{w1} 10^{-2} (kg)	68.5	67.1	$I_{x_{l_5}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	30.7
$I_{z_{w_1}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	60.5	69.4	$I_{y_{l_5}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	49.7
m_{w2} 10^{-2} (kg)	68.5	68.7	m_{l6} (kg)	1.31	1.34
$I_{z_{w_2}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	60.5	10.0	$I_{x_{l_6}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	47.0
m_{w3} (kg) 10^{-2} (kg)	68.5	68.0	$I_{y_{l_6}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	50.9
$I_{z_{w_3}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	60.5	22.6	f_{c_1}	0	0.104
m_{w4} 10^{-2} (kg)	68.5	67.7	f_{v_1}	0	0.148
$I_{z_{w_4}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	60.5	76.0	f_{c_2}	0	0.111
m_{w5} (kg) 10^{-2} (kg)	68.5	68.3	f_{v_2}	0	0.187
$I_{z_{w_5}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	60.5	44.1	f_{c_3}	0	0.0336
m_{w6} (kg) 10^{-2} (kg)	68.5	68.4	f_{v_3}	0	0.0993
$I_{z_{w_6}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	60.5	69.6	f_{c_4}	0	0.147
m_{l1} (kg)	1.31	1.33	f_{v_4}	0	0.854
$I_{x_{l_1}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	63.0	f_{c_5}	0	0.104
$I_{y_{l_1}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	68.4	f_{v_5}	0	0.0803
m_{l2} (kg)	1.31	1.32	f_{c_6}	0	0.0828
$I_{x_{l_2}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	60.7	f_{v_6}	0	0.0349
$I_{y_{l_2}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	71.8	l_p	10	10.5
m_{l3} (kg)	1.31	1.25	l_i	12	11.4
$I_{x_{l_3}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	22.4	l_d	0.1	0.164
$I_{y_{l_3}}$ 10^{-5} $(\text{kg}\cdot\text{m}^2)$	52.3	35.5			

The identification procedure is implemented off-line through simulation illustrated in Section 5.3. The optimization procedure is carried out by using optimization functions in Matlab R2016a. The minimal performance requirements of the computation platform are given as: 4 cores Intel or AMD Processor; 6 GB disk space; 4 GB RAM. The same outer loop visual servoing controller, Eq.

5.4), is employed in the simulation. The gains of the visual servoing controller are obtained through trial and error in the extensive experimental tests. The well-tuned gains are given as $k_p = 0.3, k_d = 0.001, k_i = 2.4$. And the built-in joint PID controllers, Eq. 5.5, are also implemented in the simulation. The dynamic parameters derived from manufacturer specifications are used as initial values, and the initial values of the PID gains in Eq. 5.5 are obtained through trial and error based on the simulation model. During the tuning, in the simulation system, the inertial and friction parameters are set as the values based on the manufacturer specifications, and the visual servoing controller gains are set as the same values of the controller of the real system. The identified parameters, given in Table 5.2, are derived by using $lsqnonlin$ function of the Optimization Toolbox of Matlab after 8 iterations. A total 50 out of 67 parameters are identified, and are used in the simulation model (Eq. 5.14) to capture the dynamic characteristic of the parallel robot. The other parameters neither contribute to nor have slight impact on the dynamics of the parallel robot. Those parameters can be eliminated by using QR decomposition on the regression matrix H [61, 110]. If any diagonal elements of $\underline{R}$ are smaller than the pre-defined small number, i.e., $R_{ii} < \varepsilon$, where ε is chosen as 10^{-3} in the research, the corresponding columns of the regression matrix H are deleted. By doing so, matrix H is better conditioned and the identification procedure is sped up.

Then, the pose trajectories are generated by using the identified parameters in the simulation, and are compared with the pose measurement, as shown in Figure 5.8. Table 5.3 shows the root-mean-square (RMS) levels of the pose trajectory errors.



(a) (b) (c) (d) (e) (f)

Figure 5.8: The pose trajectories of the parallel robot: the measurement of the real plant (**black dot**), the output of the simulation with initial parameters (**green line**), the output of the simulation with identified parameters (**blue line**), (**a**) Along X Direction; (**b**) Along Y Direction; (**c**) Along Z Direction; (**d**) Around α Axis; (**e**) Around β Axis; (**f**) Around γ Axis.

Table 5.3: The RMS levels of the pose trajectory errors.

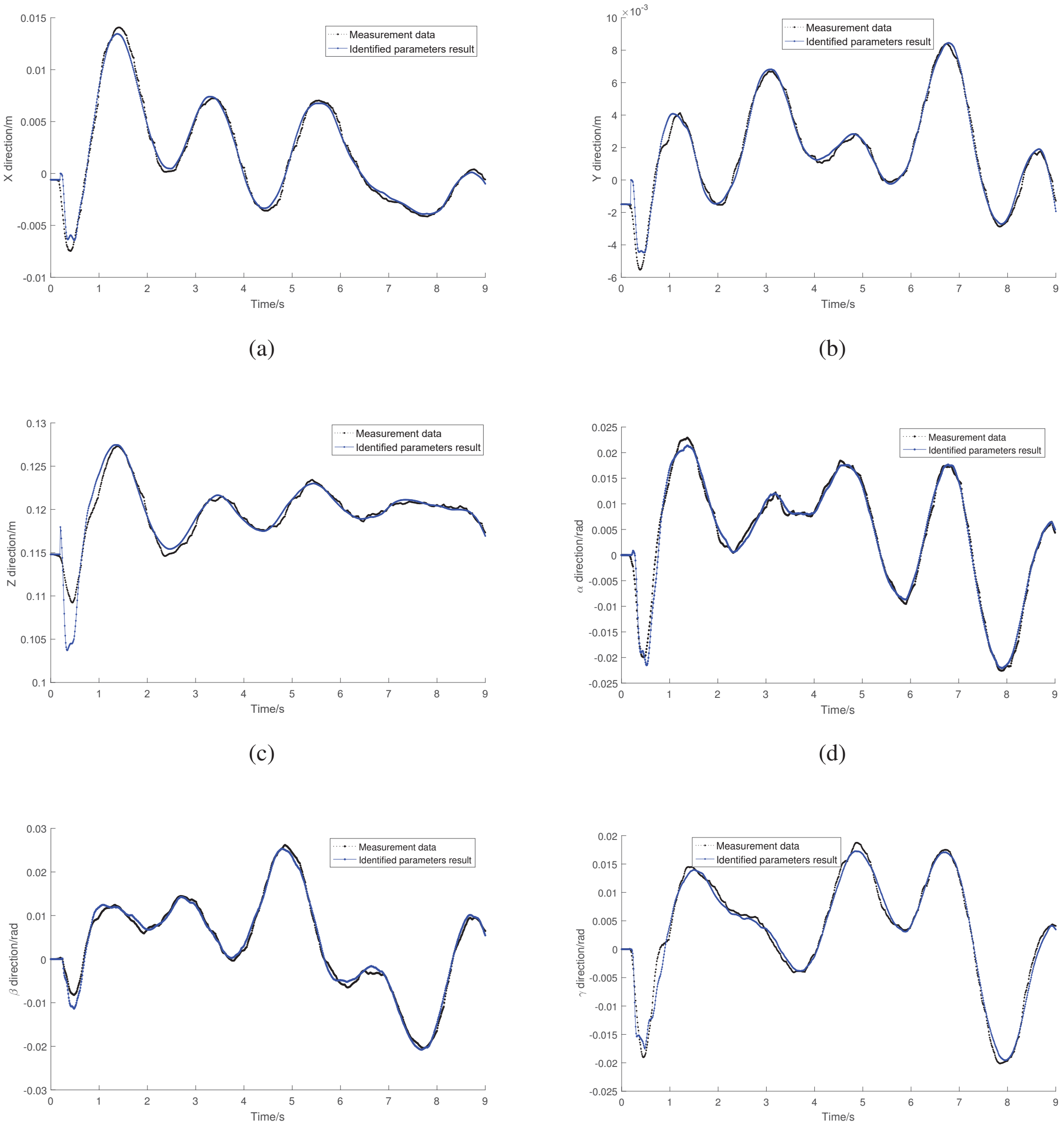
	Before Identification	**After Identification**
x direction (mm)	1.26	0.408
y direction (mm)	1.16	0.235
z direction (mm)	1.55	0.494
α direction 10^{-3} (rad)	2.52	0.956
β direction 10^{-3} (rad)	3.58	0.797
γ direction 10^{-3} (rad)	2.80	0.725

5.6.3 Identified Results Validation

To validate the identified parameters, ten more trajectories are generated according to Eq. 5.11 with random harmonic parameters under the singularity constraint. The generated trajectories are used as desired trajectories, and are fed to the parallel robot and the identified model in the simulation respectively. The RMS levels of the pose trajectory errors are given in Table 5.4, and the measurement and the simulated pose trajectories are given in Figure 5.9 according to the 1st desired trajectory. The RMS of the position and orientation errors for all ten trajectories are below 0.8 mm and 1.4×10^{-3} rad respectively, which validate the identified results of previous subsection. In addition, the proposed identification procedure is implemented based on the ten trajectories to analyze the statistic property of the identification results. After deriving ten more groups of identified parameters, the variation measure of the identification results are given in Table 5.5. The highest relative variation of the parameter is below 25%, which is acceptable. It has been stated that less than 30 percent in the variation measure of the parameters gives a good match to the real system [111].

Table 5.4: The RMS levels of the ten validation trajectory errors.

	x (mm)	y (mm)	z (mm)	α 10^{-3} (rad)	β 10^{-3} (rad)	γ 10^{-3} (rad)
1st	0.517	0.384	0.709	1.091	0.963	0.966
2nd	0.273	0.410	0.522	1.157	1.071	0.932
3rd	0.381	0.403	0.600	1.242	0.830	1.143
4th	0.341	0.511	0.617	1.136	0.814	1.161
5th	0.322	0.394	0.464	1.219	0.877	0.835
6th	0.310	0.301	0.441	1.195	1.111	1.040
7th	0.360	0.402	0.455	1.263	1.176	1.024
8th	0.418	0.483	0.460	1.251	1.211	1.015
9th	0.342	0.510	0.557	1.411	1.156	0.711
10th	0.318	0.379	0.473	1.219	1.023	0.905



(a) (b)

(c) (d)

(e) (f)

Figure 5.9: The pose trajectories of the parallel robot: the measurement of the real plant (**black dot**), the output of the simulation with identified parameters (**blue line**), (**a**) Along X Direction; (**b**) Along Y Direction; (**c**) Along Z Direction;(**d**) Around α Axis; (**e**) Around β Axis; (**f**) Around γ Axis.

Table 5.5: Variation measure of the identification result.

Parameters	**Variation Measure**	**Parameters**	**Variation Measure**
m_p (kg)	0.48%	m_{l4} (kg)	19.1%
I_{x_p} $(\text{kg} \cdot \text{m}^2)$	5.0%	$I_{x_{l_4}}$ $(\text{kg} \cdot \text{m}^2)$	0.16%
I_{y_p} $(\text{kg} \cdot \text{m}^2)$	24.9%	$I_{y_{l_4}}$ $(\text{kg} \cdot \text{m}^2)$	0.06%
I_{z_p} $(\text{kg} \cdot \text{m}^2)$	8.6%	m_{l5} (kg)	10.1%
m_{w1} (kg)	0.56%	$I_{x_{l_5}}$ $(\text{kg} \cdot \text{m}^2)$	0.15%
$I_{z_{w_1}}$ $(\text{kg} \cdot \text{m}^2)$	6.6%	$I_{y_{l_5}}$ $(\text{kg} \cdot \text{m}^2)$	0.01%
m_{w2} (kg)	1.5%	m_{l6} (kg)	12.9%
$I_{z_{w_2}}$ $(\text{kg} \cdot \text{m}^2)$	1.4%	$I_{x_{l_6}}$ $(\text{kg} \cdot \text{m}^2)$	0.17%
m_{w3} (kg)	3.4%	$I_{y_{l_6}}$ $(\text{kg} \cdot \text{m}^2)$	0.02%
$I_{z_{w_3}}$ $(\text{kg} \cdot \text{m}^2)$	5.0%	f_{c_1}	15.9%
m_{w4} (kg)	3.7%	f_{v_1}	11.7%
$I_{z_{w_4}}$ $(\text{kg} \cdot \text{m}^2)$	3.8%	f_{c_2}	21.4%
m_{w5} (kg)	6.6%	f_{v_2}	13.9%
$I_{z_{w_5}}$ $(\text{kg} \cdot \text{m}^2)$	6.6%	f_{c_3}	9.0%
m_{w6} (kg)	0.79%	f_{v_3}	16.4%
$I_{z_{w_6}}$ $(\text{kg} \cdot \text{m}^2)$	1.7%	f_{c_4}	17.8%
m_{l1} (kg)	4.1%	f_{v_4}	15.2%
$I_{x_{l_1}}$ $(\text{kg} \cdot \text{m}^2)$	0.04%	f_{c_5}	15.1%
$I_{y_{l_1}}$ $(\text{kg} \cdot \text{m}^2)$	0.11%	f_{v_5}	21.8%
m_{l2} (kg)	5.2%	f_{c_6}	23.4%
$I_{x_{l_2}}$ $(\text{kg} \cdot \text{m}^2)$	0.05%	f_{v_6}	13.0%
$I_{y_{l_2}}$ $(\text{kg} \cdot \text{m}^2)$	0.12%	l_p	2.82%
m_{l3} (kg)	20.9%	l_i	0.44%
$I_{x_{l_3}}$ $(\text{kg} \cdot \text{m}^2)$	0.09%	l_d	3.7%
$I_{y_{l_3}}$ $(\text{kg} \cdot \text{m}^2)$	0.17%		

Therefore by using the proposed visual closed-loop output-error identification method, the identified dynamic model can approximate the real plant with acceptable accuracy.

5.7 Summary

In this chapter, a visual closed-loop output-error identification method based on an optical CMM sensor for parallel robots is proposed. An outer loop visual servoing controller is employed in both the real plant and the simulation model to stabilize the two systems. The benefits of the proposed method are summarized as follows: elimination of the need for the joint and torque measurements, the exact knowledge of the built-in joint controller of the industrial robots, and the time-consuming forward kinematics calculation. The correctness and accuracy of the built dynamic model are validated by the Matlab/SimMechanics simulation. The experimental test results show that the identified dynamic model can capture the dynamics of the real parallel robot with satisfactory accuracy. The proposed method can be easily applied to other types of industrial parallel robots with unknown PID built-in controller or its variants, such as 6 DOF Stewart platforms, 6 UPS and 6 RUS parallel robots etc. The complexity of those dynamic models is similar to that of the 6-RSS parallel robot. Since the analytical solution of the forward kinematics of those 6 DOF parallel robots does not exist, the proposed visual identification method does not need the forward kinematic model. Taking the advantages of the visual sensor, the dynamic model can be identified for the visual servoing purpose. In the next chapter, the advanced model-based visual servoing control method will be further studied to improve the tracking performance of parallel robots based on the identification results.

