Lappeenranta
University of Technology

Yongbo Wang

NOVEL METHODS FOR ERROR MODELING AND PARAMETER IDENTIFICATION OF A REDUNDANT SERIAL-PARALLEL HYBRID ROBOT

Thesis for the degree of Doctor of Science (Technology) to be presented with due permission for public examination and criticism in Auditorium 1382 at Lappeenranta University of Technology, Lappeenranta, Finland, on the 13th of December, 2012, at noon

Acta Universitatis
Lappeenrantaensis 500

Supervisor

Professor Heikki Handroos
Laboratory of Intelligent Machines
Faculty of Technology
Lappeenranta University of Technology
Finland

Associate professor Huapeng Wu
Laboratory of Intelligent Machines
Faculty of Technology
Lappeenranta University of Technology
Finland

Reviewers

Professor Marco Ceccarelli
Laboratory of Robotics and Mechatronics
University of Cassino
Italy

Associate professor Luc Rolland
Faculty of Engineering and Applied Sciences
Memorial University of Newfoundland
Canada

Opponents

Associate professor Giuseppe Carbone
Laboratory of Robotics and Mechatronics
University of Cassino
Italy

Associate professor Luc Rolland
Faculty of Engineering and Applied Sciences
Memorial University of Newfoundland
Canada

ISBN 978-952-265-344-4
ISBN 978-952-265-345-1 (PDF)
ISSN 1456-4491
Lappeenrannan teknillinen yliopisto
Yliopistopaino 2012

ABSTRACT

Yongbo Wang

Novel Methods for Error Modeling and Parameter Identification of a Redundant Serial-Parallel Hybrid Robot

Lappeenranta 2012
96 p.
Acta Universitatis Lappeenrantaensis 500
Diss. Lappeenranta University of Technology
ISBN 978-952-265-344-4, ISBN 978-952-265-345-1 (PDF), ISSN 1456-4491

To obtain the desirable accuracy of a robot, there are two techniques available. The first option would be to make the robot match the nominal mathematic model. In other words, the manufacturing and assembling tolerances of every part would be extremely tight so that all of the various parameters would match the “design” or “nominal” values as closely as possible. This method can satisfy most of the accuracy requirements， but the cost would increase dramatically as the accuracy requirement increases. Alternatively, a more cost-effective solution is to build a manipulator with relaxed manufacturing and assembling tolerances. By modifying the mathematical model in the controller, the actual errors of the robot can be compensated. This is the essence of robot calibration. Simply put, robot calibration is the process of defining an appropriate error model and then identifying the various parameter errors that make the error model match the robot as closely as possible.

This work focuses on kinematic calibration of a 10 degree-of-freedom (DOF) redundant serial-parallel hybrid robot. The robot consists of a 4-DOF serial mechanism and a 6-DOF hexapod parallel manipulator. The redundant 4-DOF serial structure is used to enlarge workspace and the 6-DOF hexapod manipulator is used to provide high load capabilities and stiffness for the whole structure. The main objective of the study is to develop a suitable calibration method to improve the accuracy of the redundant serial-parallel hybrid robot. To this end, a Denavit–Hartenberg (DH) hybrid error model and a Product-of-Exponential (POE) error model are developed for error modeling of the proposed robot. Furthermore, two kinds of global optimization methods, i.e. the differential-evolution (DE) algorithm and the Markov Chain Monte Carlo (MCMC) algorithm, are employed to identify the parameter errors of the derived error model. A measurement method based on a 3-2-1 wire-based pose estimation system is proposed and implemented in a Solidworks environment to simulate the real experimental validations. Numerical simulations and Solidworks prototype-model validations are carried out on the hybrid robot to verify the effectiveness, accuracy and robustness of the calibration algorithms.

Keywords: error modeling, parameter identification, kinematic calibration, hybrid robot, serial-parallel robot, Markov Chain Monte Carlo, product-of-exponential, differential-evolution.

UDC 621.865.8:51.001.57:519.245:519.217.2

ACKNOWLEDGEMENTS

First of all, I would like to express my deepest gratitude to my supervisors, Professor Heikki Handroos and Associate professor Huapeng Wu, for giving me the opportunity to participate in this interesting research project and for organizing the financial support of my study, their inspiring guidance, suggestions and continuous encouragement guaranteed the timely accomplishment of my doctoral study. Huapeng's valuable insights and patient assistance have been significant benefit to my research efforts. I always feel fortunate to have both of you as my supervisors.

I am extremely appreciative of my thesis reviewers and opponents, Professor Marco Ceccarelli, associate professor Luc Rolland, associate professor Giuseppe Carbone, for their constructive and insightful comments and criticisms, which are very helpful to improve the quality of my thesis. Furthermore, I am also appreciative of Ms. Barbara Miraftabi, Dr. Junhong Liu, Ms. Mei Han for their proof reading and language corrections. I am especially grateful that they were able to review my thesis in a very tight schedule.

I am grateful for the financial support from the following foundations in different stages: the Academy of Finland; Graduate School Concurrent Mechanical Engineering (GSCME); the Research Foundation of Lappeenranta University of Technology (LTY: Tukisäätiö); China Scholarship Council (CSC).

During my doctoral study, I got lots of opportunities to attend conferences and seminars. It is very beneficial to improve not only my knowledge in my research area but also the capability of understanding different cultures. I have also benefited a lot from a four months research mobility work in Laboratory of Robotics and Mechatronics (LARM) at Cassino University, Italy. I would like to thank the financial support from LUT research mobility program and the invitation from Prof. Marco Ceccarelli. The work performed during this short period plays an important role in my dissertation. I would like to give my thanks to Dr. Giuseppe Carbone, Mr. Tao Li, and Mr. Franco Tedeschi for their help during my stay in LARM.

My special thanks go to Professor Bingkui Chen, at Chongqing University, China. I greatly appreciate his encouragement upon my pursuing of doctoral degree at Lappeenranta University of Technology (LUT), and I am also grateful to the research work at Chongqing University under his guidance, which prepared lots of fundamental knowledge for my study at LUT.

I would like to thank all my colleagues and friends for giving me a pleasant work environment as well as a colorful leisure time. During my work in Lappeenranta, I shall never forget my colleagues and friends with whom I shared many enjoyable discussions and memorable moment: they are Dr. Junhong Liu, Mr. Ming Li, Ms. Qiumei Li, Dr. Jianzhong Hong, Ms. Sha Sha, Mr. Yuwei Bie, Mr. Xing Shi, Ms. Bing Han, Ms. Mei Han, Ms. Shu Meng, Ms. Matina Ma, Dr. Rafael Åman, Mr. Lauri Luostarinen, Mr. Mazin Al-Saedi, Mr. Juha Koivisto, to name only a few.

Last but not least, I would give my heartfelt thanks to my closest family. I am greatly in debt to all my family members for their encouragement and support. My dearest parents are the strongest supporters throughout my life. Without their open-minded discipline, I could not have been what I am now. I would like to thank my beloved Helen for helping me polishing my English, and especially thank you for your supporting, encouragement, and always being there for me. I also would like to thank my brothers, Yongjian, Yonglin and Qingyou, for giving me advice and sharing happiness and difficulties with me throughout my life. Finally, I would like to dedicate this thesis to my beloved grandparents who watched me growing up, although they passed away during my study in Finland, their kindness and benevolence will remain forever engraved on my memory.

Yongbo Wang

Lappeenranta, December, 2012

CONTENTS

LIST OF ORIGINAL ARTICLES

This thesis, based on published papers, includes an introductory part and seven original refereed articles. Papers 1 through 4 have been published in scientific journals. Paper 5 has been submitted for review in the Journal of Fusion Engineering and Design. Papers 6 and 7 have been presented at international conferences and they can be regarded as a supplementary part of journal paper 2 and paper 5. The articles are summarized below:

Refereed scientific journal articles

1. Wang, Yongbo & Wu, Huapeng & Handroos, Heikki (2012). **Error Modelling and Differential-Evolution-Based Parameter Identification Method for Redundant Hybrid Robot.** *International Journal of Modelling and Simulation, vol.32, No. 4*, 2012, p. 255-264.

2. Wang, Yongbo & Wu, Huapeng & Handroos, Heikki (2011). **Markov Chain Monte Carlo (MCMC) Methods for Parameter Estimation of a Novel Hybrid Redundant Robot.** *Journal of Fusion Engineering and Design, 2011, vol. 86,* p. 1863-1867.

3. Wu, Huapeng & Handroos, Heikki & Pelab P. & Wang, Yongbo (2011). **IWR-solution for the ITER Vacuum Vessel Assembly.** *Journal of Fusion Engineering and Design, 2011, vol. 86,* p. 1834-1837.