Chapter 6

Dynamic Model-based Visual Servoing Control of the 6-RSS Parallel Robots

6.1 Introduction

Parallel robots are a series of closed-loop structure mechanisms whose end-effectors are supported by multiple serial kinematic chains. By actuating several active legs of the parallel robot, the pose of the end-effector frame can be controlled to track the reference pose. The existence of closed-chain mechanism yields difficulties on the controller design. How to design effective controllers to improve the tracking accuracy of the parallel robots poses a challenge to the control community. Since the numerical solution of the forward kinematic model is time-consuming and is not suitable for the real-time controller of 6-DOF parallel robots, the researchers resort to the control design in the joint space to avoid solving the forward kinematic equations [6, 21, 83]. The joint space controller similar to that of the serial robots is designed for parallel robots by translating the desired pose trajectory of the end-effector frame in Cartesian space into the one in the joint space through the inverse kinematics [21]. This control strategy is easy to be implemented, due to the fact that inverse kinematics of parallel robots can be analytically solved, and the measured joint angles can be used as feedback signal in the joint space control loop. However, it is more desirable to directly control the pose of platform of parallel robot in Cartesian space (Cartesian space controller) than to control the joint angles or positions in the joint space.

Visual servoing methods can be utilized to design the Cartesian space controller for the parallel robots. Most recent visual servoing research work on parallel robots is dedicated to the kinematic level visual servoing design. The dynamics of parallel robots are usually ignored or treated as uncertainties. Although the kinematic level visual servoing of parallel robot can achieve the position tracking to some extent, the tracking performance will be sacrificed greatly without considering the dynamic model of the parallel robot. To include the dynamic model in the visual servoing, the control designers need the accurate dynamic model and design advanced model-based control algorithms. The other issue is the implementation of designed controller in the commercial industrial parallel robots.

In this chapter, a dynamic SMC (DSMC) scheme in the PBVS is proposed to improve the tracking performance of the 6-RSS parallel robot. The proposed control scheme adopts the optical CMM sensor to obtain the real time pose information of the end-effector of parallel robot and to use it as the feedback signal. The DSMC scheme is robust to the modeling errors and uncertainties. With the benefit of the position-to-torque converter, the proposed DSMC scheme can be implemented in the industrial parallel robot. The stability of the proposed scheme has been proved by using the Lyapunov function. The experimental tests of the proposed control scheme have been carried out on the 6-RSS parallel robot. The comparison with the kinematic level controllers demonstrates the superiority of the proposed dynamic level visual servoing.

The outlines of this chapter is given as: The dynamic model is built and its properties are presented in Section 6.2. The built-in controller of the parallel robot is introduced in Section 6.3. The proposed DSMC scheme is designed in Section 6.4. The experiment implementation and results are given in Section 6.5. And the summary is drawn in Section 6.6.

6.2 Dynamic Model Properties of the Parallel Robot

The dynamic model of the given 6-RSS parallel robot has been derived based on the principle of virtual work in Section 3.6. The workspace states $\boldsymbol{\chi_E}$, $\boldsymbol{v_E}$ and $\dot{\boldsymbol{v}}_{\boldsymbol{E}}$ are employed in Eq. 3.41. Due the existence of the nonlinear relationship between $\boldsymbol{v_E}$ and $\dot{\boldsymbol{\chi}}_{\boldsymbol{E}}$ given in Eq. 3.23, the convergence of $\boldsymbol{v_E}$ cannot guarantee that of $\dot{\boldsymbol{\chi}}_{\boldsymbol{E}}$. Therefore, it is necessary to rewrite Eq. 3.41 into the form in terms of the workspace states $\boldsymbol{\chi_E}$, $\dot{\boldsymbol{\chi}}_{\boldsymbol{E}}$ and $\ddot{\boldsymbol{\chi}}_{\boldsymbol{E}}$ for the dynamic visual servoing purpose.

Substitute $\boldsymbol{v_E}$ with Eq. 3.23 and left multiply J_s on both sides of Eq. 3.41, the following dynamic equation can be derived.

$$\widetilde{\boldsymbol{\tau}}_{\boldsymbol{g}} = \widetilde{M}(\boldsymbol{\chi_E})\ddot{\boldsymbol{\chi}}_{\boldsymbol{E}} + \widetilde{C}(\boldsymbol{\chi_E}, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}})\dot{\boldsymbol{\chi}}_{\boldsymbol{E}} + \widetilde{G}(\boldsymbol{\chi_E}) + \widetilde{\boldsymbol{\tau}}_{\boldsymbol{f}}, \tag{6.1}$$

in which

$$\begin{aligned}
\widetilde{M}(\boldsymbol{\chi_E}) =& \sum_{i=1}^{6}(J_s^T J_{a_i}^T M_{w_i} J_{a_i} J_s + J_s^T J_{b_i}^T M_{l_i} J_{b_i} J_s) + J_s^T M_p J_s, \\
\widetilde{C}(\boldsymbol{\chi_E}, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}}) =& \sum_{i=1}^{6}\Big(J_s^T J_{a_i}^T (M_{w_i} \dot{J}_{a_i} J_s + C_{w_i} J_{a_i} J_s) + J_s^T J_{b_i}^T (M_{l_i} \dot{J}_{b_i} J_s + C_{l_i} J_{b_i} J_s)\Big) \\
& + J_s^T (M_p \dot{J}_s + C_p J_s), \\
\widetilde{G}(\boldsymbol{\chi_E}) =& J_s^T G_p + \sum_{i=1}^{6}(J_s^T J_{a_i}^T G_{w_i} + J_s^T J_{b_i}^T G_{l_i}),
\end{aligned} \tag{6.2}$$

where J_{ad}, J_{a_i}, and $J_{b_i} \in \Re^{6\times 6}$ are the inverse Jacobian, wrench Jacobian and link Jacobian matrices; M_p, M_{w_i} and $M_{l_i} \in \Re^{6\times 6}$ are the mass matrices of the end-effector platform, wrenches and links; C_p, C_{w_i} and $C_{l_i} \in \Re^{6\times 6}$ are Coriolis and centrifugal matrices of the end-effector platform, wrenches and links respectively; G_p, G_{w_i} and $G_{l_i} \in \Re^6$ are gravity vectors of the end-effector platform, wrenches and links respectively. The details for the above items can be found in Section 3.6. The general force item, $\widetilde{\boldsymbol{\tau}}_{\boldsymbol{g}}$, can be derived by Eq. 6.3.

$$\widetilde{\boldsymbol{\tau}}_{\boldsymbol{g}} = J_s^T J_{ad}^T \boldsymbol{\tau_a}, \tag{6.3}$$

where $\boldsymbol{\tau_a} = [\tau_{a_1}, \tau_{a_2} ... \tau_{a_6}]^T \in \Re^6$ is the actuator torque vector exerted on the revolute joints. The Coulomb and viscous friction model has been employed in the dynamic identification of the parallel robot in Chapter 5. Then the friction item $\widetilde{\boldsymbol{\tau}}_{\boldsymbol{f}}$ in Eq.6.1 can be derived as:

$$\widetilde{\boldsymbol{\tau}}_{\boldsymbol{f}} = J_s^T J_{ad}^T \begin{bmatrix} f_{c_1} sign(J_{ad_1} J_s\, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}}) + f_{v_1} J_{ad_1} J_s\, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}} \\ f_{c_2} sign(J_{ad_2} J_s\, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}}) + f_{v_2} J_{ad_2} J_s\, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}} \\ \dots \\ f_{c_6} sign(J_{ad_6} J_s\, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}}) + f_{v_6} J_{ad_6} J_s\, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}} \end{bmatrix}, \tag{6.4}$$

where f_{c_i} and f_{v_i} are the Coulomb and viscous friction parameters of the i^{th} $(i = 1, 2, ..., 6)$ revolute rotation joint.

Property 1. *The inertial matrix* $\widetilde{M}(\boldsymbol{\chi}_{\boldsymbol{E}})$ *is symmetric.*

Proof 1 (Proof of Property 1)**.** *According to Eq. 6.2, the transpose of* $\widetilde{M}(\boldsymbol{\chi}_{\boldsymbol{E}})$ *can be derived as:*

$$\widetilde{M}(\boldsymbol{\chi}_{\boldsymbol{E}})^T = \sum_{i=1}^{6} (J_s^T J_{a_i}^T M_{w_i}^T J_{a_i} J_s + J_s^T J_{b_i}^T M_{l_i}^T J_{b_i} J_s) + J_s^T M_p^T J_s, \tag{6.5}$$

The inertial matrix M_* *of any moving body of the parallel robot can be represented as:*

$$M_* = \begin{bmatrix} m_* E_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & I_* \end{bmatrix}, \tag{6.6}$$

where $E_{3\times3} \in \mathbb{R}^{3\times3}$ *denotes the identity matrix,* $I_* \in \mathbb{R}^{3\times3} = diag[I_{x*}, I_{y*}, I_{z*}]$ *is a diagonal matrix according to Assumptions 1 and 2. Therefore* $M_*^T = M_*$ *holds. Moreover,* $\widetilde{M}(\boldsymbol{\chi}_{\boldsymbol{E}})^T = \widetilde{M}(\boldsymbol{\chi}_{\boldsymbol{E}})$ *can be derived.*
Q.E.D.

Property 2. *The inertial matrix* $\widetilde{M}(\boldsymbol{\chi}_{\boldsymbol{E}})$ *and the Coriolis and centrifugal matrix* $\widetilde{C}(\boldsymbol{\chi}_{\boldsymbol{E}}, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}})$ *satisfy the following relationship:*

$$\boldsymbol{x}^T(\,\dot{\widetilde{M}}(\boldsymbol{\chi}_{\boldsymbol{E}}) - 2\widetilde{C}(\boldsymbol{\chi}_{\boldsymbol{E}}, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}}))\boldsymbol{x} = 0 \qquad \forall \boldsymbol{x} \in \mathbb{R}^6. \tag{6.7}$$

Proof 2 (Proof of Property 2)**.** *Substituting Eq. 6.2 into the items in Eq. 6.7 yields Eq. 6.8.*

$$
\begin{aligned}
&\dot{\widetilde{M}}(\boldsymbol{\chi_E}) - 2\widetilde{C}(\boldsymbol{\chi_E}, \dot{\boldsymbol{\chi}}_{\boldsymbol{E}}) = 2J_s^T M_p \dot{J}_s + J_s^T \dot{M}_p J_s - 2J_s^T (M_p \dot{J}_s + C_p J_s) \\
&+ \sum_{i=1}^{6} (J_s^T J_{a_i}^T \dot{M}_{w_i} J_{a_i} J_s + 2J_s^T J_{a_i}^T M_{w_i} \dot{J_{a_i}} J_s + J_s^T J_{b_i}^T \dot{M}_{l_i} J_{b_i} J_s + 2J_s^T J_{b_i}^T M_{l_i} \dot{J_{b_i}} J_s) \\
&- 2\sum_{i=1}^{6} \Big(J_s^T J_{a_i}^T (M_{w_i} \dot{J_{a_i}} J_s + C_{w_i} J_{a_i} J_s) + J_s^T J_{b_i}^T (M_{l_i} \dot{J_{b_i}} J_s + C_{l_i} J_{b_i} J_s) \Big) \\
&= J_s^T (\dot{M}_p - 2C_p) J_s + \sum_{i=1}^{6} (J_s^T J_{a_i}^T (\dot{M}_{w_i} - 2C_{w_i}) J_{a_i} J_s + J_s^T J_{b_i}^T (\dot{M}_{l_i} - 2C_{l_i}) J_{b_i} J_s).
\end{aligned}
\tag{6.8}
$$

The Coriolis and centrifugal matrix C_* can be presented as:

$$
C_* = \begin{bmatrix} \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \boldsymbol{\omega}_* \times I_* \end{bmatrix}, \tag{6.9}
$$

The following relationship holds for any moving body in Eq. 6.8.

$$
\dot{M}_* - 2C_* = \begin{bmatrix} \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \dot{I}_* - 2\boldsymbol{\omega}_* \times I_* \end{bmatrix}. \tag{6.10}
$$

The moment of inertial I_* can be obtained by $I_* = R_* I'_* R_*^T$. $I'_* \in \mathbb{R}^{3\times 3}$ is the moment of inertial with respect to the moving body frame and $R_* \in SO(3)$ is the rotation matrix. Then Eq. 6.10 can be rewritten as:

$$
\begin{aligned}
\dot{M}_* - 2C_* &= \begin{bmatrix} \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \dot{R}_* I'_* R_*^T + R_* I'_* \dot{R}_*^T - 2\dot{R}_* I'_* R_*^T \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \end{bmatrix}.
\end{aligned}
\tag{6.11}
$$

Substituting Eq. 6.8 and 6.11 into Eq. 6.7, Property 2 can be derived.

Q.E.D.

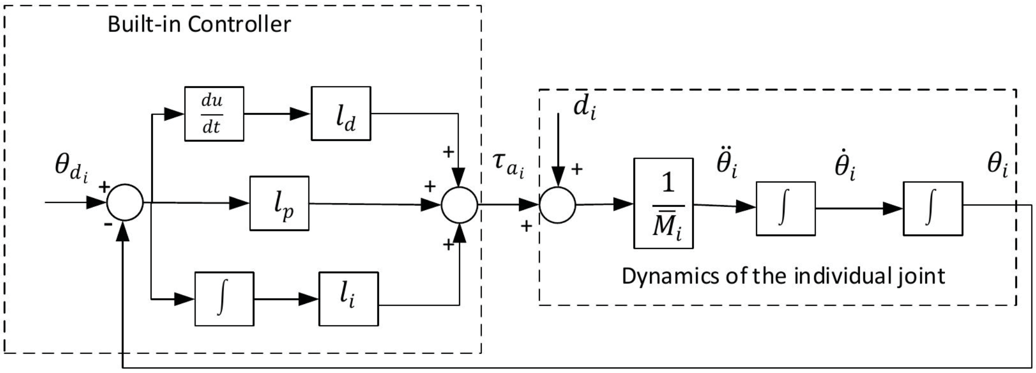


Figure 6.1: The block diagram of the built-in controller for the individual joint

6.3 Built-in Controller of the 6-RSS Parallel Robot

The controllers of the industrial robots are normally integrated with the functions of motion trajectory generation, human user interface, and the built-in real-time motion controller. The classical independent-joint proportional integral derivative (PID) controller is commonly used as the built-in control method due to its simplicity in design and implementation. The built-in controller can be identified by the methods introduced in [59, 60, 79, 80]. By considering PID as the built-in controller of the parallel robot, the controller gains and the dynamic parameters of the 6-RSS parallel robot can be identified simultaneously through the closed-loop identification method according to the result in Chapter 5.

The built-in controller of the 6-RSS parallel robot is illustrated as Fig. 6.1. The dynamic model of the parallel robot, Eq. 3.41, can be rewritten as the decoupled double integrator model with disturbance in individual joint space, such as

$$\begin{aligned}\tau_{a_i} &= \sum_{j=1}^{n} M_{i,j}\ddot{\theta}_j + \sum_{j=1}^{n} C_{i,j}\dot{\theta}_j + G_i + \tau_{f_i} \\ &= M_{i,i}\ddot{\theta}_i + \sum_{j\neq i}^{n} M_{i,j}\ddot{\theta}_j + \sum_{j=1}^{n} C_{i,j}\dot{\theta}_j + G_i + \tau_{f_i} \\ &= M_{i,i}\ddot{\theta}_i + d_i,\end{aligned} \tag{6.12}$$

where the coupling, friction and gravity items are considered as the disturbance, denoted as d_i; $M_{i,j}$ and $C_{i,j}$ are the corresponding element of the mass, Coriolis and centrifugal matrices of the dynamic model in terms of the joint states. $M_{i,i}$ is estimated as a constant inertial moment, $\bar{M}_i$, given by Eq. 6.13.

$$\bar{M}_i = \max\left(M_{i,i}(\boldsymbol{\theta})\right). \tag{6.13}$$

Define θ_{d_i} as the i^{th} desired joint angle command and θ_i is the measurement of the i^{th} joint from the potentiometer. Let $e_{\theta_i} = \theta_{d_i} - \theta_i$, then the built-in controller of the parallel robot is designed as

$$\tau_{a_i} = l_p e_{\theta_i} + l_d \dot{e}_{\theta_i} + l_i \int_0^t e_{\theta_i} dt, \tag{6.14}$$

where l_p, l_d, and l_i are the constant proportional, derivative and integrator gains.