4. Wang, Yongbo & Pessi, Pekka & Wu, Huapeng & Handroos, Heikki (2009). **Accuracy Analysis of Hybrid Parallel Robot for the Assembling of ITER**, *Journal of Fusion Engineering and Design, 2009, vol. 84, No. 2*, p. 1964-1968.

5. Wang, Yongbo & Wu, Huapeng & Handroos, Heikki. **Accuracy Improvement of a Hybrid Robot for ITER Application Using POE Modeling Method**, *Journal of Fusion Engineering and Design (Under review).*

Refereed conference articles

6. Wang, Yongbo & Wu, Huapeng & Handroos, Heikki. **Identifiable Parameter Analysis for the Kinematic Calibration of a Hybrid Robot**. *The ASME 2011 International Design Engineering Technical Conferences (IDETC) and Computers and Information in Engineering Conference (CIE)*, Aug. 28-31, 2011, Washington DC, USA, p. 911-919.

7. Wang, Yongbo & Wu, Huapeng & Handroos, Heikki. **Differential-Evolution-based Parameter Identification Method for a Redundant Hybrid Robot Using POE Model**. *The 43rd International Symposium on Robotics (ISR 2012)*, Aug. 29-31, 2012, Taipei, Taiwan, p. 974-979.

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional
CAD	Computer Aided Design
CR	Crossover Rate of DE Algorithm
D	Individual Index of DE Algorithm
DE	Differential-Evolution
DH	Denavit–Hartenberg
DOF	Degree of Freedom
EAs	Evolutionary Algorithms
EKF	Extended Karman Filter
F	Mutation Scale Factor of DE Algorithm
G	Generation Index of DE Algorithm
GA	Genetic Algorithms
Hexa-WH	Hexapod Water Hydraulic Actuated Robot
ITER	International Thermonuclear Experimental Reactor
IWR	Intersector Welding/Cutting Robot
LM	Levenberg and Marquardt
MCMC	Markov Chain Monte Carlo
NDT	Non-Destructive Testing
NP	Number of Population
POE	Product-Of-Exponentials
PSO	Particle Swarm Optimization
R&D	Research and Development
RMS	Root Mean Square
VV	Vacuum Vessel

PART I: OVERVIEW OF THE DISSERTATION

CHAPTER 1

INTRODUCTION

1.1 Background and Motivations

This work results from a joint international R&D project named ITER (International Thermonuclear Experimental Reactor). ITER will be the largest experimental fusion facility in the world and is designed to demonstrate the scientific and technological feasibility of fusion power for energy purposes [1]. The 3D model of the ITER machine is shown in Figure 1. The vacuum vessel (VV) of ITER consists of nine sectors whose inner and outer walls are welded together by a field weld. It will measure over 19 meters across by 11 meters high, and weigh in excess of 5,000 tons [2]. The assembly of VV involves various tasks, such as welding, machining, NDT testing, measuring the gap between two adjacent sectors and transporting a premade splice plate to match the measured gap. All of these assembly tasks are required to be performed by a robot from inside the ITER VV. The detailed discussion can be found in Publication 3. Due to the requirements of a big workspace, a big payload and high accuracy (±0.1 mm) for the assembly robot, neither a commercially available serial robot nor a parallel robot can be directly used. To solve this problem, a 10 degree-of-freedom (DOF) redundant serial-parallel hybrid robot, IWR (Intersector-Welding/Cutting-Robot), was developed at Lappeenranta University of Technology, Finland [3], as shown in Figure 2. The serial part of the hybrid robot is used to enlarge workspace while the parallel part is used to provide high load capabilities and stiffness for the whole structure.

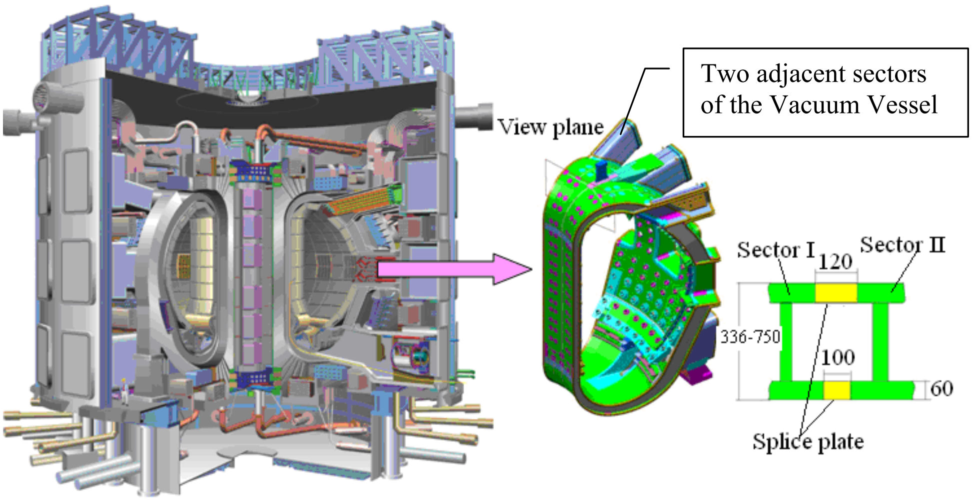
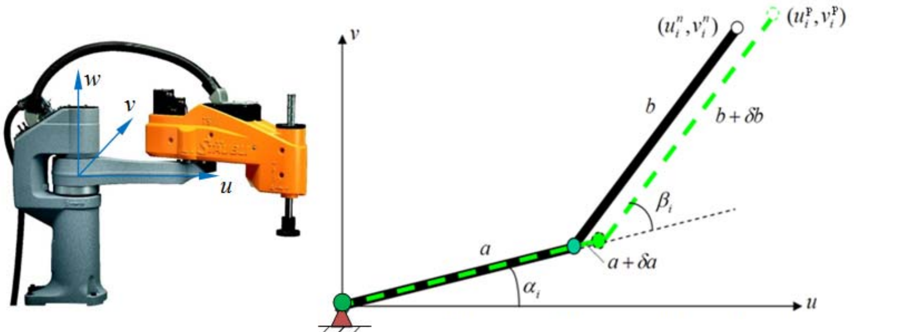


Figure 1. International Thermonuclear Experimental Reactor (ITER).

Figure 2. The experimental robot prototype developed at LUT.

The inaccuracy of a robot may originate from a number of error sources, geometric errors such as backlash, manufacturing and assembly, gear and bearing wear, measurement and control, dimensional tolerances of joint actuators and controllers, and non-geometric errors, such as temperature variation of the environment, elastic deformations of the structural components of robots and so on [4][5]. As a matter of fact, all these errors are uncertain in nature; therefore a suitable error model has to be established to predict the robot's performance. For more details of the significance of various error sources, please refer to Publication 4. The essence of kinematic calibration is to define an appropriate error model, identify a vector of parameter errors and to compensate them in the robot controller so as to make the error model match the real robot as closely as possible. It is an integrated process consisting of the modelling, measurement, identification and compensation [8]. To illustrate this calibration concept and hence provide a framework for later chapters, a simple SCARA robot is considered, as shown in Figure 3. In the design stage, the link lengths of a and b would be given by a nominal dimension and a machining tolerance limit. The two axes of the revolute joints are intended to be parallel to each other and perpendicular to the u-v plane.



a) SCARA robot b) Kinematic diagram of the two revolute joints

Figure 3. SCARA robot and kinematic diagram of its two revolute joints.

The relationship between the revolute joint displacements (α_i, β_i) and the nominal position of the end point $(\boldsymbol{y}_i^n)$ in the i^{th} pose configuration can be written as

$$\mathbf{y}_i^n = \begin{bmatrix} u_i^n \\ v_i^n \end{bmatrix} = \begin{bmatrix} a\cos\alpha_i + b\cos(\alpha_i + \beta_i) \\ a\sin\alpha_i + b\sin(\alpha_i + \beta_i) \end{bmatrix}. \tag{1}$$

To develop an error model, assume that the manipulator has been constructed and the lengths of link a and b are affected by slight machining errors δa and δb; then the error model of the two revolute joint mechanism can be written as

$$\mathbf{y}_i^{\mathrm{p}} = \begin{bmatrix} u_i^p \\ v_i^p \end{bmatrix} = \begin{bmatrix} (a+\delta a)\cos\alpha_i + (b+\delta b)\cos(\alpha_i + \beta_i) \\ (a+\delta a)\sin\alpha_i + (b+\delta b)\sin(\alpha_i + \beta_i) \end{bmatrix}. \tag{2}$$

The second step after obtaining the error model is to measure the end-point pose accurately to get a set of measured positions, $\boldsymbol{y}_i^m$. In the third step, we can establish a least-square objective function based on the deviations between the measured data and the error model predicted data. The parameter errors can be identified by optimizing the following objective function

$$f(\delta\alpha, \ \delta\beta) = \sum_{i=0}^{N} (\mathbf{y}_i^m - \mathbf{y}_i^{\mathrm{p}})^2 \quad . \tag{3}$$

In the final step, the identified parameter errors are substituted into the error model to obtain an accurate kinematic model with known parameters that characterizes an accurate relationship between the joint variables and end-effector pose.

1.2 Objective and Scope of the Study

The main objective of the study is to develop a calibration method to improve the accuracy of a serial-parallel hybrid robot. The scope of the study includes:

- Kinematic and error modeling of the serial, parallel and redundant serial-parallel hybrid robot.

- Parameter identification of high nonlinear, high dimensional, multi-modal and global optimization problems.

1.3 Main Contributions

The most significant contributions of this work are summarized as follows:

- A hybrid modeling method, a combination of the Denavit–Hartenberg (DH) modeling method and the vector chain analytical modeling method, developed to calibrate the redundant serial-parallel hybrid robot.
- Extending the product-of-exponentials (POE) modeling and calibration method from a serial robot to a redundant serial-parallel hybrid robot.
- Integrating the Marko Chain Monte Carlo (MCMC) algorithm with the Differential-Evolution (DE) optimization algorithm for parameter identification and parameter redundancy analysis.
- Employing the Differential-Evolution optimization algorithm for parameter identification of the robot with the POE-based error model.

1.4 Organization of the Thesis

This thesis consists of two parts. The first part has six chapters which gives an introductory overview. The second part is composed of five original scientific journal papers and conference articles. In the first part, Chapter 1 introduces the background, motivation, objective, research scope and contributions of the thesis; Chapter 2 gives a literature review of the error modeling and parameter identification method for robot calibration; Chapter 3 is the heart of the work. It proposes solutions to solve the kinematic and identification problems of the redundant serial-parallel hybrid robot. The main idea has been included in publications 1-7; Chapter 4 demonstrates some numerical simulations to verify the effectiveness of the proposed methods for a 10-DOF redundant serial-parallel hybrid robot developed at Lappeenranta University of Technology, Finland. The relevant work of the 10-DOF hybrid robot can be referred to the attached Publications 1-7. Chapter 5 presents a cost-effective wire-based measurement system which is simulated in the Solidworks® assembly CAD prototype model to calculate the end-effector poses of the proposed robot. Based on these measured end-effected poses, the actual experimental conditions can be simulated. Chapter 6 concludes the study.

CHAPTER 2

STATE OF THE ART – LITERITURE REVIEW

In most general situations, robot calibration can be classified into two types [6], static calibration and dynamic calibration. Static calibration identifies the parameters primarily influencing the static or time invariant positioning characteristics of a manipulator (e.g. joint-axis geometries, joint offset and gear eccentricities, etc.) whereas dynamic calibration is used to identify parameters primarily influencing motion characteristics of the manipulator (e.g. forces, actuator torques, accelerations, mass, inertias, damping, elasticity, etc.) [7].

Robot calibration is a process integrating four steps [8]: The first step is to select a suitable mathematic model to relate the joint displacements to the end-effector pose. The accuracy of the robot will be largely dependent on how accurately this mathematic model can reflect the real robot. The second step is about data acquisition. For self-calibration methods [9][10][11], the built-in sensor readings from the passive joints and the actuated-joints are imperative. The self-calibration methods are suitable for calibration of a closed-loop mechanism (parallel robot) if the passive joint displacements can be obtained from built-in sensors. Otherwise, classical or external calibration methods have to be used [12][13][14]. The purpose of the external calibration methods is to calibrate an open-loop mechanism by using an external measurement instrument to obtain the position and orientation values of the end-effector. Following the error modeling and data acquisition processes is parameter identification, which is usually carried out by means of numerical optimization methods based on least-square fitting. Finally, the identified parameters and the refined model are implemented in the robot's position control software to get the desired position.

In this work, we focus on the error modeling and parameter identification issues for a static or kinematic calibration. Section 2.1 reviews the state-of-the-art kinematic and error modeling methods for serial, parallel and hybrid serial-parallel robots. Section 2.2 gives the literature review of the main contributions made so far to parameter identification.

2.1 Kinematic and Error Modeling Methods

A kinematic model needs to be developed for static robot calibration in order to find true mapping between the joint displacements and the end-effector poses. A good kinematic model for calibration should meet three requirements, i.e., completeness, proportionality, and minimality [15][16].

Completeness: A complete model should contain a sufficient number of independent and identifiable parameters to specify the mechanical structure of a robot. For a serial robot, Khalil and Gautier [17] proposed an identification method in which the identifiable parameters are calculated from QR decomposition of the analytical observation matrix. Besnard and Khalil [18] extended this method to determine the identifiable parameters of parallel robots even though the identification Jacobian matrix cannot be obtained analytically. Furthermore, for the serial robot, the minimum number of geometrical parameters is given by Mooring et al. [8]

$$C = 4R + 2P + T\ , \tag{4}$$

where R and P are the number of revolute and prismatic joints respectively, and T is the number of end-effector pose parameters measured by an external measurement instrument.

For multi-loop parallel robots, the number of independent parameters can be calculated by using the formula proposed by Vischer [19]

$$C = 3R + P + SS + E + 6L + 6(F-1)\ , \tag{5}$$

where R is the number of revolute joints, P is the number of prismatic joints, SS is the number of pairs of spherical joints, E is the number of measurement encoders, L is the number of loops and F is the number of arbitrarily located frames.

Proportionality or model continuity: Proportionality addresses the problem of mathematic singularities, which implies that small changes in the real robot structure should reflect the corresponding small changes in the parameters. For instance, the Denavit & Hartenberg (DH) model [20] uses a minimum set of kinematic parameters to describe the relationship between two adjacent joint axes. This model can meet the completeness property, but it fails to be proportional when the two consecutive joint axes are parallel or nearly parallel. To avoid the singularity problem, a succession of models have been developed: Hayati [21] proposed a modified DH modeling method by incorporating an extra rotation parameter into the parallel revolute axes; Veitschegger and Wu [22] developed a linear and a second-order error modeling methods based on the modified DH model; Stone and Sanderson [23] developed an S-model which uses six parameters for each link and these parameters are converted to DH parameters. The zero-reference model proposed by Mooring [24] does not rely on the DH formalism; it contains a reference coordinate system fixed in the work space, and an end-effector coordinate system attached to the end-effector of the robot. The product-of-exponential (POE) model presented by Park and Okamura [25] can also be regarded as a zero-reference model which is suitable for modeling manipulators with both revolute and prismatic joints. The POE modeling method is mathematically appealing because of its connection with the Lie group, especially the one-parameter subgroups of Euclidean motions [26]. It has proven to be a useful tool in many fields such as robot kinematics [27], motion control [28][29]and descriptions of mechanical compliance [30]. Significantly, the POE model can perfectly meet the proportionality properties since the kinematic parameters in the POE model show smooth variations in accordance with the small changes in joint axes. Furthermore, it is unnecessary to attach local frames to each joint since all the kinematic parameters are expressed in a fixed reference frame.

Minimality: The kinematic model must contain only a minimal number of parameters and the redundant parameters have to be eliminated since they would deteriorate the identification result [31][32].

For a serial robot, the most popular modeling methods are the DH model and the Modified DH model. The POE modeling method has also attracted some researchers' interests in recent years. Chen, et al. [33] proposed a local POE formula for modular robot calibration. Unlike the traditional POE formula, the joint axes in the local POE formula are expressed in their respective local frames instead of in the base frame. The main advantage of this formula is that the local coordinate frames can be arbitrarily assigned onto their corresponding links. Therefore, one can always assume that the kinematic errors only exist in the initial poses of the consecutive local frames. The local POE formula has been employed for calibration of the 4-DOF SCARA type serial robot, the 5-DOF tree-typed modular robot [34] and the three-legged modular reconfigurable parallel robots [35]. In the work by He [36], the identifiability of the POE error model was discussed and the explicit expressions of the POE error model were presented. It greatly simplifies the analysis of the mechanism and makes the POE representation superior to the DH method. For parallel robots, the commonly used kinematic modeling method is the vector chain analytical method [37][38]. However, very few publications can be found and there is no generic modeling method available for a hybrid serial- parallel mechanisms. Fan, et al. [39] presented a calibration method for a hybrid five-degree-of-freedom (DOF) manipulator. In his work, the serial part of the robot is taken as a ruler to measure the end-effector's offset caused by a parallel mechanism at different configurations. The calibration error model is dependent on the measurement method. In Publications 1, 2 and 4, we propose a hybrid error modeling method for a redundant serial-parallel robot. This method combines the DH model for a serial mechanism and the vector chain analytical method for a parallel mechanism. The advantage of this method is that the external pose measurement of the connection point between serial and parallel mechanisms is avoided. Therefore, the two hybrid parts do not need to be calibrated separately but can be regarded as a whole, and then the pose measurement of the end-effector can fulfill the calibration purpose effectively. In Publications 5 and 7, we extend the application of the POE error modeling method from serial robots to serial-parallel hybrid robots.