Compared with the joint space positioning and tracking accuracy, Cartesian space pose accuracy is more appealing to the industrial robot users. Given a nominal Cartesian space trajectory of the end-effector, $\boldsymbol{\chi_d}(t)$, the desired joint space trajectory $\boldsymbol{\theta_d}(t)$ can be derived by analytically solving the inverse kinematics of the parallel robot and is served as the reference signal to the built-in controller. It is easy to be implemented and is adopted in the most industrial robots control practice. However, the Cartesian space control performance is largely compromised due to the kinematic errors introduced during the reference signal transforming. Therefore, the visual servoing controller is proposed to improve the Cartesian space tracking performance of the parallel robot.

6.4 Visual Servoing Controller Design

Considering the identified built-in controller, the Cartesian space tracking problem of the parallel robot can be stated as following: given a nominal trajectory of the pose of the end-effector, $\boldsymbol{\chi_d(t)}$, a controller is designed to make the actual pose $\boldsymbol{\chi_E(t)}$ measured by the optical CMM sensor track the nominal trajectory.

In order to design a robust visual servoing control law, the SMC methodology is combined with the Kalman filter and the position-to-torque converter to obtain a DSMC scheme, which is robust to the bounded uncertainties including identification errors, unmodeled dynamics and the sensor noise. The proposed DSMC scheme is shown in Fig. 6.2. The image of target reflectors attached on the end-effector is captured by the optical CMM sensor. The 6D pose of the end-effector can be obtained

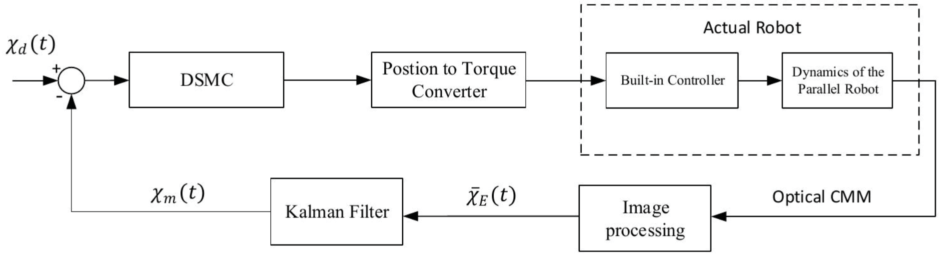


Figure 6.2: The block diagram of the DSMC scheme for the parallel robot

in real time through the visual measurement procedure given by Section 4.2. Then, the Kalman filter is employed to further eliminate the measurement noise and estimate the Cartesian space velocity of the end-effector in real time. The DSMC is designed to generate the torque vectors for actuating the parallel robot. Finally, the generated torque signal is fed to torque-to-position converter to derive the desired position commands for the built-in controller of the parallel robot.

6.4.1 Kalman Filter Design

The existence of the measurement noise of optical CMM is unavoidable and the movement of the end-effector can also introduce the vibration and blur to the image data. The Kalman filter has been adopted to handle the image data noise in [20]. The system model commonly used in Kalman filter for the pose estimation of robots is the all-integrator model [20, 112, 113]. In order to estimate the velocity of the pose, a second or higher order model is introduced in [112]. In this research, a third order integrator model is utilized in the framework of Kalman filter to smooth the pose data of the end-effector and also estimate the velocity of the pose for model-based control purpose. Assume the system modeling error $\boldsymbol{W}(t)$ and the image data noise $e_n(t)$ are subject to the white noise with Gaussian distribution. The end-effector pose measured by the optical CMM can be written as $\boldsymbol{\bar{\chi}_E(t)} = [x(t), y(t), z(t), \alpha(t), \beta(t), \gamma(t)]^T$. Then for each dimension of the pose, $\chi_{E_i}(t)$, the state vector can be represented by:

$$\boldsymbol{S_i(t)} = [\chi_{E_i}(t), \dot{\chi}_{E_i}(t), \ddot{\chi}_{E_i}(t)]^T \quad i = 1, 2, ...6. \tag{6.15}$$

Accordingly, the third order integrator Kalman filter model for each dimension of $\boldsymbol{\chi_E(t)}$ is given as:

$$\begin{aligned} \dot{\boldsymbol{S}}_{\boldsymbol{i}}(t) &= A\boldsymbol{S}_{\boldsymbol{i}}(t) + \boldsymbol{W}_{\boldsymbol{i}}(t) \\ \bar{\chi}_{E_i}(t) &= C\boldsymbol{S}_{\boldsymbol{i}}(t) + e_{n_i}(t) \end{aligned} \tag{6.16}$$

where $A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$, $\boldsymbol{W}_{\boldsymbol{i}}(t)$ and $e_{n_i}(t)$ are the white noise subjected to Gaussian distribution of $(0, Q_{c_i})$ and $(0, R_{c_i})$, $\bar{\chi}_{E_i}(t)$ is the i^{th} dimension of the pose measurement from the optical CMM. Then the continuous-time Kalman filter algorithm for the system described in Eq. 6.16 can be summarized as following steps [114]:

1. Initialization:

$$\begin{aligned} \hat{\boldsymbol{S}}_{\boldsymbol{i}}(0) &= E[\boldsymbol{S}_{\boldsymbol{i}}(0)] \\ P_i(0) &= E\left[(\boldsymbol{S}_{\boldsymbol{i}}(0) - \hat{\boldsymbol{S}}_{\boldsymbol{i}}(0))(\boldsymbol{S}_{\boldsymbol{i}}(0) - \hat{\boldsymbol{S}}_{\boldsymbol{i}}(0))^T\right] \end{aligned} \tag{6.17}$$

2. State estimation:

$$\begin{aligned} K_i(t) &= P_i(t)C^T R_{c_i}^{-1} \\ \dot{\hat{\boldsymbol{S}}}_{\boldsymbol{i}}(t) &= A\hat{\boldsymbol{S}}_{\boldsymbol{i}}(t) + K_i(t)(\bar{\chi}_{E_i}(t) - C\hat{\boldsymbol{S}}_{\boldsymbol{i}}(t)) \\ \dot{P}_i(t) &= -P_i(t)C^T R_{c_i}^{-1} C P_i(t) + AP_i(t) + P_i(t)A^T + Q_{c_i} \end{aligned} \tag{6.18}$$

where $E[*]$ represent the expected value of $*$, $P_i(t)$ is the covariance matrices of the estimation error of the states, $K_i(t)$ is the Kalman filter gain, and $\hat{}$ denotes the estimation of the value. Further, the estimation of the pose and its derivative, $\boldsymbol{\chi_m}$ and $\dot{\boldsymbol{\chi}}_{\boldsymbol{m}}$, can be obtained by rearranging $\hat{\boldsymbol{S}}(t)$.

6.4.2 DSMC for Parallel Robot

Treating the unmodeled dynamics such as backlash, sensor noise and friction as uncertainties, SMC can provide a good tacking performance robust to the uncertainties and modeling errors. The filtered pose measurement through Kalman filter is denoted as $\boldsymbol{\chi_m}(t)$.

Define the Cartesian tracking error as

$$\boldsymbol{e} = \boldsymbol{\chi_d} - \boldsymbol{\chi_m}. \tag{6.19}$$

Correspondingly, $\dot{\boldsymbol{e}}$ is as

$$\dot{\boldsymbol{e}} = \dot{\boldsymbol{\chi}}_{\boldsymbol{d}} - \dot{\boldsymbol{\chi}}_{\boldsymbol{m}}, \tag{6.20}$$

where $\boldsymbol{\chi_m}$ and $\dot{\boldsymbol{\chi}}_{\boldsymbol{m}}$, can be derived by the Kalman filter as introduced in Section 6.4.1.

Then the sliding surface, F, is selected as

$$\boldsymbol{F} = \dot{\boldsymbol{e}} + \lambda \boldsymbol{e}, \tag{6.21}$$

where $\lambda = diag[\lambda_1, \lambda_2, ..., \lambda_6]$ is a positive diagonal matrix. Since the sliding surface is chosen as Eq. 6.21, when $\boldsymbol{F} = 0$,

$$\dot{\boldsymbol{e}} = -\lambda \boldsymbol{e}, \tag{6.22}$$

The solution to ODE Eq. 6.22 is

$$\boldsymbol{e}(t) = \boldsymbol{e}(0) exp(-\lambda t), \tag{6.23}$$

Therefore, all the error states will exponentially converge to zeros when the time goes to infinity with positive definite λ. By tuning λ, the desired converging time of $\boldsymbol{F}$ can be achieved.

By defining $\dot{\boldsymbol{\chi}}_r = \boldsymbol{F} + \dot{\boldsymbol{\chi}}_m$, the following relations hold:

$$\ddot{\boldsymbol{\chi}}_r = \dot{\boldsymbol{F}} + \ddot{\boldsymbol{\chi}}_m, \; \dot{\boldsymbol{\chi}}_r = \dot{\boldsymbol{\chi}}_d + \lambda \boldsymbol{e}, \; \ddot{\boldsymbol{\chi}}_r = \ddot{\boldsymbol{\chi}}_d + \lambda \dot{\boldsymbol{e}}. \tag{6.24}$$

The designed DSMC is given as:

$$\boldsymbol{\tau_a} = J_{ad}^{-T} J_s^{-T} (\boldsymbol{\tau}_{\mathrm{m}} + K_p \boldsymbol{F} + K_i \int_0^t \boldsymbol{F} \mathrm{d}t + \boldsymbol{\tau}_{\mathrm{r}}), \tag{6.25}$$

where

$$\boldsymbol{\tau}_m = \boldsymbol{M}_0(\boldsymbol{\chi}_m)\ddot{\boldsymbol{\chi}}_r + \boldsymbol{C}_0(\boldsymbol{\chi}_m, \dot{\boldsymbol{\chi}}_m)\dot{\boldsymbol{\chi}}_r + \boldsymbol{G}_0(\boldsymbol{\chi}_m) + \boldsymbol{\tau_{f0}}, \tag{6.26}$$

$$\boldsymbol{\tau}_r = K_r \ \mathrm{sat}(\boldsymbol{F}), \tag{6.27}$$

$$\mathrm{sat}(\boldsymbol{F}) = [\mathrm{sat}(F_1), \mathrm{sat}(F_2), ..., \mathrm{sat}(F_6)]^T,$$
$$\mathrm{sat}(F_i) = \begin{cases} F_i & \text{if } |F_i| \le \epsilon \\ \mathrm{sgn}(F_i) & \text{otherwise} \end{cases}. \tag{6.28}$$

And the dynamic model items calculated by the identified dynamic parameters and measurements are denoted as $M_0(\boldsymbol{\chi_m})$, $C_0(\boldsymbol{\chi_m}, \dot{\boldsymbol{\chi}}_{\boldsymbol{m}})$, $\boldsymbol{G_0}(\boldsymbol{\chi_m})$ and $\boldsymbol{\tau_{f0}}$ respectively.

To add more freedom to the controller, the propotional and intergration items are adopted in the controller Eq. 6.25. The gain matrix are K_p and K_i respectively. The sat function is employed to reduce the chattering phenomenon. ϵ is a small positive constant, by selecting ϵ, a compromise between accuracy and chattering can be made.

The Lyapunov direct method is employed to analyze the stability of the proposed controller.

Theorem 1. *For the robotic system given in Eq. 6.1 with the CMM visual sensor in eye-to-hand setup, the ultimate bounded Cartesian space tracking errors can be achieved with the dynamic SMC controller Eq. 6.25.*

Proof 3. *By choosing Lyapunov function as*

$$V = \frac{1}{2}\boldsymbol{F}^{\mathrm{T}}\widetilde{M}(\boldsymbol{\chi}_m)\boldsymbol{F} + \frac{1}{2}\left(\int_0^t \boldsymbol{F}dt\right)^{\mathrm{T}} \boldsymbol{K}_i \left(\int_0^t \boldsymbol{F}dt\right), \tag{6.29}$$

the time derivative of Lyapunov function is given by

$$\dot{V} = \boldsymbol{F}^{\mathrm{T}}\left[\widetilde{M}(\boldsymbol{\chi}_m)\dot{\boldsymbol{F}} + \frac{1}{2}\dot{\widetilde{M}}(\boldsymbol{\chi}_m)\boldsymbol{F} + K_i\int_0^t \boldsymbol{F}dt\right]. \tag{6.30}$$

According to Eq. 6.7, Eq. 6.30 can be rewritten as

$$\dot{V} = \boldsymbol{F}^{\mathrm{T}}\left[\widetilde{M}(\boldsymbol{\chi}_m)\dot{\boldsymbol{F}} + \widetilde{C}(\boldsymbol{\chi}_m, \dot{\boldsymbol{\chi}}_m)\boldsymbol{F} + K_i\int_0^t \boldsymbol{F}dt\right] \tag{6.31}$$

Substituting Eq. 6.24 into Eq. 6.1, the following equation is derived.

$$\begin{aligned}
\widetilde{\boldsymbol{\tau}}_{\boldsymbol{g}} &= \widetilde{M}(\boldsymbol{\chi}_m)\ddot{\boldsymbol{\chi}}_m + \widetilde{C}(\boldsymbol{\chi}_m, \dot{\boldsymbol{\chi}}_m)\dot{\boldsymbol{\chi}}_m + \widetilde{G}(\boldsymbol{\chi}_m) + \widetilde{\boldsymbol{\tau}}_{\boldsymbol{f}} + \boldsymbol{\tau}_{\boldsymbol{u}} \\
&= \widetilde{M}(\boldsymbol{\chi}_m)\left(\ddot{\boldsymbol{\chi}}_r - \dot{\boldsymbol{F}}\right) + \widetilde{C}(\boldsymbol{\chi}_m, \dot{\boldsymbol{\chi}}_m)\left(\dot{\boldsymbol{\chi}}_r - \boldsymbol{F}\right) + \widetilde{G}(\boldsymbol{\chi}_m) \\
&\quad + \widetilde{\boldsymbol{\tau}}_{\boldsymbol{f}} + \boldsymbol{\tau}_{\boldsymbol{u}} \\
&= \widetilde{M}(\boldsymbol{\chi}_m)\ddot{\boldsymbol{\chi}}_r + \widetilde{C}(\boldsymbol{\chi}_m, \dot{\boldsymbol{\chi}}_m)\dot{\boldsymbol{\chi}}_r + \widetilde{G}(\boldsymbol{\chi}_m) + \widetilde{\boldsymbol{\tau}}_{\boldsymbol{f}} + \boldsymbol{\tau}_{\boldsymbol{u}} \\
&\quad - \widetilde{M}(\boldsymbol{\chi}_m)\dot{\boldsymbol{F}} - \widetilde{C}(\boldsymbol{\chi}_m, \dot{\boldsymbol{\chi}}_m)\boldsymbol{F} \\
&= M_0(\boldsymbol{\chi}_m)\ddot{\boldsymbol{\chi}}_r + C_0(\boldsymbol{\chi}_m, \dot{\boldsymbol{\chi}}_m)\dot{\boldsymbol{\chi}}_r + G_0(\boldsymbol{\chi}_m) + \boldsymbol{\tau}_{\boldsymbol{f0}} + \boldsymbol{H} \\
&\quad - \widetilde{M}(\boldsymbol{\chi}_m)\dot{\boldsymbol{F}} - \widetilde{C}(\boldsymbol{\chi}_m, \dot{\boldsymbol{\chi}}_m)\boldsymbol{F},
\end{aligned} \tag{6.32}$$

where $\boldsymbol{H} = \mathit{\Delta} M\ddot{\boldsymbol{\chi}}_r + \mathit{\Delta} C\dot{\boldsymbol{\chi}}_r + \boldsymbol{\mathit{\Delta} G} + \mathit{\Delta}\boldsymbol{\tau}_{\boldsymbol{f}} + \boldsymbol{\tau}_u$; $\boldsymbol{\tau}_u$ *represents the unmodeled dynamics, and* $\mathit{\Delta} M, \mathit{\Delta} C, \boldsymbol{\mathit{\Delta} G}, \mathit{\Delta}\boldsymbol{\tau}_{\boldsymbol{f}}$ *are the modeling errors given by*

$$\begin{aligned}
&\mathit{\Delta} M = \widetilde{M}(\boldsymbol{\chi}_{\boldsymbol{m}}) - M_0(\boldsymbol{\chi}_{\boldsymbol{m}}), \\
&\mathit{\Delta} C = \widetilde{C}(\boldsymbol{\chi}_{\boldsymbol{m}}, \dot{\boldsymbol{\chi}}_m) - C_0(\boldsymbol{\chi}_{\boldsymbol{m}}, \dot{\boldsymbol{\chi}}_m), \\
&\boldsymbol{\mathit{\Delta} G} = \widetilde{\boldsymbol{G}}(\boldsymbol{\chi}_{\boldsymbol{m}}) - \boldsymbol{G}_{\boldsymbol{0}}(\boldsymbol{\chi}_{\boldsymbol{m}}), \\
&\mathit{\Delta}\boldsymbol{\tau}_{\boldsymbol{f}} = \widetilde{\boldsymbol{\tau}}_{\boldsymbol{f}} - \boldsymbol{\tau}_{\boldsymbol{f0}}.
\end{aligned} \tag{6.33}$$

Suppose the kinematic calibration and dynamical identification can guarantee that $\boldsymbol{H}$ *satisfies:*

$$||\boldsymbol{H}|| \leq \hat{\boldsymbol{H}}. \tag{6.34}$$

Further substituting Eq. 6.25 and Eq. 6.32 into Eq. 6.31, the positive definite time derivative of Lyapunov function can be derived as

$$\begin{aligned}
\dot{V} &= -\boldsymbol{F}^{\mathrm{T}} K_{\mathrm{p}} \boldsymbol{F} - \boldsymbol{F}^{\mathrm{T}} K_{\mathrm{r}} \operatorname{sat}(\boldsymbol{F}) + \boldsymbol{F}^{\mathrm{T}} \boldsymbol{H} \\
&= -\boldsymbol{F}^{\mathrm{T}} K_{\mathrm{p}} \boldsymbol{F} - \sum_{i=1}^{n} K_{\mathrm{ri}} |F|_i + \boldsymbol{F}^{\mathrm{T}} \boldsymbol{H}, \quad \textit{if } |F_i| > \epsilon
\end{aligned} \tag{6.35}$$

where $K_r = diag[K_{r1}, K_{r2}, ..., K_{r6}]$ *is a constant matrix. By selecting* $K_{ri} > \hat{H}_i$*, then one has*

$$\dot{V} < -\boldsymbol{F}^{\mathrm{T}} K_{\mathrm{p}} \boldsymbol{F} < 0 \tag{6.36}$$

Therefore, when $|F_i|$ *is in the case* $|F_i| > \epsilon$*,* $\boldsymbol{F}$ *will keep decreasing until the boundary layer* $|F_i| \leq \epsilon$ *is reached in a finite time. Once F reaches the boundary layer,* $\boldsymbol{F}$ *will remain inside it, and thus the uniformly ultimately bounded stability with an ultimate bound can be guaranteed [115].*
Q.E.D.

6.4.3 Torque to Position Converter

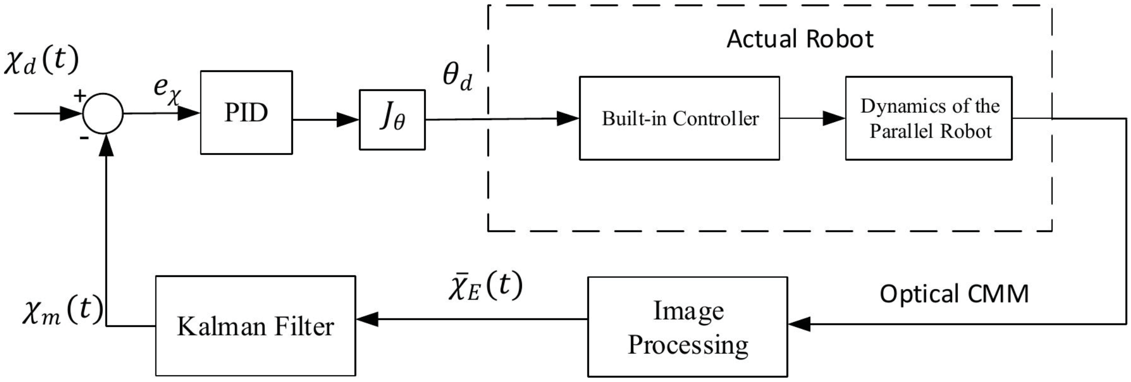
The purpose of the torque to position converter is to generate a position profile command for the built-in controller of the parallel robot based on the torque signal of the designed SMC controller. The torque signal produced by the built-in controller can be equivalent to that produced by the DSMC controller. The converter is similar to the method proposed in [79], in which P-type controller is considered as the built-in controller of the robot. As the PID controller is employed in the built-in controller of the parallel robot, the modifications are made based on the results given in [79]. According to Eq. 6.14, the position command θ_{d_i} can be derived as followings

$$\tau_{a_i} = (l_p + l_d \boldsymbol{p} + l_i \boldsymbol{p}^{-1}) e_{\theta_i}, \tag{6.37}$$

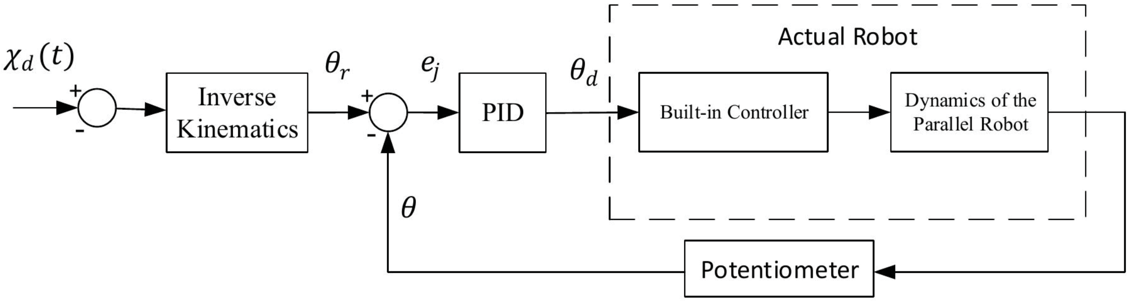
$$\theta_{d_i} = \frac{1}{l_p + l_d \boldsymbol{p} + l_i \boldsymbol{p}^{-1}} \tau_{a_i} + \theta_i, \tag{6.38}$$

where $\boldsymbol{p}$ is the differential operator. τ_{a_i} is derived by Eq. 6.25. The joint position feedback θ_i is measured by the potentiometer.

In the proposed controller, the identified dynamic model can guarantee the boundedness of the uncertainties. Since the dynamic model has been used for visuals servoing control design, a better tracking performance in terms of robust, convergence speed and accuracy can be achieved compared with kinematic level visual servoing controller.



(a)



(b)

Figure 6.3: The block diagram of kinematic level controller (a) KCSC, (b) KJSC

6.5 Experiment Results

The proposed DSMC scheme is tested on the 6-RSS parallel robot, as shown in Fig. 4.6a, and the experiment setup is shown in Fig. 4.6b. In this section, two path tracking tests are performed to investigate the effectiveness of the designed controller and compare the control performance of the proposed DSMC scheme with the kinematic-level Cartesian space controller (KCSC) and joint space controller (KJSC).

The KCSC is shown in Fig. 6.3a. By Defining the Cartesian space trajectory error as

$$e_\chi = \chi_d - \chi_m, \tag{6.39}$$

Table 6.1: Experiment 1- the RMSE and MAE levels of the three control schemes

Pose	Characteristics	KJSC	KCSC	DSMC
x direction	RMSE (mm)	2.8	1.1	0.44
	MAE (mm)	2.5	0.97	0.36
y direction	RMSE (mm)	3.4	1.5	1.4
	MAE (mm)	2.7	1.1	0.61
z direction	RMSE (mm)	2.1	1.2	0.63
	MAE (mm)	1.8	1.0	0.46
α rotation	RMSE $(10^{-3}rad)$	14.8	3.2	1.1
	MAE $(10^{-3}rad)$	12.5	2.8	0.92
β rotation	RMSE $(10^{-3}rad)$	14.0	3.8	2.1
	MAE $(10^{-3}rad)$	12.4	3.3	1.7
γ rotation	RMSE $(10^{-3}rad)$	14.6	3.5	1.6
	MAE $(10^{-3}rad)$	14.2	3.1	1.3

the controller is designed as:

$$\boldsymbol{\theta}_d = J_\theta(k_p\boldsymbol{e}_\chi + k_d\dot{\boldsymbol{e}}_\chi + k_i\int_0^t \boldsymbol{e}_\chi dt), \tag{6.40}$$

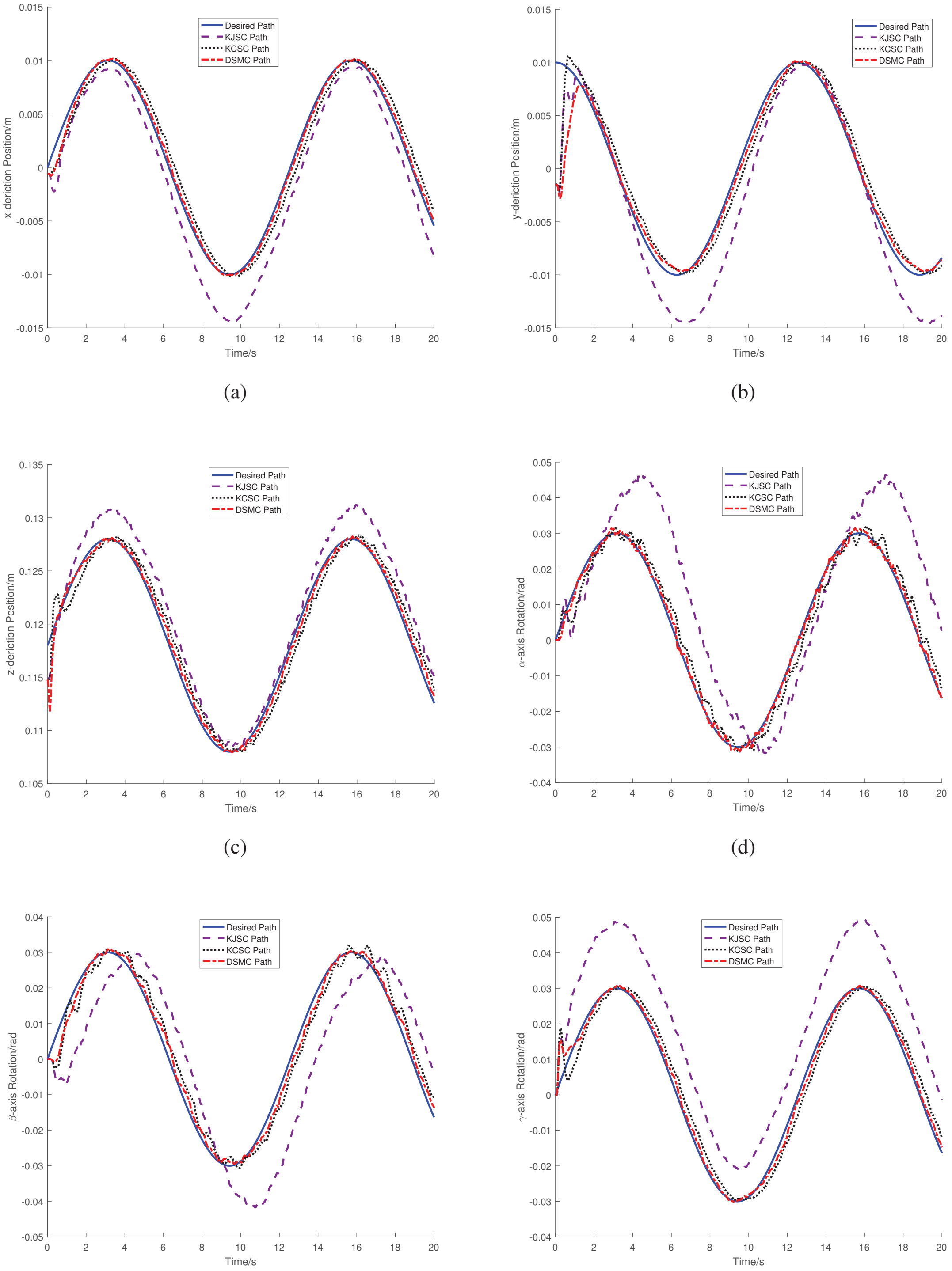
where J_θ is the analytical Jacobian matrix given by Eq. 5.3. In Eq. 6.40, k_p, k_d, and k_i are the constant positive definite diagonal matrices.