2.2 Parameter Identification Methods

Once a suitable mathematic model has been selected for a robot, the task of parameter identification would be to select a suitable optimization method to identify the parameter errors. Generally, the optimization method in this step can be categorized into three different types: iterative linearization, nonlinear optimization and statistical estimation.

- **The iterative linearization method**

The idea behind this method is to linearize the kinematic model to obtain an identification Jacobian matrix and an initial estimation of the structural parameters, and, recursively, to solve the linear system until the average error reaches a stable minimum. The advantage of this method is less computation time to converge, but the identification Jacobian may suffer from numerical problems of ill-conditioning. To overcome this problem, the Levenberg and Marquardt (LM) minimization techniques can be used [40][41]. The application of this method for the calibration of parallel mechanisms can be seen in works [42][43][44].

- **Nonlinear optimization method**

The nonlinear optimization method minimizes the sum of square errors between the measured and predicted values based on the Euclidean norm. This method is commonly used in high nonlinear and complex systems where the identification Jacobian matrix is not easy to derive. For the nonlinear optimization method, some global optimization algorithms (such as the artificial neural network [45], genetic programming [46], particle swarm optimization (PSO) [47], genetic algorithms (GA) [48] and differential-evolution (DE) [49] algorithms) have been successfully employed to calibrate specific serial or parallel robots. Comparison of these global optimization methods for benchmark or real-world applications can be found in literature publications [50][51][52]. The benchmark comparison of DE, GA, PSO and evolutionary algorithm (EA) [50][51] demonstrated that DE algorithms are more reliable and easy-to-use than other optimization algorithms. The comparison of DE, GA and PSO [52] shows that DE is clearly and consistently superior to GA and PSO in terms of precision as well as robustness of results for hard clustering problems. In general, DE is a simple but effective evolutionary computation method to solve nonlinear and global optimization problems [53][54]. The DE-based identification method is a nonlinear optimization method and is purely stochastic; it avoids problems in defining search direction, and whether the initial values are close to the optimum solution or not is insignificant. Therefore, the development of an identification matrix is not necessary and the numerical problem of ill-conditioning of identification matrix can be avoided. Owing to the outstanding performance of DE and the complicated error model of the proposed robot, the DE algorithm was employed in Publications 1, 5 and 7 to identify parameter errors and to find numerical solutions for the robot forward kinematics.

- **Statistical estimation method**

Due to the uncertainty of parameter errors, some statistical estimation algorithms have been employed to identify robot parameters and to analyze the uncertainties of identification. Faraz [55] proposed an extended Karman filter (EKF) for the IMU-camera calibration. In the work of Omodei [56], the EKF estimation method was used to identify the parameter errors of a 4-DOF SCARA robot. In the same paper, the experimental comparison of the iterative linearization method, the nonlinear optimization method and the EKF parameter identification method for the same industrial robot were also discussed. Julier [57] pointed out that the disadvantage of the EKF is difficult to implement and tune, as it is only reliable for the systems that are almost linear on the time scale of the updates. Many of these difficulties arise from the use of linearization. If the distribution of the prediction errors deviates further from normality, for instance, when the measurement noises are not normally distributed, or when higher-order moments are needed to account for the high nonlinearities in one's model, alternative approaches, such as particle filters, MCMC methods and Gaussian mixture filters can be used [58]. In Publication 2, the MCMC method was used to estimate parameter errors of the hybrid robot. Furthermore, the MCMC method has also been used to analyze parameter redundancy and parameter identifiability of the hybrid error model in Publication 6.

CHAPTER 3

NOVEL METHODS FOR KINEMATIC CALIBRATION OF A HYBRID ROBOT

In this chapter, the main contributions of our seven publications are summarized. We propose two kinds of error modeling methods and two kinds of parameter identification methods for a 10-DOF redundant serial-parallel hybrid robot. Section 3.1 and Section 3.4 present the Denavit-Hartenberg (DH) hybrid modeling method and the Markov Chain Monte Carlo (MCMC) parameter identification method which can also be found in Publications 1, 2 and 4. Section 3.2 and Section 3.3 report a Product-of-Exponential (POE) error modeling method and a differential-evolution (DE) parameter identification method which can also be found in Publications 5 and 7.

3.1 A Denavit-Hartenberg Hybrid Error Model for a Serial-Parallel Hybrid Robot

The Denavit-Hartenberg (DH) modeling method and the modified Denavit-Hartenberg modeling method are commonly used for the calibration of serial robots [8]. However, for a parallel robot connected by spherical and universal joints, the DH model would not be a suitable modeling method. The vector chain modeling method for the inverse kinematics of a parallel robot is the most popular solution [59][60]. In this section, a hybrid modeling method is proposed. The combination of the DH model and the vector chain model can be used for the hybrid robot serially connected by serial and parallel mechanisms.

3.1.1 The kinematic model

Given two consecutive link frames, F_{i-1} and F_i, on a robot manipulator, frame F_i will be uniquely determined from frame F_{i-1} using parameters d_i, a_i, α_i, and θ_i in Figure 4. The DH parameters can be established according to the following rules [20]:

- The Z vector of any link frame is always on a joint axis. The only exception to this rule is for the robot end-effector (tool) with no joint axis.
- Let d_i be the joint distance from the origin of the coordinate system i-1 to the intersection of Z_{i-1} axis and X_i-axis along Z_{i-1}-axis. Then d_i is variable for the prismatic joint and constant for the revolute joint.
- The link length a_i is defined as the common perpendicular of axes Z_{i-1} and Z_i.
- Let θ_i be the rotated angle from X_{i-1}-axis to X_i-axis about Z_{i-1}-axis. Then θ_i is variable for the revolute joint and constant for the prismatic joint.
- The twist angle α_i is defined as the rotation from Z_{i-1}-axis to Z_i-axis about X_i-axis.

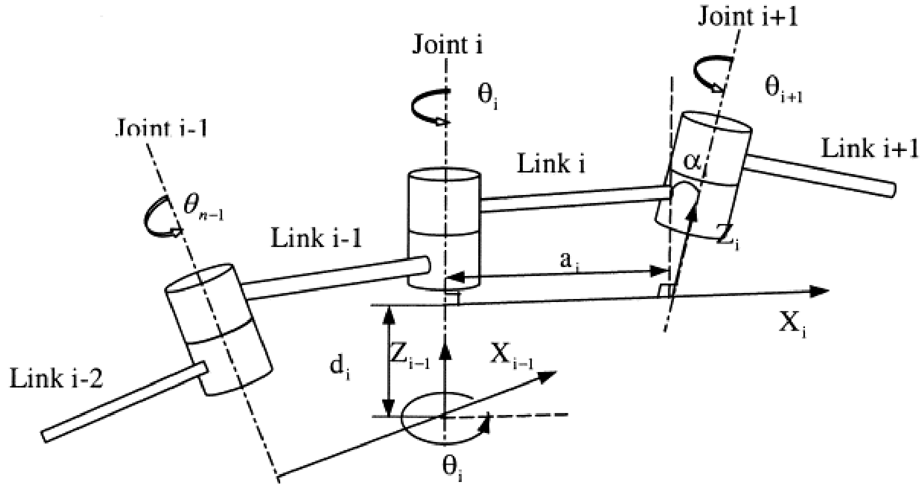


Figure 4. DH convention for the robot link coordinate system [61].

Based on the above DH parameter convention, the coordinates of the 4-DOF serial mechanism for the 10-DOF hybrid robot can be established as shown in Figure 5, and the corresponding kinematic parameters are listed in Table 1. In what follows, the 4-DOF serial mechanism is named as carriage.

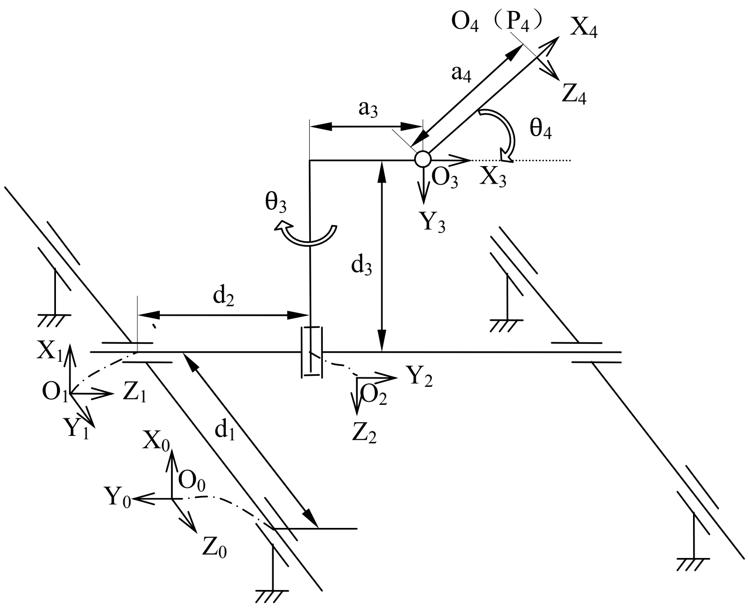


Figure 5. Coordinate system of the carriage.

Table 1. DH parameters for the carriage

Link No.	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	d_1(variable)	0
2	$\pi/2$	0	d_2(variable)	$\pi/2$
3	$\pi/2$	a_3	d_3	θ_3(variable)
4	$-\pi/2$	a_4	0	θ_4(variable)

Substituting the above DH link parameters into Equation (6), we can obtain the DH homogeneous transformation matrices ${}^0\mathbf{A}_1$, ${}^1\mathbf{A}_2$, ${}^2\mathbf{A}_3$, ${}^3\mathbf{A}_4$ and nominal forward kinematics of the carriage ${}^0T_4$ by

$$ {}^{i-1}\mathbf{A}_i = \begin{bmatrix} c\theta_i & -c\alpha_i s\theta_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & c\alpha_i c\theta_i & -s\alpha_i c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{6} $$

$$ {}^0\mathbf{T}_4 = {}^0\mathbf{A}_1 \cdot {}^1\mathbf{A}_2 \cdot {}^2\mathbf{A}_3 \cdot {}^3\mathbf{A}_4 $$
$$ = \begin{bmatrix} s\theta_4 & 0 & c\theta_4 & a_1 + d_3 + a_4 s\theta_4 \\ -s\theta_3 c\theta_4 & -c\theta_3 & s\theta_3 s\theta_4 & -d_2 - a_3 s\theta_3 - a_4 s\theta_3 c\theta_4 \\ c\theta_3 c\theta_4 & -s\theta_3 & -c\theta_3 s\theta_4 & d_1 + a_3 c\theta_3 + a_4 c\theta_3 c\theta_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^0\mathbf{R}_4 & {}^0\mathbf{P}_4 \\ 0 & 1 \end{bmatrix}, \tag{7} $$

where sine and cosine are abbreviated as *s* and *c*, and the same abbreviations will also be adopted in the following sections.

A schematic diagram of the hexapod parallel mechanism is shown in Figure 6. Two Cartesian coordinate systems, frame {4}, and frame {5}, are attached to the connecting platform and the moving platform respectively. Six actuated legs are connected to the connecting platform by universal joints and to the moving platform by spherical joints. In the following, we denote this water-hydraulic-actuated hexapod parallel manipulator as Hexa-WH.

For nominal kinematic parameters of Hexa-WH, let $\mathbf{l}_i$ be the unit vector of the direction from A_i to B_i, and l_i be the magnitude. Then the inverse kinematics of leg *i* for the hexapod parallel manipulator [62][63] can be expressed by the following vector-loop equation

$$ l_i \mathbf{l}_i = {}^4\mathbf{P}_5 + {}^4\mathbf{R}_5 \, {}^5\mathbf{b}_i - {}^4\mathbf{a}_i, \qquad i = 1,2,...6 \ , \tag{8} $$

where ${}^4\mathbf{P}_5$ is the position vector of the moving platform frame {5} with respect to the connecting platform frame {4}; ${}^4\mathbf{a}_i$ and ${}^5\mathbf{b}_i$ are the coordinate vectors of the universal joint A_i in frame {4} and the spherical joint B_i in frame {5}; ${}^4\mathbf{R}_5$ is the Z-Y-X Euler transformation matrix which represents the orientation of frame {5} with respect to frame {4}

$$
{}^4\mathbf{R}_5 = \begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\lambda - s\alpha c\lambda & c\alpha s\beta c\lambda + s\alpha s\lambda \\ s\alpha c\beta & s\alpha s\beta s\lambda + c\alpha c\lambda & s\alpha s\beta c\lambda + c\alpha s\lambda \\ -s\beta & c\beta s\lambda & c\beta c\lambda \end{bmatrix} . \tag{9}
$$

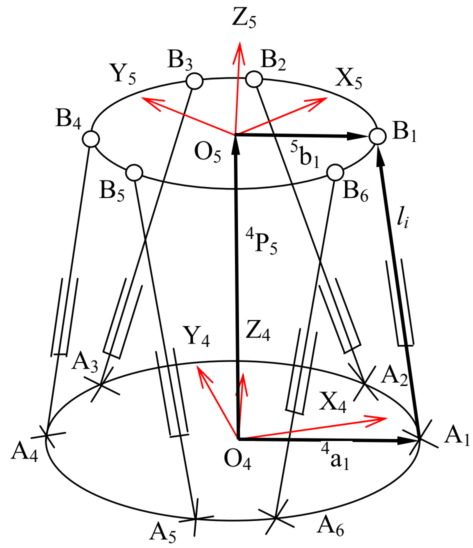


Figure 6. Coordinate system of Hexa-WH parallel manipulator.

The schematic diagram of the redundant serial-parallel hybrid manipulator, as shown in Figure 7, can be obtained by combining the carriage and the Hexa-WH mechanisms together. The coordinate frame {4} of Hexa-WH is coincident with the end-tip frame of the carriage. The fixed reference frame {0} is placed at the left rail of the carriage. For this hybrid structure, a hybrid model can be expressed as a vector-loop equation

$$
\begin{aligned}
{}^0\boldsymbol{P}_5 &= {}^0\boldsymbol{P}_4 + {}^0\boldsymbol{R}_4 \cdot {}^4\boldsymbol{P}_5 = {}^0\boldsymbol{P}_4 + {}^0\boldsymbol{R}_4\left(l_i \boldsymbol{l}_i + {}^4\boldsymbol{a}_i - {}^4\boldsymbol{R}_5 \cdot {}^5\boldsymbol{b}_i\right) \\
&= {}^0\boldsymbol{P}_4 + {}^0\boldsymbol{R}_4 \cdot l_i \boldsymbol{l}_i + {}^0\boldsymbol{R}_4 \cdot {}^4\boldsymbol{a}_i - {}^0\boldsymbol{R}_5 \cdot {}^5\boldsymbol{b}_i ,
\end{aligned} \tag{10}
$$

From Equation (10), the inverse solution of the hybrid robot, i.e., the nominal leg lengths can be derived as

$$
l_i \boldsymbol{l_i} = \left({}^0\boldsymbol{R_4} \right)^{-1} \left({}^0\boldsymbol{P_5} - {}^0\boldsymbol{P_4} - {}^0\boldsymbol{R_4} \cdot {}^4\boldsymbol{a_i} + {}^0\boldsymbol{R_5} \cdot {}^5\boldsymbol{b_i} ,\ \ \mathrm{i} = 1, 2, \cdots, 6 \right) , \tag{11}
$$

where ${}^0\mathbf{R}_5$ and ${}^0\mathbf{P}_5$ are the orientation matrix and the position vector of the end-effector frame {5} with respect to the fixed reference frame {0}.