The KJSC is shown in Fig. 6.3b. The desired trajectory $\boldsymbol{\chi_d}(t)$ in the Cartesian space is transformed into the one in the joint space ($\boldsymbol{\theta}_r$) through analytically solving the inverse kinematic model of the parallel robot. The joint space trajectory error is defined as:

$$\boldsymbol{e}_j = \boldsymbol{\theta}_r - \boldsymbol{\theta}, \tag{6.41}$$

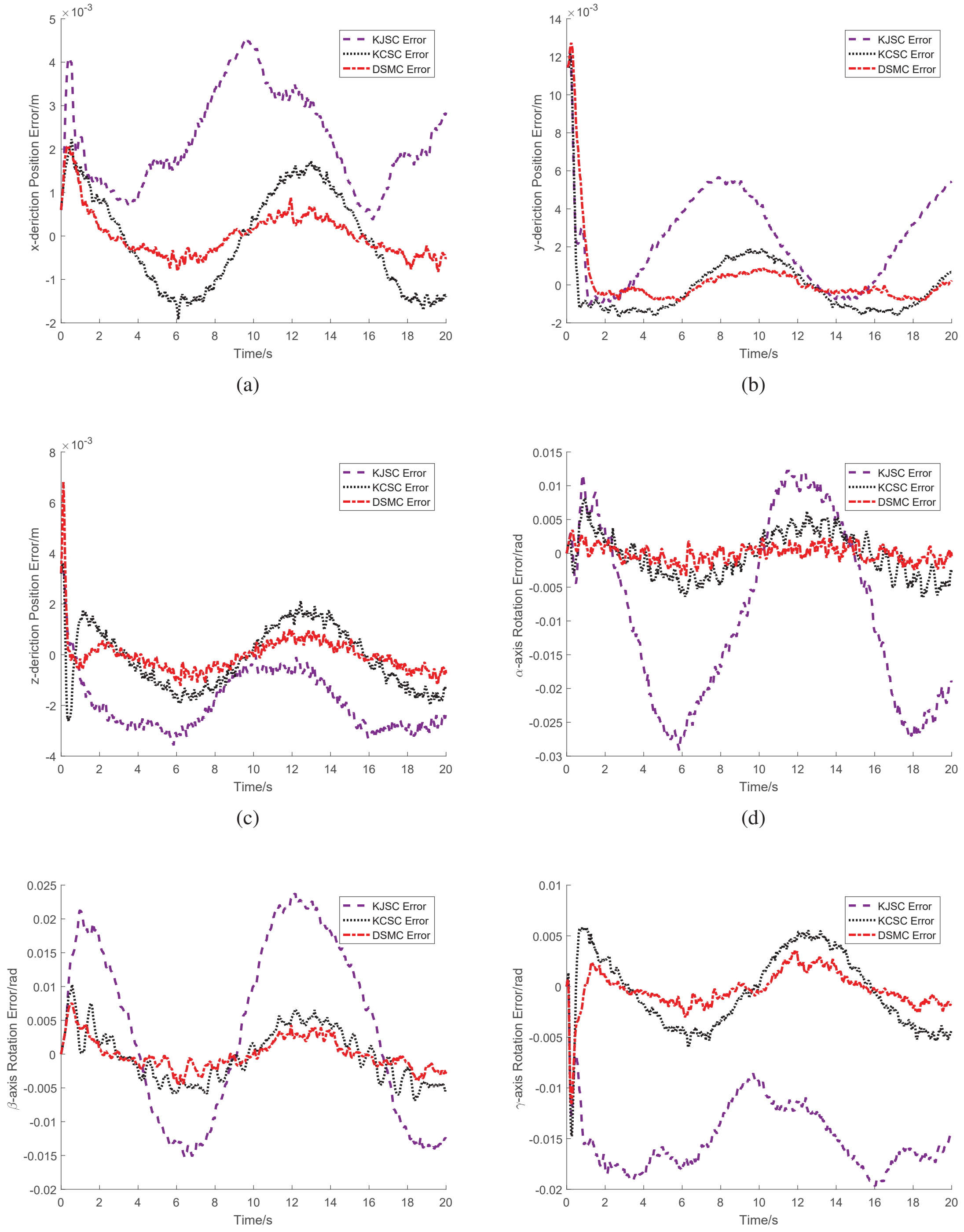
and the KJSC is designed as:

$$\boldsymbol{\theta}_d = k_p\boldsymbol{e}_j + k_d\dot{\boldsymbol{e}}_j + k_i\int_0^t \boldsymbol{e}_j dt. \tag{6.42}$$



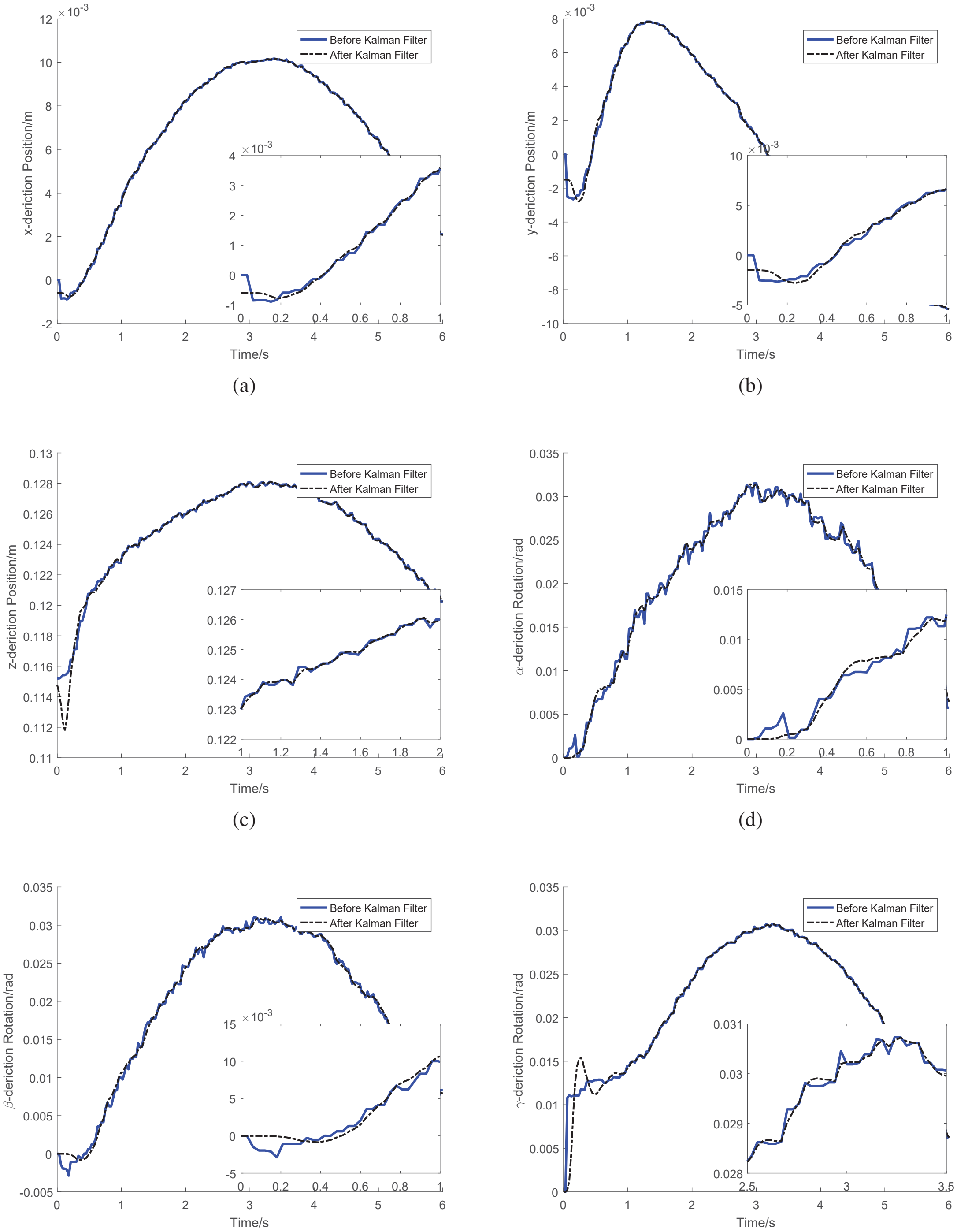
(a) (b) (c) (d) (e) (f)

Figure 6.4: Experiment 1-the tracking performances of the three control schemes tested on the 6-RSS parallel robot: the desired trajectories (blue solid line), the pose of KJSC scheme (purple dashed line), the pose of KCSC scheme (black dotted line), the pose of DSMC scheme (red dash-dotted line), (a) Along X Direction; (b) Along Y Direction; (c) Along Z Direction; (d) Around α Axis; (e) Around β Axis; (f) Around γ Axis



(a) (b) (c) (d) (e) (f)

Figure 6.5: Experiment 1-the tracking errors of the three control schemes tested on the 6-RSS parallel robot: KJSC scheme (purple dashed line), KCSC scheme (black dotted line), DSMC scheme (red dash-dotted line), (a) Along X Direction; (b) Along Y Direction; (c) Along Z Direction; (d) Around α Axis; (e) Around β Axis; (f) Around γ Axis



(a) (b)

(c) (d)

(e) (f)

Figure 6.6: Experiment 1- the Kalman filter results: the measurement from optical CMM before Kalman filter (blue solid line), the output of the Kalman filter (black dash-dotted line), (a) Along X Direction; (b) Along Y Direction; (c) Along Z Direction; (d) Around α Axis; (e) Around β Axis; (f) Around γ Axis

6.5.1 Experiment 1

In Experiment 1, the pose of the end-effector frame is expected to track the sinusoid reference Cartesian space trajectories with the frequency of $0.5rad/s$. Note that the reference trajectories are given in the kinematic and Euler singularity free workspace of the parallel robot. The details of the singularity free workspace analysis for the 6-RSS parallel robot can be found in Section 4.4.1. The dynamic parameters of the parallel robot and the gains of the built-in controller are based on the identification results. The initial pose of the $\boldsymbol{\chi} = [0(m), 0(m), 0.115(m), 0(rad), 0(rad), 0(rad)]$ and the initial velocity and acceleration are zeros. The standard deviation of the measurement noise, R_c, can be derived in the numerous measurement tests of the optical CMM. A satisfactory Kalman filter result can be derived by tuning the standard deviation of the modeling error Q_c. The control rate is set the same as the sampling rate of the optical CMM, $0.033s$. Both positional and rotational tracking performances of the three controllers are shown in Fig. 6.4, and the tracking errors between the reference trajectories and the measurement can be seen in Fig. 6.5. In the figures, the reference trajectories, the measurement pose trajectories, and the tracking errors of the three controllers are given in the individual dimension of the pose. Note that there are step changes in the y and z direction reference path at the initial status to test the effectiveness of the controllers. And the root-mean-square error (RMSE), and mean absolute error (MAE) are given in Table 6.1 to compare the tracking performance of the three controllers.

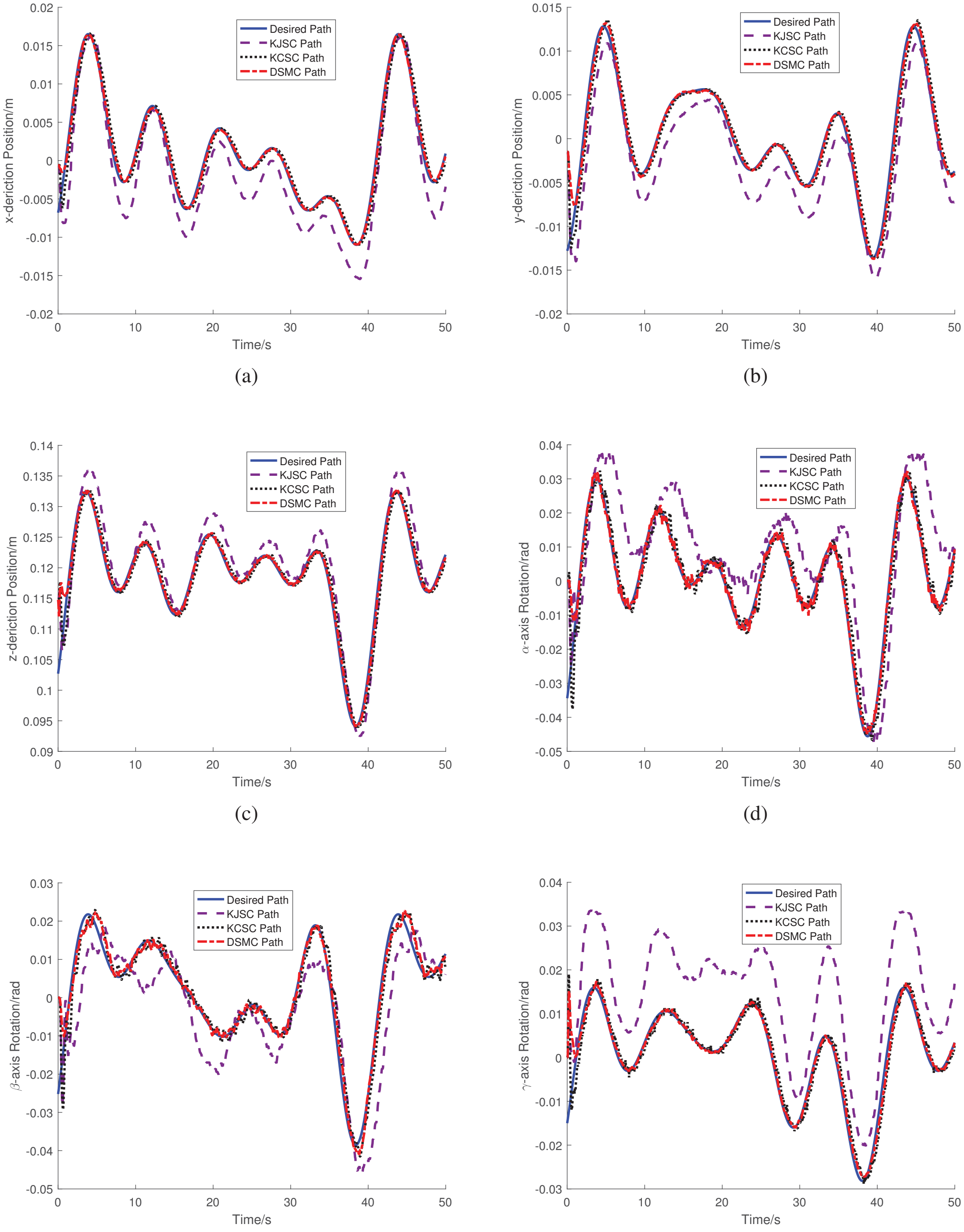
In Figures 6.4 and 6.5, it is evident that the joint space controller (KJSC) has much larger errors compared with the Cartesian space controllers (KCSC and DSMC), due to the kinematic errors during the transformation of the desired trajectories. The Cartesian space controllers can effectively track the sinusoid 6D trajectories with small steady states errors and rapid converge speed, and are robust to the sensor noises and modeling errors. It can be seen that the rotational tracking performance is more oscillatory than the positional tracking performance especially in α and β rotations. This phenomenon may be caused by the fact that the orientation measurement tends to be nosier than the position measurement from the optical CMM sensor. This can also be seen in the Kalman filter results in Fig. 6.6. For brevity, only the positional and rotational results of the Kalman filter in DSMC scheme are given. It is also worthwhile mentioning that a compromise should be made between the converge speed and the noise filtering effect when tuning the Kalman filter. With the dynamic compensation in the DSMC scheme, the steady state errors in the DSMC scheme is smaller than those in compared with the KCSC scheme. Although, in some directions and rotations, the KCSC owns that in a faster converge speed, but with larger overshoot compared with the DSMC scheme. Furthermore, in terms of the RMSE and MAE, it is obvious that the DSMC scheme achieves better tracking performance in all dimensions. In conclusion, the first experiment can validate the effectiveness and the robustness of the designed DSMC scheme when tracking the constant frequency sinusoid reference trajectories. And the superiority of the DSMC scheme to KJSC and KCSC is further illustrated in Table 6.1 on the RMSE and MAE results.

6.5.2 Experiment 2

To further test the effectiveness and the robustness of the proposed DSMC scheme under varying frequency harmonic reference trajectories, the Finite Fourier series-based trajectories shown in Eq. 6.43 are used as the references.

$$\chi_{d_i}(t) = \sum_{l=1}^{n}[\frac{sin(\omega_0 lt)}{\omega_0 l} s_i^l - \frac{cos(\omega_0 lt)}{\omega_0 l} c_i^l] + \chi_{0_i} \tag{6.43}$$

where $\chi_{d_i}(t)$ is the i-th column of the pose trajectory, n is the harmonics number, ω_0 is the fundamental frequency, and s_i^l, c_i^l, χ_{0_i} are the trajectory parameters. In the second experiment, n is chosen as 5 and ω_0 is set as $0.15rad/s$. Therefore, the reference trajectories combines 5 different frequencies varying from $0.15rad/s$ to $0.75rad/s$. Other parameters are chosen to generate singularity free reference trajectories. The step changes are given to all dimensions of the reference pose path at the beginning. To test the robustness of the control schemes, in Experiment 2, all the settings and parameters for the control schemes and the Kalman filter are set as the same as Experiment 1. The tracking performances of the control schemes are shown in Fig. 6.7 and 6.8. The reference trajectories, the measurement pose trajectories and the errors of the three controllers are given. The Kalman filter results working with the DSMC scheme are given in Fig. 6.9. The RMSE and MAE of the control schemes are summarized in Table 6.2. From Fig. 6.7, it is illustrated that compared with the joint space controller, both Cartesian space control schemes have satisfactory tracking errors

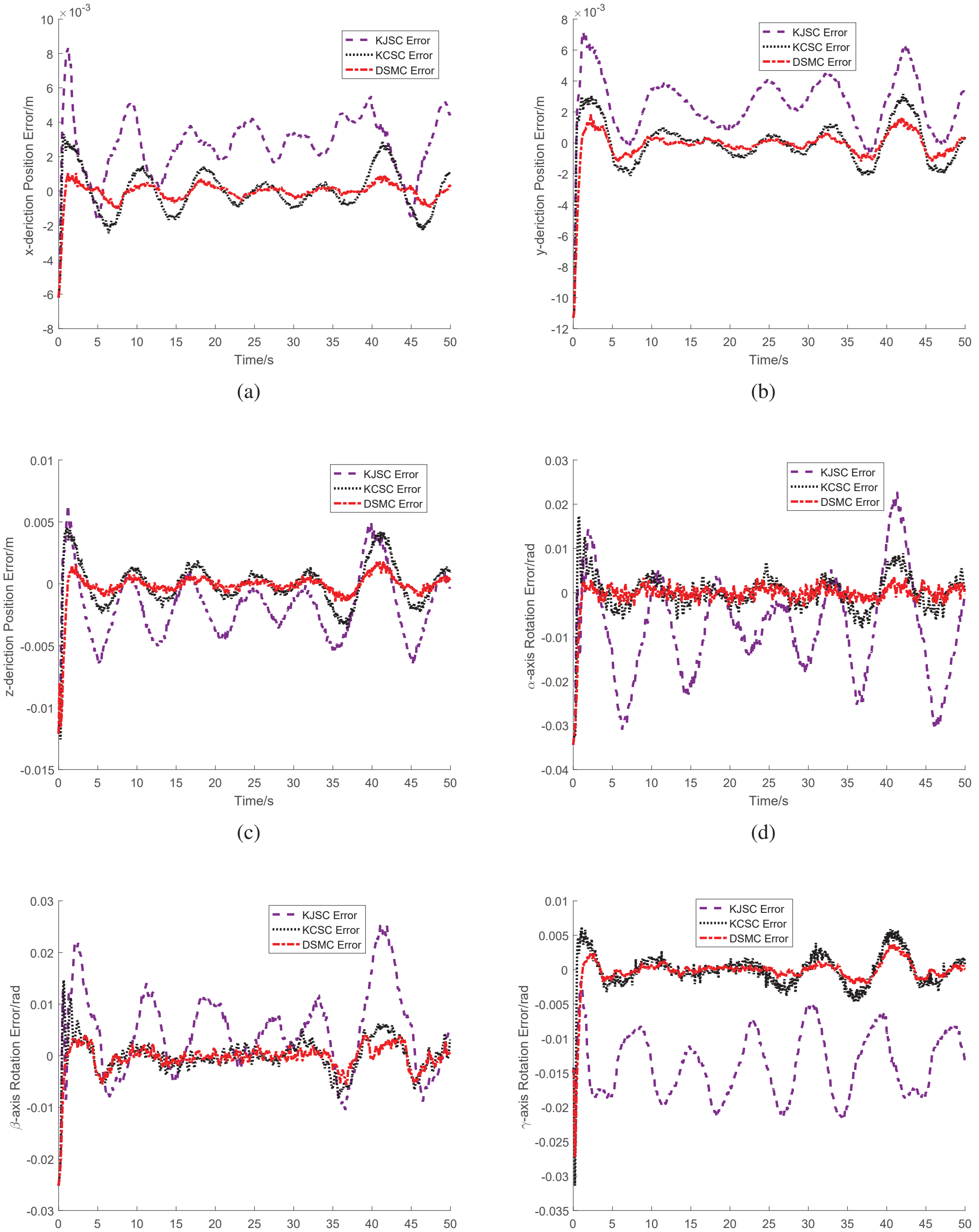


(a) (b)

(c) (d)

(e) (f)

Figure 6.7: Experiment 2-the tracking performances of the three control schemes tested on the 6-RSS parallel robot: the desired trajectories (blue solid line), the pose of KJSC scheme (purple dashed line), the pose of KCSC scheme (black dotted line), the pose of DSMC scheme (red dash-dotted line), (a) Along X Direction; (b) Along Y Direction; (c) Along Z Direction; (d) Around α Axis; (e) Around β Axis; (f) Around γ Axis

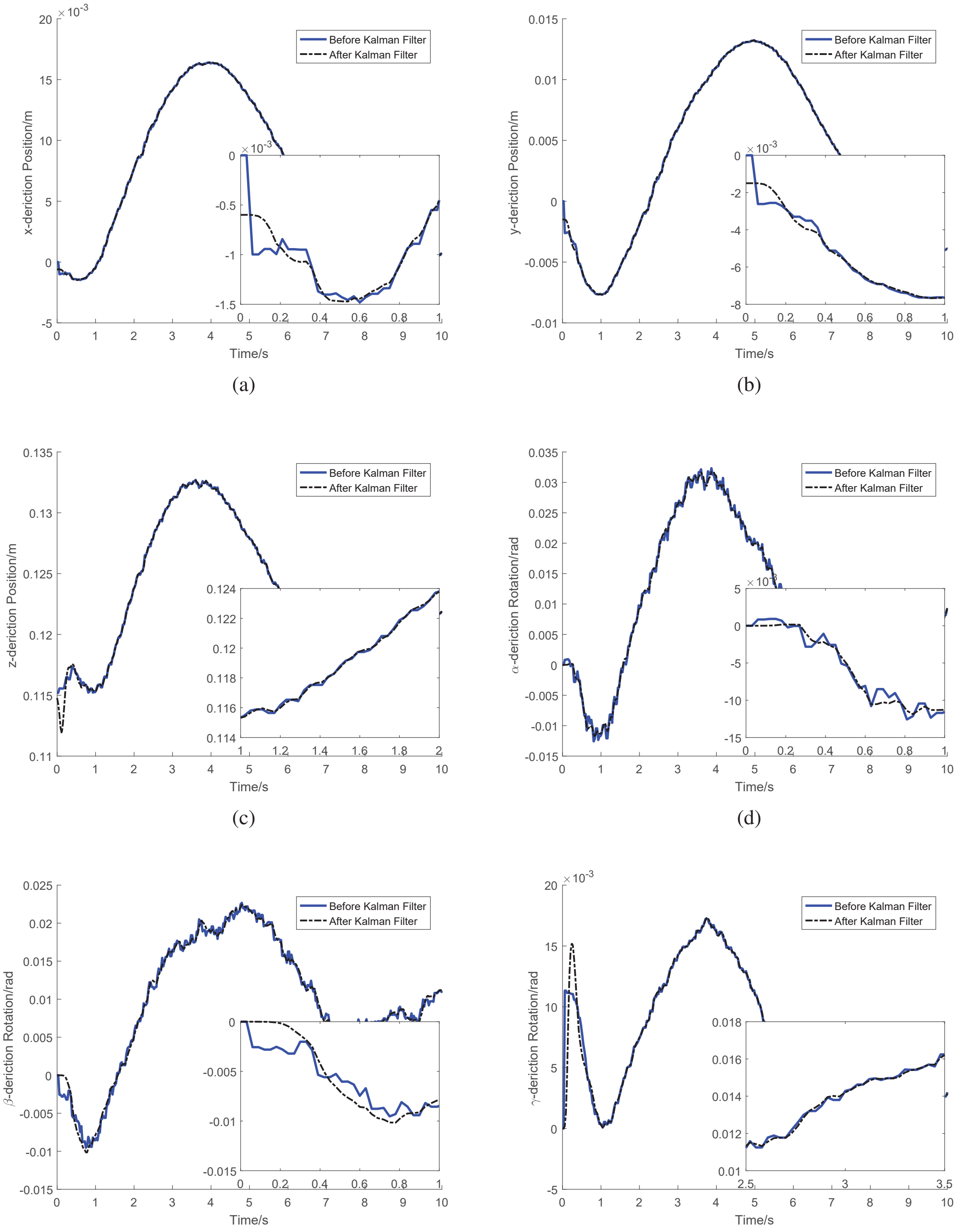


(a) (b)

(c) (d)

(e) (f)

Figure 6.8: Experiment 2-the tracking errors of the three control schemes tested on the 6-RSS parallel robot: KJSC scheme (purple dashed line), KCSC scheme (black dotted line), DSMC scheme (red dash-dotted line), (a) Along X Direction; (b) Along Y Direction; (c) Along Z Direction; (d) Around α Axis; (e) Around β Axis; (f) Around γ Axis



(a) (b) (c) (d) (e) (f)

Figure 6.9: Experiment 2- the Kalman filter results: the measurement from optical CMM before Kalman filter (blue solid line), the output of the Kalman filter (black dash-dotted line), (a) Along X Direction; (b) Along Y Direction; (c) Along Z Direction; (d) Around α Axis; (e) Around β Axis; (f) Around γ Axis

Table 6.2: Experiment 2- the RMSE and MAE levels of the three control schemes

Pose	Characteristics	KJSC	KCSC	DSMC
x direction	RMSE (mm)	3.2	1.2	0.60
	MAE (mm)	2.9	0.96	0.33
y direction	RMSE (mm)	3.2	1.5	1.1
	MAE (mm)	2.7	1.0	0.54
z direction	RMSE (mm)	3.1	1.9	1.3
	MAE (mm)	2.6	1.3	0.57
α rotation	RMSE $(10^{-3}rad)$	13.9	4.5	3.4
	MAE $(10^{-3}rad)$	11.3	3.0	1.4
β rotation	RMSE $(10^{-3}rad)$	9.4	3.6	2.8
	MAE $(10^{-3}rad)$	7.4	2.4	1.6
γ rotation	RMSE $(10^{-3}rad)$	14.5	2.9	2.5
	MAE $(10^{-3}rad)$	13.7	1.8	1.0

and rapid converge speed, and can track the harmonic references with varying frequencies. Similar to Experiment 1, more oscillations are observed in the rotational tracking than that in the positional tracking due to the noisier rotation measurement of the optical CMM. Compared with Experiment 1, the tracking responds in Experiment 2 have the similar properties except that the tracking errors of the control schemes are increased at the high frequency part of the reference trajectories in Fig. 6.7 and 6.8, especially in x, y, z and α directions. The main reason is that the influences of the dynamics of the parallel robot and other uncertainties such as the friction are increased substantially with the increase of the velocity. However, due to the dynamic compensation and robustness of the DSMC scheme, the tracking errors in DSMC scheme are significantly less than those in the KJSC and KCSC schemes. Furthermore, from Table 6.2, it is evident that the DSMC scheme achieves better tracking performance than KJSC and KCSC do.

In the second experiment, the effectiveness and the robustness of the proposed DSMC scheme are further validated by giving varying and higher frequency harmonic reference trajectories. The DSMC scheme is more robust to the higher velocity situation compared with KJSC and KCSC scheme. And the superiority of the DSMC scheme to KJSC and KCSC scheme is further illustrated by the better RMSE and MAE results shown in Table 6.2.

6.6 Summary

In this chapter, a dynamic sliding mode control (DSMC) scheme combined with the visual servoing method is proposed to improve the tracking performance of the 6-RSS parallel robot based on the optical coordinate measuring machine (CMM) sensor. The torque command generated by DSMC can be applied to the position controlled industrial robot through a position-to-torque converter. A third-order Kalman filter is utilized to eliminate the noise in the measurement data from C-track. The real-time experiment tests on a 6-RSS parallel robot demonstrated that the developed DSMC scheme is robust to the modeling errors and uncertainties. Compared with the classical kinematic level joint space and Cartesian space controllers, the proposed DSMC exhibits the superiority in terms of tracking performance and robustness.

Chapter 7

Conclusion and Future Works

7.1 Summary of the Thesis

The parallel robots exhibit some outstanding superiorities on the repeatability, stiffness and force-to-weight ratio compared with the serial robots. As a result, parallel robots, especially 6-DOF parallel robots have been applied to various industrial applications such as flight simulator, and have great potential to be utilized in the AFP system. However, the absolute accuracy of the parallel robots cannot meet the growing requirement from industry, due to the existence of numerous legs and joints. How to design effective algorithms and controllers to improve the positioning and tracking accuracy of the parallel robots poses a challenge to the control community. Researchers have developed various methods to calibrate the kinematic parameters, identify the dynamic parameters and design the controllers to achieve a better positioning and tracking accuracy of the parallel robots. In this Ph.D. project, the existing problems and gaps of the current research are analyzed. By introducing the optical CMM sensor, a series of vision-based solutions including visual calibration, identification and servoing algorithms for the 6-RSS parallel robot are proposed.

(1) **Kinematic analysis and dynamic modeling of the 6-RSS parallel robot**

The inverse kinematic problem of the 6-RSS parallel robot is solved analytically. A quasi-Stewart forward kinematic method is developed to solve the forward kinematic problem of the 6-RSS parallel robot numerically. The superiority of the method in terms of the convergence speed is demonstrated by the simulation, comparing with the normal forward kinematic method. The derived kinematic solutions will be used to determine the proper workspace and to develop the kinematic calibration method. The Jacobian matrices reveling the velocity mappings from the joint space, wrench and link frame velocity to the workspace velocity of the 6-RSS parallel robot are developed for building the dynamic model. The explicit form of the dynamic model of the 6-RSS parallel robot is obtained based on the virtual work principle. The built dynamic model will be utilized to design the identification method and advanced dynamic model-based controller for the 6-RSS parallel robot.

(2) **Relative pose-based kinematic calibration method for the 6-RSS parallel robot using optical CMM sensor**

A relative pose-based kinematic calibration method is proposed for a 6-RSS parallel robot by using the optical CMM system. The developed calibration algorithm can improve the positioning accuracy according to more accurate kinematic parameters. In this algorithm, the base frame pose w.r.t. the sensor frame is not needed, which leads to an effective relative calibration method for the parallel robot. Since the forward kinematic model is used in the relative pose based calibration, the optimal actuator strokes of the parallel robot are derived to ensure the homeomorphism mapping of the forward kinematic model. The simulation results show the relative pose based calibration algorithm successfully improves the relative accuracy of the parallel robot. The comparison with the implicit calibration demonstrates that the RPBA can deliver a more satisfactory accuracy. The experimental tests on an arbitrary trajectory with 100 configurations further show the proposed RPBA has improved the positional accuracy in the workspace effectively. The developed calibration algorithm can be applied to other types of parallel and serial robots. And the calibrated kinematic parameters would be used in a 6-RSS parallel robot visual servoing system.