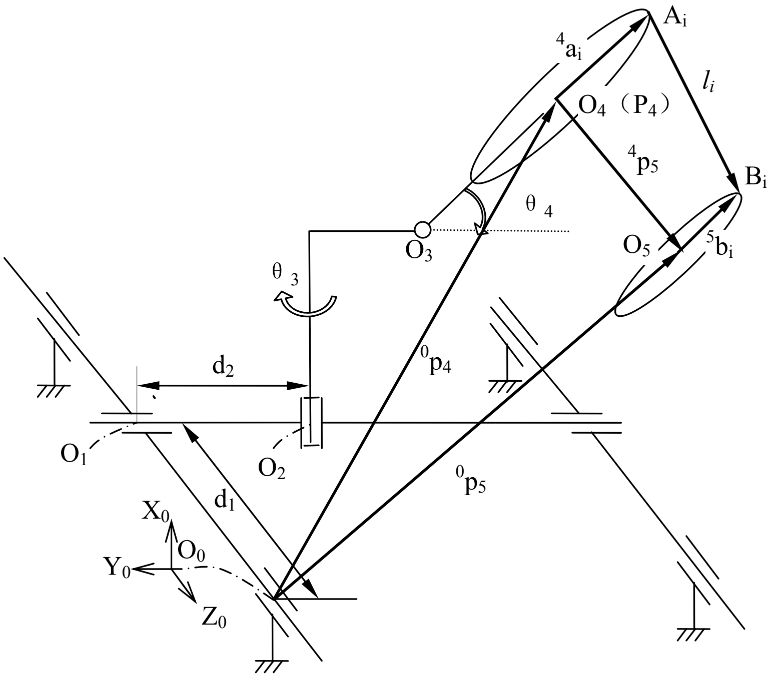


Figure 7. Schematic diagram of the hybrid IWR robot.

3.1.2 Error model

According to the approaches proposed by Veitschegger and Wu [22], a differential change $d^{i-1}\mathbf{A}_i$ between two successive joint frames will result if small errors occur in the DH parameters θ_i, d_i, a_i and α_i, and the predicted relationship between two consecutive joint frames can be expressed as

$$ {}^{i-1}\mathbf{A}_i^p = {}^{i-1}\mathbf{A}_i + d^{i-1}\mathbf{A}_i \, , \tag{12} $$

Where ${}^{i-1}\mathbf{A}_i$, the homogeneous transformation matrix containing four nominal DH link parameters, can express the nominal relationship between the consecutive joint frames i and (i-1); $d^{i-1}A_i$, the differential changes resulting from the link parameter errors and the actuator joint offset errors, can be approximated as a linear function by the first order Taylor's series

$$ d^{i-1}\mathbf{A}_i = \frac{\partial^{i-1}\mathbf{A}_i}{\partial \theta_i}\delta\theta_i + \frac{\partial^{i-1}\mathbf{A}_i}{\partial d_i}\delta d_i + \frac{\partial^{i-1}\mathbf{A}_i}{\partial a_i}\delta a_i + \frac{\partial^{i-1}\mathbf{A}_i}{\partial \alpha_i}\delta\alpha_i \, , \tag{13} $$

where $\delta\theta_i, \delta d_i, \delta a_i$ and $\delta\alpha_i$ are the small DH parameter errors; and $\frac{\partial^{i-1}A_i}{\partial\theta_i}$, $\frac{\partial^{i-1}A_i}{\partial d_i}, \frac{\partial^{i-1}A_i}{\partial a_i}$ and $\frac{\partial^{i-1}A_i}{\partial\alpha_i}$ are the partial derivatives calculated by nominal geometrical link parameters. From Equation (6), taking the partial derivative of ${}^{i-1}\mathbf{A}_i$ with respect to θ_i, d_i, a_i and α_i respectively, we can obtain

$$\frac{\partial\, {}^{i-1}\mathbf{A}_i}{\partial \theta_i} = \begin{bmatrix} -s\theta_i & -c\theta_i\, c\alpha_i & c\theta_i\, s\alpha_i & -a_i s\theta_i \\ c\theta_i & -s\theta_i\, c\alpha_i & s\theta_i\, s\alpha_i & a_i c\theta_i \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{14}$$

$$\frac{\partial\, {}^{i-1}\mathbf{A}_i}{\partial d_i} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{15}$$

$$\frac{\partial\, {}^{i-1}\mathbf{A}_i}{\partial a_i} = \begin{bmatrix} 0 & 0 & 0 & c\theta_i \\ 0 & 0 & 0 & s\theta_i \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{16}$$

$$\frac{\partial\, {}^{i-1}\mathbf{A}_i}{\partial \alpha_i} = \begin{bmatrix} 0 & s\alpha_i s\theta_i & c\alpha_i s\theta_i & 0 \\ 0 & -s\alpha_i c\theta_i & -c\alpha_i c\theta_i & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{17}$$

Let $\mathrm{d}\,{}^{i-1}\mathbf{A}_i = {}^{i-1}\mathbf{A}_i \delta\, {}^{i-1}\mathbf{A}_i$, then

$$\delta\, {}^{i-1}\mathbf{A}_i = \mathrm{D}_{\theta_i}\delta\theta_i + \mathrm{D}_{d_i}\delta d_i + \mathrm{D}_{a_i}\delta a_i + \mathrm{D}_{\alpha_i}\delta\alpha_i, \tag{18}$$

where $\mathbf{D}_{\theta_i}, \mathbf{D}_{d_i}, \mathbf{D}_{a_i}, \mathbf{D}_{\alpha_i}$ can be solved as follows:

$$\mathbf{D}_{\theta_i} = \left({}^{i-1}\mathbf{A}_i\right)^{-1} \cdot \frac{\partial\, {}^{i-1}\mathbf{A}_i}{\partial \theta_i} = \begin{bmatrix} 0 & -c\alpha_i & s\alpha_i & 0 \\ c\alpha_i & 0 & 0 & a_i.\, c\alpha_i \\ -s\alpha_i & 0 & 0 & -a_i.\, s\alpha_i \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{19}$$

$$\mathbf{D}_{d_i} = \left({}^{i-1}\mathbf{A}_i\right)^{-1} \cdot \frac{\partial\, {}^{i-1}\mathbf{A}_i}{\partial d_i} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & s\alpha_i \\ 0 & 0 & 0 & c\alpha_i \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{20}$$

$$\mathbf{D}_{a_i} = \left({}^{i-1}\mathbf{A}_i\right)^{-1} \cdot \frac{\partial\, {}^{i-1}\mathbf{A}_i}{\partial a_i} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{21}$$

$$\mathbf{D}_{\alpha_i} = \left({}^{i-1}\mathbf{A}_i\right)^{-1} \cdot \frac{\partial\, {}^{i-1}\mathbf{A}_i}{\partial \alpha_i} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{22}$$

Substituting Equations (19) through (22) into Equation (18) and expanding it into a matrix form we can obtain

$$\delta\, {}^{i-1}\mathbf{A}_i = \begin{bmatrix} 0 & -c\alpha_i\delta\theta_i & s\alpha_i\delta\theta_i & \delta a_i \\ c\alpha_i\delta\theta_i & 0 & -\delta\alpha_i & a_i c\alpha_i\delta\theta_i + s\alpha_i\delta d_i \\ -s\alpha_i\delta\theta_i & \delta\alpha_i & 0 & -a_i s\alpha_i\delta\theta_i + c\alpha_i\delta d_i \\ 0 & & 0 & 0 \end{bmatrix}. \tag{23}$$

The above expression gives the general differential translation and rotation errors for joints which are not parallel or nearly parallel. In the case of the 4-DOF carriage, the predicted forward solution with kinematic errors can be expressed as

$$ {}^{0}\mathbf{T}_4^p = {}^{0}\mathbf{T}_4 + d\,{}^{0}\mathbf{T}_4 = \prod_{i=1}^{4}\left({}^{i-1}\mathbf{A}_i + d\,{}^{i-1}\mathbf{A}_i\right) = \begin{bmatrix} {}^{0}\mathbf{R}_4^p & {}^{0}\mathbf{P}_4^p \\ 0 & 1 \end{bmatrix}. \tag{24}$$

Expanding Equation (24), we can get the first-order, second-order and higher-order differential products. The work conducted by Veitschegger and Wu [22] concluded that the first-order model is sufficiently accurate for most applications. As the size of the manipulator structure or the size of the input kinematic errors increases, the effect of the second-order error terms increases. Whether or not the first-order model is adequate will always depend on the manipulator size, configuration, input kinematic errors, and the required accuracy of the model. If the second- and higher-order differential errors are not considered, the relationship between the differential change in the carriage end-tip point and the change in link parameters can be expressed as

$$d\,{}^{o}\mathbf{T}_4 = \delta\mathbf{T}^1 \cdot {}^{0}\mathbf{T}_4\,, \quad \delta\mathbf{T}^1 = \sum_{i=1}^{4}\left(\left[{}^{0}\mathbf{A}_i\right] \cdot \delta\, {}^{i-1}\mathbf{A}_i \cdot \left[{}^{0}\mathbf{A}_i\right]^{-1}\right), \tag{25}$$

where $\delta\mathbf{T}^1$ is the first-order error transformation matrix in the fixed reference frame. According to Paul's work [20], the first-order error transformation matrix has the following matrix structure, although values of their elements are in general different

$$\delta\mathbf{T}^1 = \begin{bmatrix} 0 & -\delta\theta_z & \delta\theta_y & \delta d_x \\ \delta\theta_z & 0 & -\delta\theta_x & \delta d_y \\ -\delta\theta_y & \delta\theta_x & 0 & \delta d_z \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{26}$$

where $[\delta d_x, \delta d_y, \delta d_z]^{\mathrm{T}}$ are the translational errors and $[\delta\theta_x, \delta\theta_y, \delta\theta_z]^{\mathrm{T}}$ are the rotational errors.