(3) **Visual closed-loop output-error identification method based on an optical CMM sensor for parallel robot**

A visual closed-loop output-error identification method based on an optical CMM sensor for parallel robots is proposed. An outer loop visual servoing controller is employed in both the real plant and the simulation model to stabilize the two systems. The benefits of the proposed method are summarized as follows: elimination of the need for the joint and torque measurements, the exact knowledge of the built-in joint controller of the industrial robots, and the time-consuming forward kinematics calculation. The accuracy of the built dynamic model are validated by the Matlab/SimMechanics simulation. The experimental test results show that the identified dynamic model can capture the dynamics of the real parallel robot with satisfactory accuracy. The proposed method can be easily applied to other types of industrial parallel robots with unknown PID built-in controller or its variants, such as 6 DOF Stewart platforms, 6 UPS and 6 RUS parallel robots etc. The complexity of those dynamic models is similar to that of the 6-RSS parallel robot. Since the analytical solution of the forward kinematics of those 6 DOF parallel robots does not exist, the proposed visual identification method does not need the forward kinematic model. Taking the advantages of the visual sensor, the dynamic model can be identified for the visual servoing purpose.

(4) **Dynamic model based visual servoing method for the 6-RSS parallel robot**

A dynamic sliding mode control (DSMC) scheme combined with the visual servoing method is proposed to improve the tracking performance of the 6-RSS parallel robot based on the optical coordinate measuring machine (CMM) sensor. The torque command generated by DSMC can be applied to the position controlled industrial robot through a position-to-torque converter. A third-order Kalman filter is utilized to eliminate the noise in the measurement data from C-track. The real-time experiment tests on a 6-RSS parallel robot demonstrated that the developed DSMC scheme is robust to the modeling errors and uncertainties. Compared with the classical kinematic level joint space and Cartesian space controllers, the proposed DSMC exhibits the superiority in terms of tracking performance and robustness.

7.2 Future Works

The positioning and tracking accuracy of the 6-RSS parallel robot has been improved by developed visual calibration, identification and servoing algorithms in this project. There are some recommended research work needed to be conducted, which are listed as followings.

(1) The proposed visual calibration, identification and servoing methods has great potentials to be applied to other type of parallel robots. Some experiment tests and research work are needed to be carried out to make the algorithms universal.

(2) The backlash and time-delay nonlinearities are taken as uncertainties in the proposed visual servoing algorithm. More efficient control methods dealing with the nonlinearities are needed to be studied. For example, by considering the flexible body dynamic model of the parallel robots and adding encoders on the revolute joints, combine the singularity perturbed theory and visual servoing to develop more effective controllers.

(3) Since the dynamic modeling and identification process is very complicated for the parallel robots, some black box model-based methods like neural networks model can be developed to implement on-line identification and control for the parallel robot.

(4) Another potential research field is designing more effective filters which have faster converge speed and higher filtering ability.

(5) In the AFP system, the parallel robot will collaborate with the serial robot. Hence to improve the final manufacturing results, the cooperative calibration and control methods are needed to improve the relative pose accuracy between the parallel robot and the serial robot instead of emphasizing the individual accuracy of the robots.

Bibliography

[1] Jian Wang and Oren Masory. On the accuracy of a stewart platform. i. the effect of manufacturing tolerances. In *Proceedings of 1993 IEEE International Conference on Robotics and Automation*, pages 114–120, Atlanta, USA, May 1993. IEEE.

[2] Yan Hao, Li Changchun, Liu Xiaodong, and Zhang Jinying. Calibration of Stewart platform based on coordinate measurement. In *Proceedings of 2011 International Conference on Modelling Identification and Control*, pages 469–474, Shanghai, China, June 2011. IEEE.

[3] Longfei Zhao, Ahmed Joubair, Pascal Bigras, and Ilian A Bonev. Metrological evaluation of a novel medical robot and its kinematic calibration. *International Journal of Advanced Robotic Systems*, 12(9):126, 2015.

[4] Robotic Industries Association. Unimate - the first industrial robot. `https://www.robotics.org/joseph-engelberger/unimate.cfm`. Accessed March 4, 2020.

[5] Stäubli INC. Product range - robotics experts in man and machine. `https://www.staubli.com/en/file/7985.show`. Accessed March 4, 2020.

[6] Jean-Pierre Merlet. *Parallel robots*. Springer Science & Business Media, Berlin, Germany, 2012.

[7] Omeron INC. Omron adept hornet 565. `http://onexia.com/adept/pdf/onexia-adept-hornet-robot.pdf`. Accessed March 4, 2020.

[8] Ilian Alexandrov Bonev. *Geometric analysis of parallel mechanisms*. Citeseer, 2002.

[9] Servo & Simulation INC. 6-dof mobile motion platform. `https://www.picuki.com/media/2269110919161114600`. Accessed March 4, 2020.

[10] Mario G Perhinschi and Marcello R Napolitano. Flight dynamics and control education enhancement using a motion based flight simulator. In *Proceedings of the Spring 2007 American Society of Mechanical Engineering Education North Central Section Conference at West Virginia Institute of Technology (WVUTech)*, Beckley, USA, May 2007. ASEE.

[11] Hamid D Taghirad. *Parallel robots: mechanics and control*. CRC press, Boca Raton, FL, USA, 2013.

[12] Nicolas Andreff, Tej Dallej, and Philippe Martinet. Image-based visual servoing of a gough—stewart parallel manipulator using leg observations. *The International Journal of Robotics Research*, 26(7):677–687, 2007.

[13] David Groppe. Robots improve the quality and cost-effectiveness of composite structures. *Industrial Robot: An International Journal*, 27(2):96–102, 2000.

[14] Nicolas Andreff and Philippe Martinet. Unifying kinematic modeling, identification, and control of a gough–stewart parallel robot into a vision-based framework. *IEEE Transactions on Robotics*, 22(6):1077–1086, 2006.

[15] Gangqi Dong and Zheng H Zhu. Kinematics-based incremental visual servo for robotic capture of non-cooperative target. *Robotics and Autonomous Systems*, 112:221–228, 2019.

[16] Yoshiaki Shirai and Hirochika Inoue. Guiding a robot by visual feedback in assembling tasks. *Pattern recognition*, 5(2):99IN3107–106108, 1973.

[17] Madhusudan Raghavan. The stewart platform of general geometry has 40 configurations. *Journal of Mechanical Design*, 115(2):277–282, 1993.

[18] Mohammad Keshmiri, Wen-Fang Xie, and Abolfazl Mohebbi. Augmented image-based visual servoing of a manipulator using acceleration command. *IEEE Transactions on Industrial Electronics*, 61(10):5444 – 5452, 2014.

[19] Gangqi Dong and Zheng H Zhu. Predictive visual servo kinematic control for autonomous robotic capture of non-cooperative space target. *Acta Astronautica*, 151:173–181, 2018.

[20] Tingting Shu, Sepehr Gharaaty, Wen-Fang Xie, Ahmed Joubair, and Ilian A Bonev. Dynamic path tracking of industrial robots with high accuracy using photogrammetry sensor. *IEEE/ASME Transactions on Mechatronics*, 23(3):1159–1170, 2018.

[21] Houssem Abdellatif and Bodo Heimann. Advanced model-based control of a 6-dof hexapod robot: A case study. *IEEE/ASME Transactions On Mechatronics*, 15(2):269–279, 2010.

[22] Panagiotis D. Christofides. Robust output feedback control of nonlinear singularly perturbed systems. *Automatica*, 36(1):45–52, 2000.

[23] Steve Gregory. Fuzzy overlapping communities in networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2011(02):P02017, 2011.

[24] K Liu, FL Lewis, and M Fitzgerald. Solution of nonlinear kinematics of a parallel-link constrained stewart platform manipulator. *Circuits, Systems and Signal Processing*, 13(2-3):167–183, 1994.

[25] Pratik J Parikh and Sarah SY Lam. A hybrid strategy to solve the forward kinematics problem in parallel manipulators. *IEEE Transactions on Robotics*, 21(1):18–25, 2005.

[26] Ali Ghasemi, Mohammad Eghtesad, and Mehrdad Farid. Neural network solution for forward kinematics problem of cable robots. *Journal of Intelligent & Robotic Systems*, 60(2):201–215, 2010.

[27] A Omran, M Bayoumi, A Kassem, and G El-Bayoumi. Optimal forward kinematics modeling of stewart manipulator using genetic algorithms. *Jordan Journal of Mechanical and Industrial Engineering*, 3(4):280–293, 2009.

[28] Bhaskar Dasgupta and TS Mruthyunjaya. A newton-euler formulation for the inverse dynamics of the stewart platform manipulator. *Mechanism and machine theory*, 33(8):1135–1152, 1998.

[29] Charles C Nguyen and Farhad J Pooran. Dynamic analysis of a 6 dof ckcm robot end-effector for dual-arm telerobot systems. *Robotics and Autonomous Systems*, 5(4):377–394, 1989.

[30] Jiegao Wang and Clément M Gosselin. A new approach for the dynamic analysis of parallel manipulators. *Multibody System Dynamics*, 2(3):317–334, 1998.

[31] Kourosh E Zanganeh, Rosario Sinatra, and Jorge Angeles. Kinematics and dynamics of a six-degree-of-freedom parallel manipulator with revolute legs. *Robotica*, 15(4):385–394, 1997.

[32] Ahmet Dumlu and Koksal Erenturk. Modeling and trajectory tracking control of 6-dof rss type parallel manipulator. *Robotica*, 32(4):643–657, 2014.

[33] Oren Masory, Jian Wang, and Hanqi Zhuang. Kinematic modeling and calibration of a Stewart platform. *Advanced Robotics*, 11(5):519–539, 1996.

[34] Wisama Khalil and Sébastien Besnard. Self calibration of Stewart-Gough parallel robots without extra sensors. *IEEE Transactions on Robotics and Automation*, 15(6):1116–1121, 1999.

[35] Vijay Pradeep, Kurt Konolige, and Eric Berger. Calibrating a multi-arm multi-sensor robot: A bundle adjustment approach. In *Springer Tracts in Advanced Robotics*, volume 79, pages 211–225, Berlin, Heidelberg, 2014. Springer.

[36] David Daney, Nicolas Andreff, Gilles Chabert, and Yves Papegay. Interval method for calibration of parallel robots: Vision-based experiments. *Mechanism and Machine Theory*, 41(8):929–944, 2006.

[37] John M. Hollerbach and Charles W. Wampler. The Calibration Index and Taxonomy for Robot Kinematic Calibration Methods. *The International Journal of Robotics Research*, 15(6):573–591, 1996.

[38] D. Daney. Choosing Measurement Poses for Robot Calibration with the Local Convergence Method and Tabu Search. *The International Journal of Robotics Research*, 24(1994):501–518, 2005.

[39] Newport News. Calibration of Stewart Platforms Using Neural Networks. In *Proceedings of 2012 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)*, pages 170–175, Madrid, Spain, May 2012. IEEE.

[40] Dayong Yu, Hongren Li, and Weifang Chen. Kinematic calibration of parallel robots for docking mechanism motion simulation. *International Journal of Advanced Robotic Systems*, 8(4):47, 2011.

[41] Liping Wang, Yuzhe Liu, Jun Wu, Jinsong Wang, and Binbin Zhang. Study of error modeling in kinematic calibration of parallel manipulators. *International Journal of Advanced Robotic Systems*, 13(5):1729881416672560, 2016.

[42] ZVIS Roth, B Mooring, and Bahram Ravani. Overview of Robot Calibration. *IEEE Journal on Robotics and Automation*, 5(3):377–385, 1987.

[43] Pierre Renaud, Nicolas Andreff, Jean Marc Lavest, and Michel Dhome. Simplifying the kinematic calibration of parallel mechanisms using vision-based metrology. *IEEE Transactions on Robotics*, 22(1):12–22, 2006.

[44] Hanqi Zhuang. Self-calibration of parallel mechanisms with a case study on Stewart platforms. *IEEE Transactions on Robotics and Automation*, 13(3):387–397, 1997.

[45] Mansour Abtahi, Hodjat Pendar, Aria Alasty, and Gholamreza Vossoughi. Experimental kinematic calibration of parallel manipulators using a relative position error measurement system. *Robotics and Computer-Integrated Manufacturing*, 26(6):799–804, 2010.

[46] S Besnard, W Khalil, and De Noë. Calibration of parallel robots using two inclinometers. In *Proceedings of 1999 IEEE International Conference on Robotics and Automation*, volume 3, pages 0–5, Detroit, USA, May 1999. IEEE.

[47] Soichi Ibaraki, Takeshi Yokawa, Yoshiaki Kakino, Masao Nakagawa, and Tetsuya Matsushita. Kinematic calibration on a parallel kinematic machine tool of the stewart platform by circular tests. In *Proceedings of the American Control Conference*, volume 2, pages 1394–1399, Boston, USA, June 2004. IEEE.

[48] Albert Nubiola, Mohamed Slamani, Ahmed Joubair, and Ilian A Bonev. Comparison of two calibration methods for a small industrial robot based on an optical cmm and a laser tracker. *Robotica*, 32(3):447–466, 2014.

[49] Charles W Wampler, John M Hollerbach, and Tatsuo Arai. An implicit loop method for kinematic calibration and its application to closed-chain mechanisms. *IEEE Transactions on Robotics and Automation*, 11(5):710–724, 1995.

[50] Lennart Ljung, editor. *System Identification (2Nd Ed.): Theory for the User*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.

[51] John Hollerbach, Wisama Khalil, and Maxime Gautier. Model identification. In *Springer Handbook of Robotics*, pages 113–138. Springer, 2016.

[52] Jan Swevers, Walter Verdonck, and Joris De Schutter. Dynamic model identification for industrial robots. *IEEE control systems magazine*, 27(5):58–71, 2007.

[53] Jun Wu, Jinsong Wang, and Zheng You. An overview of dynamic parameter identification of robots. *Robotics and computer-integrated manufacturing*, 26(5):414–419, 2010.

[54] Andrea Calanca, Luca M Capisani, Antonella Ferrara, and Lorenza Magnani. Mimo closed loop identification of an industrial robot. *IEEE Transactions on Control Systems Technology*, 19(5):1214–1224, 2010.

[55] Martin M Olsen, Jan Swevers, and Walter Verdonck. Maximum likelihood identification of a dynamic robot model: Implementation issues. *The international Journal of robotics research*, 21(2):89–96, 2002.

[56] Alexandre Janot, Pierre Olivier Vandanjon, and Maxime Gautier. An instrumental variable approach for rigid industrial robots identification. *Control Engineering Practice*, 25:85–101, 2014.

[57] Maxime Gautier, Alexandre Janot, and Pierre-Olivier Vandanjon. A new closed-loop output error method for parameter identification of robot dynamics. *IEEE Transactions on Control Systems Technology*, 21(2):428–444, 2013.

[58] Eric Walter and Luc Pronzato. *Identification of parametric models from experimental data*. Springer Verlag, London,UK, 1997.

[59] Andrea Del Prete, Nicolas Mansard, Oscar E Ramos, Olivier Stasse, and Francesco Nori. Implementing torque control with high-ratio gear boxes and without joint-torque sensors. *International Journal of Humanoid Robotics*, 13(01):1550044, 2016.

[60] Mathieu Brunot, Alexandre Janot, Peter C Young, and Francisco Carrillo. An improved instrumental variable method for industrial robot model identification. *Control Engineering Practice*, 74:107–117, 2018.

[61] Sébastien Briot and Maxime Gautier. Global identification of joint drive gains and dynamic parameters of parallel robots. *Multibody System Dynamics*, 33(1):3–26, 2015.

[62] Martin Grotjahn, Bodo Heimann, and Houssem Abdellatif. Identification of friction and rigid-body dynamics of parallel kinematic structures for model-based control. *Multibody System Dynamics*, 11(3):273–294, 2004.

[63] Andrès Vivas, Philippe Poignet, Frédéric Marquet, François Pierrot, and Maxime Gautier. Experimental dynamic identification of a fully parallel robot. In *Proceedings of 2003 IEEE International Conference on Robotics and Automation*, volume 3, pages 3278–3283, Taipei, Taiwan, China, September 2003. IEEE.

[64] Sylvain Guegan, Wisama Khalil, and Philippe Lemoine. Identification of the dynamic parameters of the orthoglide. In *Proceedings of 2003 IEEE International Conference on Robotics and Automation*, volume 3, pages 3272–3277, Taipei, Taiwan, China, September 2003. IEEE.

[65] Pierre Renaud, Andres Vivas, Nicolas Andreff, Philippe Poignet, Philippe Martinet, François Pierrot, and Olivier Company. Kinematic and dynamic identification of parallel mechanisms. *Control engineering practice*, 14(9):1099–1109, 2006.

[66] Chris Lightcap and Scott Banks. Dynamic identification of a mitsubishi pa10-6ce robot using motion capture. In *Proceedings of 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3860–3865, San Diego, CA, USA, 2007.

[67] Ryad Chellal, Loïc Cuvillon, and Edouard Laroche. Model identification and vision-based h∞ position control of 6-dof cable-driven parallel robots. *International Journal of Control*, 90(4):684–701, 2017.

[68] Pengcheng Li, Rui Zeng, Wenfang Xie, and Xiaoming Zhang. Relative posture-based kinematic calibration of a 6-rss parallel robot by optical coordinate measurement machine. *International Journal of Advanced Robotic Systems*, 15(2):1729881418765861, 2018.

[69] Redwan Dahmouche, Nicolas Andreff, Youcef Mezouar, Omar Ait-Aider, and Philippe Martinet. Dynamic visual servoing from sequential regions of interest acquisition. *The International Journal of Robotics Research*, 31(4):520–537, 2012.

[70] Nicolas Andreff and Philippe Martinet. Visually servoing a gough-stewart parallel robot allows for reduced and linear kinematic calibration. In *Proceedings of the Second International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, pages 119–124, Barcelona, Spain, September 2005.

[71] Koichi Hashimoto. A review on vision-based control of robot manipulators. *Advanced Robotics*, 17(10):969–991, 2003.

[72] François Chaumette and Seth Hutchinson. Visual servo control. i. basic approaches. *IEEE Robotics & Automation Magazine*, 13(4):82–90, 2006.

[73] Nicolas Andreff and Philippe Martinet. Vision-based self-calibration and control of parallel kinematic mechanisms without proprioceptive sensing. *Intelligent Service Robotics*, 2(2):71–80, 2009.

[74] Pengcheng Li, Ahmad Ghasemi, Wenfang Xie, and Wei Tian. Visual closed-loop dynamic model identification of parallel robots based on optical cmm sensor. *Electronics*, 8(8):836, 2019.

[75] Alberto Trasloheros, José M Sebastián, Jesús Torrijos, Ricardo Carelli, and Flavio Roberti. Using a 3dof parallel robot and a spherical bat to hit a ping-pong ball. *International Journal of Advanced Robotic Systems*, 11(5):76, 2014.

[76] Enrique Coronado, Mauro Maya, Antonio Cardenas, Orlando Guarneros, and Davide Piovesan. Vision-based control of a delta parallel robot via linear camera-space manipulation. *Journal of Intelligent & Robotic Systems*, 85(1):93–106, 2017.

[77] Sébastien Briot, Philippe Martinet, and Victor Rosenzveig. The hidden robot: an efficient concept contributing to the analysis of the controllability of parallel robots in advanced visual servoing techniques. *IEEE Transactions on Robotics*, 31(6):1337–1352, 2015.

[78] Shangke Lyu and Chien Chern Cheah. Vision based neural network control of robot manipulators with unknown sensory jacobian matrix. In *Proceedings of 2018 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 1222–1227, Auckland, USA, July 2018. IEEE.

[79] Nabanita Adhikary and Chitralekha Mahanta. Inverse dynamics based robust control method for position commanded servo actuators in robot manipulators. *Control Engineering Practice*, 66:146–155, 2017.

[80] Oussama Khatib, Peter Thaulad, and Jaehoung Park. Torque-position transformer for task control of position controlled robots, May 1 2007. US Patent 7,211,979.

[81] Moussâb Bennehar, Ahmed Chemori, Mohamed Bouri, Laurent Frédéric Jenni, and François Pierrot. A new rise-based adaptive control of pkms: design, stability analysis and experiments. *International Journal of Control*, 91(3):593–607, 2018.

[82] Dong Hwan Kim, Ji-Yoon Kang, and Kyo-Il Lee. Robust tracking control design for a 6 dof parallel manipulator. *Journal of Robotic Systems*, 17(10):527–547, 2000.

[83] HongBo Guo, YongGuang Liu, GuiRong Liu, and HongRen Li. Cascade control of a hydraulically driven 6-dof parallel robot manipulator based on a sliding mode. *Control Engineering Practice*, 16(9):1055–1068, 2008.

[84] Yangjun Pi and Xuanyin Wang. Trajectory tracking control of a 6-dof hydraulic parallel robot manipulator with uncertain load disturbances. *Control Engineering Practice*, 19(2):185–193, 2011.

[85] Zhen Qi, John E McInroy, and Farhad Jafari. Trajectory tracking with parallel robots using low chattering, fuzzy sliding mode controller. *Journal of Intelligent and Robotic Systems*, 48(3):333–356, 2007.

[86] James E Dieudonne. An actuator extension transformation for a motion simulator and an inverse transformation applying newton-raphson's method. *NASA Technical Note, NASA TN 0-7067*, 1972.

[87] James Hardy Wilkinson. *The algebraic eigenvalue problem*, volume 662. Oxford Clarendon, Oxford,UK, 1965.

[88] Michael JD Powell. A hybrid method for nonlinear equations. In *Numerical Methods for Nonlinear Algebraic Equations*. Gordon and Breach, 1970.

[89] Wen-Fang Xie, Marek Krzeminski, El-Tahan Hussein, and El-Tahan Mona. Intelligent friction compensation (ifc) in a harmonic drive. In *Proceedings of Newfoundland Electrical and Computer Engineering Conference (NECEC)*, St. John's, Canada, November 2002. IEEE.

[90] Liao Wu and Hongliang Ren. Finding the kinematic base frame of a robot by hand-eye calibration using 3d position data. *IEEE Transactions on Automation Science and Engineering*, 14(1):314–324, 2017.

[91] Benjamin W Mooring, Zvi S Roth, and Morris R Driels. *Fundamentals of manipulator calibration*. Wiley, New York, USA, 1991.

[92] Yoshiaki Shirai. *Three-dimensional computer vision*. Springer Science & Business Media, Berlin, German, 2012.

[93] Richard I Hartley and Peter Sturm. Triangulation. *Computer vision and image understanding*, 68(2):146–157, 1997.

[94] Kenichi Kanatani, Yasuyuki Sugaya, and Hirotaka Niitsuma. Triangulation from two views revisited: Hartley-sturm vs. optimal correction. *In practice*, 4:5, 2008.

[95] William J Wilson, Carol C Williams Hulls, and Graham S Bell. Relative end-effector control using cartesian position based visual servoing. *IEEE Transactions on Robotics and Automation*, 12(5):684–696, 1996.

[96] Joseph SC Yuan. A general photogrammetric method for determining object position and orientation. *IEEE Transactions on Robotics and Automation*, 5(2):129–142, 1989.

[97] Konstantinos Daniilidis. Hand-eye calibration using dual quaternions. *The International Journal of Robotics Research*, 18(3):286–298, 1999.

[98] Ning Tan, Xiaoyi Gu, and Hongliang Ren. Simultaneous robot-world, sensor-tip, and kinematics calibration of an underactuated robotic hand with soft fingers. *IEEE Access*, 6:22705–22715, 2017.

[99] Rui Zeng, Shu-ling Dai, Wen-Fang Xie, and Zhang Xiao-Ming. Determination of the Proper Motion Range of the Rotary Actuators of 6-RSS Parallel Robot. In *Proceedings of 2015 CCToMM Symposium on Mechanisms, Machines, and Mechatronics*, pages 94–105, Ottawa, Camada, May 2015. CCToMM.

[100] Rui Zeng, Shuling Dai, Wen-Fang Xie, and Rama Bhat. Constraint conditions determination for singularity-free workspace of central symmetric parallel robots. In *15th IFAC Symposium on Information Control Problems in Manufacturing*, volume 28, pages 1930–1935, Ottawa, Canada, May 2015. IFAC.

[101] F C Park and J W Kim. Singularity analysis of closed loop kinematic chains. *Trans. of ASME, Journal of Mechanical Design*, 121(1):32–38, 1999.

[102] Ilian A. Bonev, Dimiter Zlatanov, and Clément M. Gosselin. Singularity Analysis of 3-DOF Planar Parallel Mechanisms via Screw Theory. *Journal of Mechanical Design*, 125(3):573, 2003.

[103] Jin-Hwan Borm and Chia-Hsiang Meng. Determination of optimal measurement configurations for robot calibration based on observability measure. *The International Journal of Robotics Research*, 10(1):51–63, 1991.

[104] Albert Nubiola and Ilian A Bonev. Absolute calibration of an abb irb 1600 robot using a laser tracker. *Robotics and Computer-Integrated Manufacturing*, 29(1):236–245, 2013.

[105] Coleman Brosilow and Babu Joseph. *Techniques of model-based control*. Prentice Hall Professional, Upper Saddle River, USA, 2002.

[106] Thomas E Marlin. *Process control*. McGraw-Hill International Editions: Europe, London, UK, 1995.

[107] Kyung-Jo Park. Fourier-based optimal excitation trajectories for the dynamic identification of robots. *Robotica*, 24(5):625–633, 2006.

[108] Basilio Bona and Aldo Curatella. Identification of industrial robot parameters for advanced model-based controllers design. In *Proceedings of International Conference on Robotics and Automation (ICRA)*, pages 1681–1686, Barcelona, Spain, 2005. IEEE.

[109] JJP Van Boekel. Simmechanics, maplesim and dymola: a first look on three multibody packages, 2009. Accessed on: Nov. 7, 2018.

[110] Maxime Gautier. Numerical calculation of the base inertial parameters of robots. *Journal of robotic systems*, 8(4):485–506, 1991.

[111] HD Taghirad and PR Belanger. Modeling and parameter identification of harmonic drive systems. *Journal of dynamic systems, measurement, and control*, 120(4):439–444, 1998.

[112] PR Belanger, P Dobrovolny, A Helmy, and X Zhang. Estimation of angular velocity and acceleration from shaft-encoder measurements. *The International Journal of Robotics Research*, 17(11):1225–1233, 1998.

[113] Cong Wang, Chung-Yen Lin, and Masayoshi Tomizuka. Statistical learning algorithms to compensate slow visual feedback for industrial robots. *Journal of Dynamic Systems, Measurement, and Control*, 137(3):031011, 2015.

[114] Dan Simon. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, Hoboken, USA, 2006.

[115] Mahsa Parsapour, Soheil RayatDoost, and Hamid D. Taghirad. Position based sliding mode control for visual servoing system. In *2013 First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*, pages 337–342, Tehran, Iran, Feberary 2013. IEEE.

Appendix

The derivative of the wrench Jacobian, $\dot{J}_{a_i}$, is given as follows:

$$\dot{J}_{a_i} = \begin{bmatrix} \dot{J}_{au_i} \\ \hat{\boldsymbol{s}}\dot{J}_{ad_i} \end{bmatrix}, \tag{A.1}$$

where

$$\dot{J}_{ad_i} = -\frac{\dot{m}}{m^2}\left[\ \boldsymbol{l_i}^T \quad (\boldsymbol{a_i} \times \boldsymbol{l_i})^T\ \right] + \frac{1}{m}\left[\ (\boldsymbol{\omega_{2i}} \times \boldsymbol{l_i})^T \quad ([\boldsymbol{a_i}]_X[\boldsymbol{\omega_2}]_X\boldsymbol{l_i})^T - ([\boldsymbol{l_i}]_X[\boldsymbol{\omega}]_X\boldsymbol{a_i})^T\ \right], \tag{A.2}$$

and

$$\begin{aligned} m &= (\boldsymbol{w_i} \times \boldsymbol{l_i}) \cdot \hat{\boldsymbol{s}} \\ \dot{m} &= ([\boldsymbol{w_i}]_X[\boldsymbol{\omega_2}]_X\boldsymbol{l_i} - [\boldsymbol{l_i}]_X[\boldsymbol{\omega_1}]_X\boldsymbol{w_i}) \cdot \hat{\boldsymbol{s}} \\ \dot{J}_{au_i} &= ([\hat{\boldsymbol{s}}]_X[\boldsymbol{\omega_1}]_X\boldsymbol{c_{w_i}})J_{ad_i} + ([\hat{\boldsymbol{s}}]_X\boldsymbol{c_{w_i}})\dot{J}_{ad_i} \end{aligned} \tag{A.3}$$

In addition, the link Jacobian $\dot{J}_{b_i}$ is obtained by:

$$\dot{J}_{b_i} = \begin{bmatrix} \dot{J}_{bu_i} \\ \dot{J}_{bd_i} \end{bmatrix} \tag{A.4}$$

in which

$$\begin{aligned} \dot{J}_{bd_i} =& \frac{1}{\|\boldsymbol{l_i}\|^2}\{[\boldsymbol{\omega_2} \times \boldsymbol{l_i}]_X[\boldsymbol{w_i}]_X\hat{\boldsymbol{s}}J_{ad_i} + \\ & \left[\ [\boldsymbol{\omega_2} \times \boldsymbol{l_i}]_X \quad [\boldsymbol{l_i}]_X[\boldsymbol{\omega} \times \boldsymbol{a_i}]_X - [\boldsymbol{\omega_2} \times \boldsymbol{l_i}]_X[\boldsymbol{a_i}]_X\ \right] \\ & + [\boldsymbol{l_i}]_X[\boldsymbol{\omega_1} \times \boldsymbol{w_i}]_X\hat{\boldsymbol{s}}J_{ad_i} + [\boldsymbol{l_i}]_X[\boldsymbol{w_i}]_X\hat{\boldsymbol{s}}\dot{J}_{ad_i}\}, \end{aligned} \tag{A.5}$$

and

$$\dot{J}_{bu_i} = -[\boldsymbol{\omega_1} \times \boldsymbol{w_i}]_X \hat{\boldsymbol{s}} J_{ad_i} - [\boldsymbol{w_i}]_X \hat{\boldsymbol{s}} \dot{J}_{ad_i} - [\boldsymbol{\omega_2} \times \boldsymbol{l_i}]_X J_{bd_i} - [\boldsymbol{c_{l_i}}]_X \dot{J}_{bd_i} \tag{A.6}$$