From Equation (24), the predicted orientation matrix ${}^{0}\mathbf{R}_4^{\mathrm{p}}$ and position vector ${}^{0}\mathbf{P}_4^{\mathrm{p}}$ of frame {4} with respect to frame {0} can be formulated, and the unknown parameter errors $\delta\theta_i$, δd_i, δa_i and $\delta\alpha_i$ will be taken as variables in the final objective function. The DH convention from Paul shows that: for a revolute joint whose axis Z_i is a line in space, all four parameter errors, including the kinematic parameters and the joint offset errors, have to be calibrated; for a prismatic joint whose Z_i is a free vector, only two parameters that describe its orientation ($\delta\alpha_i$ and $\delta\theta_i$) are required, and the other two must be set to be zero. Since the carriage consists of two prismatic joints and two revolute joints, the number of parameter errors for the serial part is 12.

For the Hexa-WH parallel manipulator, when the manufacturing and assembling errors are introduced, different error models can be derived based on a different error modeling method. For instance, Wang and Masory [64] employed the DH modeling method to develop an error model where the universal joint is replaced by two revolute joints and the spherical joint is replaced by three revolute joints; then the problem of error modeling for the 6-UPS mechanism is transferred to that of error modeling for the 6-RRPRRR mechanism. By using this configuration, 22 parameter errors can be obtained in each joint-link train. Another modeling method used for the hexapod parallel manipulator is the vector chain modeling method. The applications of this method can be found in the literature [65][66][67]. With this method, the universal joint and the spherical joint can be simplified as a coordinate point; thus the consideration of joint axis misalignments of the universal joint is unnecessary. Denoting the coordinate deviations between the real coordinate values (${}^{4}\mathbf{a}_i^{\mathrm{r}}$, ${}^{5}\mathbf{b}_i^{\mathrm{r}}$) and their nominal values (${}^{4}\mathbf{a}_i$, ${}^{5}\mathbf{b}_i$) as $\delta^{4}a_i$ and $\delta^{5}b_i$, and the leg offset error as δl_i, then the error model of the Hexa-WH can be written as

$$l_{\mathrm{i}}^{\mathrm{p}} = (l_{\mathrm{i}} + \delta l_{\mathrm{i}})\mathbf{l}_{\mathrm{i}}^{\mathrm{p}} = {}^{4}\mathbf{P}_5^{\mathrm{m}} + {}^{4}\mathbf{R}_5^{\mathrm{m}}\left({}^{5}\mathbf{b}_{\mathrm{i}} + \delta\, {}^{5}\mathbf{b}_{\mathrm{i}} \right) - \left({}^{4}\mathbf{a}_{\mathrm{i}} + \delta\, {}^{4}\mathbf{a}_{\mathrm{i}} \right),\ \mathrm{i} = 1,2,\cdots,6\,. \tag{27}$$

Accordingly, we have seven parameter errors in each leg: three coordinate parameter errors for joint A_i, three coordinate parameter errors for joint B_i, and one error parameter for leg joint offset. Thus, the number of parameter errors for the Hexa-WH is 42.

Integrating the serial and parallel error model together, the final hybrid error model for the hybrid robot can be obtained as

$${}^{0}\mathbf{P}_5^{\mathrm{m}} = {}^{0}\mathbf{P}_4^{\mathrm{p}} + {}^{0}\mathbf{R}_4^{\mathrm{p}}\, {}^{4}\mathbf{P}_5^{\mathrm{m}} = {}^{0}\mathbf{P}_4^{\mathrm{p}} + {}^{0}\mathbf{R}_4^{\mathrm{p}}\left[(l_i + \delta l_i)\mathbf{l}_{\mathrm{i}}^{p} + \left({}^{4}\mathbf{a}_i + \delta^{4}\mathbf{a}_i\right) - {}^{4}\mathbf{R}_5^{\mathrm{m}}\left({}^{5}\mathbf{b}_i + \delta^{5}\mathbf{b}_i\right)\right]. \tag{28}$$

From Equation (28), the predicted leg lengths can be rewritten as

$$\begin{aligned} l_{\mathrm{i}}^{\mathrm{p}} &= (l_{\mathrm{i}} + \delta l_{\mathrm{i}})\mathbf{l}_{\mathrm{i}}^{\mathrm{p}} \\ &= \left({}^{0}\mathbf{R}_4^{\mathrm{p}} \right)^{-1}\left[{}^{0}\mathbf{P}_5^{\mathrm{m}} - {}^{0}\mathbf{P}_4^{\mathrm{p}} - {}^{0}\mathbf{R}_4^{\mathrm{p}}\left({}^{4}\mathbf{a}_{\mathrm{i}} + \delta\, {}^{4}\mathbf{a}_{\mathrm{i}} \right) + {}^{0}\mathbf{R}_5^{\mathrm{m}}\left({}^{5}\mathbf{b}_{\mathrm{i}} + \delta\, {}^{5}\mathbf{b}_{\mathrm{i}} \right)\right], \end{aligned} \tag{29}$$

where, ${}^{0}\boldsymbol{P}_5^{m}$ and ${}^{0}\boldsymbol{R}_5^{m}$ denote the measured position vector and orientation matrix of the end-effector, whose values can be obtained via the accurate measurement instrument; ${}^{0}\boldsymbol{P}_4^{p}$ and ${}^{0}\boldsymbol{R}_4^{p}$ denote the carriage end-tip position vector and orientation matrix predicted by the model.

3.1.3 Nonlinear identification model

The purpose of the parameter identification process is to find a vector of parameter errors to improve the kinematic model's accuracy. To accomplish this, a linear or nonlinear least-square objective function has to be constructed. For the proposed hybrid error model, the error residuals between the measured leg length l_i^m and the predicted leg length l_i^p in Equation (29) can be adopted to formulate an objective function based on the Euclidean norm. Supposing a set of measurement data has been collected, the task of the identification algorithm is to find a suitable combination of 54 parameter errors (variables)

$$\boldsymbol{\theta} = [\delta\theta_1, \delta\alpha_1, \delta\theta_2, \delta\alpha_2, \delta\theta_3, \delta\alpha_3, \delta d_3, \delta a_3, \delta\theta_4, \delta\alpha_4, \delta d_4, \delta a_4, \delta a_{ix}, \delta a_{iy}, \delta a_{iz}, \delta b_{ix}, \delta b_{iy}, \delta b_{iz}, \delta l_i]^{\mathrm{T}} \quad (i = 1, \cdots, 6),$$

to minimize the objective function

$$\mathrm{SS}_{\boldsymbol{\theta}} = \sum_{j=1}^{N} \sum_{i=1}^{6} (l_{i,j}^m - l_{i,j}^p)^2 \tag{30}$$

In Equation (30), N is the number of measurement configurations; $l_{i,j}^p$ is the model predicted leg length at the measurement location j for leg i, whereas $l_{i,j}^m$ is the measured leg length at the measurement location j for leg i; $\boldsymbol{\theta}$ is the vector of parameter errors (variables) in the hybrid error model. The total number of these variables is 54, of which 12 are from the carriage while the remaining 42 variables are from the Hexa-WH parallel manipulator.

3.2 The Product-of-Exponential Error Model for the Serial-parallel Hybrid Robot

The product-of-exponential (POE) error modeling method was originally developed for the calibration of a serial robot connected by a revolute and prismatic joint [68]. In the following two sections, we will demonstrate that the POE modeling method can also be used for the error modeling of redundant serial-parallel hybrid robot. The mathematic background of the POE modeling method can be found in Appendix A.

3.2.1 Kinematic model

Unlike the Denavit-Hartenberg modeling method, there are only two frames, the base frame {S} and the tool frame {T}, that are needed for the POE model. Furthermore, the reference configuration of the POE model is usually chosen to make the kinematic analysis as simple as possible since any configuration of the manipulator can be defined as a reference configuration. Based on the POE conventions, the schematic of the hybrid IWR robot in its reference configuration can be established (Figure 8). In this reference configuration, the base frame {S} and the tool frame {T} coincide with each other on the end-effector. Parameters q_1, q_2, q_3, q_4 and d_i (i=1, 2, ⋯, 6) represent the actuated-joint displacements; $\boldsymbol{\xi}_{s1}$, $\boldsymbol{\xi}_{s2}$, $\boldsymbol{\xi}_{s3}$, $\boldsymbol{\xi}_{s4}$ and $\boldsymbol{\xi}_{pi}$ ($i = 1,2,\cdots,6$) represent the joint twist of serial and parallel mechanisms; points $\mathbf{p}_3$ and $\mathbf{p}_4$ represent the arbitrarily selected points on the corresponding joint axis, which can be used to calculate the position of the axis with respect to the origin of the base frame {S}; l_0, l_1, l_2, l_3 and l_4 represent the link lengths.

Due to the redundant structure, the inverse solution of the hybrid robot can have an infinite number of joint configurations for the same given end-effector pose. However, if the forward solution of the serial mechanism has been decided, then the inverse solution of the parallel mechanism can be easily obtained. Therefore, in this section, the forward kinematics of the 4-DOF serial mechanism will be derived first, and then its predicted solution can be integrated into the inverse kinematics of the 6-DOF hexapod parallel manipulator to obtain the hybrid model.

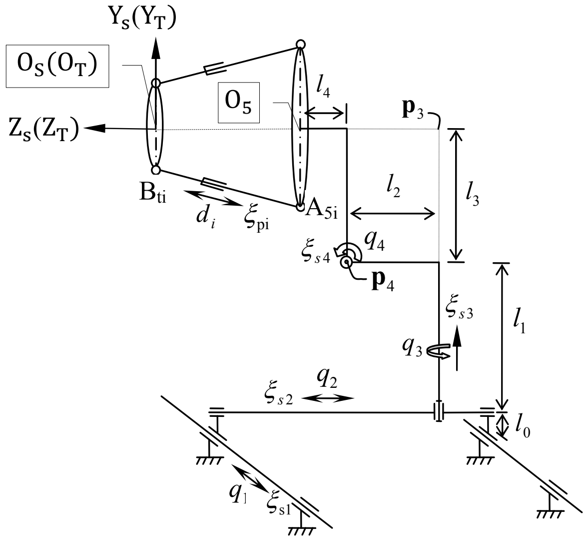


Figure 8. Schematic diagram of the IWR robot in its reference configuration.

Based on the serial-parallel hybrid structure and the POE formula, the forward kinematics of the carriage, i.e. the pose of the connecting platform frame {5} in terms of the base frame {S}, can be expressed as

$$g_{s5}(\mathbf{q}) = e^{\hat{\xi}_{s1} \cdot q_1} \cdot e^{\hat{\xi}_{s2} \cdot q_2} \cdot e^{\hat{\xi}_{s3} \cdot q_3} \cdot e^{\hat{\xi}_{s4} \cdot q_4} \cdot g_{s5}(\mathbf{0}) \tag{31}$$

where $g_{s5}(0)$ represents the transformation matrix of platform frame {5} with respect to base frame {S} in the reference configuration where the input joint displacements $\mathbf{q}$=0.

The inverse solution of the hexapod platform is quite simple and obvious from the geometry of the manipulator. Let $\mathbf{a}_{si} \in \mathcal{R}^3$, $\mathbf{b}_{si} \in \mathcal{R}^3$ be the position vector of points A_i and B_i with respect to the base frame {S}; let $\mathbf{a}_{5i} \in \mathcal{R}^3$ be the position vector of point A_i with respect to the platform frame {5}; let $\mathbf{b}_{ti} \in \mathcal{R}^3$ be the position vector of point B_i with respect to the tool frame {T}. Then the extension of the prismatic joints, i.e. the leg lengths of the Hexa-WH, can be obtained

$$d_i = \|\mathbf{b}_{si} - \mathbf{a}_{si}\| = \|g_{st} \cdot \mathbf{b}_{ti} - g_{s5}(\mathbf{q}) \cdot \mathbf{a}_{5i}\|, \ i = 1, 2, \cdots, 6, \tag{32}$$

where g_{st} is the desired pose configuration of tool frame {T} with respect to base frame {S}; and $g_{s5}(\mathbf{q})$ is the forward solution of the carriage.

3.2.2 The error model

In practice, the manufacturing and assembling errors are unavoidable. For instance, the actual leg length would have a joint offset error, the real locations of points A_i and B_i would never agree with the designed ones, and the twist of the serial mechanism would also have some deviations. Taking these kinematic parameter errors into consideration, the error model of the hybrid robot can be written as

$$d_i^p = d_i + \delta d_i = \left\| \mathbf{b}_{si}^p - \mathbf{a}_{si}^p \right\|, \qquad i = 1, 2, \cdots, 6, \tag{33}$$

where δd_i is the leg joint offset; $\mathbf{b}_{si}^p = \mathrm{g}_{st}^m \cdot (\mathbf{b}_{ti} + \delta \mathbf{b}_{ti})$; g_{st}^m is the measured pose of the end-effector frame {T} with respect to base frame {S}; $\mathbf{a}_{si}^p$ is the predicted position vector of point A_i with respect to base frame {S}, which can be expressed as

$$\begin{aligned} \mathbf{a}_{si}^p &= \mathbf{a}_{si} + \delta \mathbf{a}_{si} = \mathbf{a}_{si} + \delta \mathrm{g}_{s5}(\boldsymbol{q}) \cdot \mathbf{a}_{5i} + \mathrm{g}_{s5}(\boldsymbol{q}) \cdot \delta \mathbf{a}_{5i} \\ &= \mathbf{a}_{si} + \left(\delta \mathrm{g}_{s5}(\boldsymbol{q}) \cdot \mathrm{g}_{s5}(\boldsymbol{q})^{-1}\right) \cdot \mathbf{a}_{5i} + \mathrm{g}_{s5}(\boldsymbol{q}) \cdot \delta \mathbf{a}_{5i}, \qquad i = 1, 2, \cdots, 6, \end{aligned} \tag{34}$$

where the transformation error matrix $\delta \mathrm{g}_{s5}(\boldsymbol{q}) \cdot \mathrm{g}_{s5}(\boldsymbol{q})^{-1}$ can be calculated according to Equation (A.12).

According to the identifiability anylysis by He [36], the maximum number of identifiable parameters in a serial robot with *r* number of revolute joints and *t* number of prismatic joints is *6r+3t+6*. Since we have two prismatic joints and two revolute joints while there is no pose measurement from the tip-point of the carriage, the number of independent identifiable parameters resulted from the carriage is 18. Furthermore, each location of the spherical joints A_i and B_i is affected by three coordinate parameter errors, and each leg is affected by one joint offset error. Thus a vector of 42 parameter errors would result from the 6-DOF Hexa-WH parallel manipulator. Note that the parameter errors can also be reduced from 42 to 30 by attaching the upper platform frame {T} and the lower platform frame {5} of the Hexa-WH to the joints B_{t1} and A_{51} respectively [69], but this rearrangement would increase the complexity of the hybrid model and would not satisfy the completeness requirement as a good model.

3.2.3 Nonlinear identification model

Based on the error model Equation (33), a nonlinear objective function can be formulated. The idea of this nonlinear optimization algorithm is to minimize the deviations between the measured and the predicted leg lengths based on the Euclidean norm. Supposing the number of measured manipulator locations is *N*, then the task of the identification algorithm is to find a suitable combination of the 60 parameter errors (variables)

$$\boldsymbol{\theta} = (\delta\boldsymbol{\xi}_{s1},\ \delta\boldsymbol{\xi}_{s2},\ \delta\boldsymbol{\xi}_{s3},\ \delta\boldsymbol{\xi}_{s4},\ \delta d_i,\ \delta\mathbf{a}_{5i},\ \delta\mathbf{b}_{ti},),\ i=1, 2,\cdots,6\ , \tag{35}$$

to minimize

$$\mathrm{SS}_\theta = \sum_{j=1}^{N} \sum_{i=1}^{6} \left(d_{i,j}^m - d_{i,j}^p\right)^2, \tag{36}$$

where *N* is the number of measurement configurations; $d_{i,j}^m$ is the measured leg length and $d_{i,j}^p$ is the predicted leg length at the measurement location *j* for leg *i*.

3.3 Differential-Evolution Based Parameter Identification Algorithms

This section gives a basic introduction to the global optimization algorithm, namely the differential-evolution algorithm (DE). For more details, please refer to [53]. DE is tailored for minimizing real-valued, multi-modal, and nonlinear objective functions. It belongs to a class of evolutionary computation algorithms, which utilize mutation, crossover and selection operations to mimic the evolutionary process of the real world. According to the DE algorithm, the parameter errors in our developed error models can be represented as an individual vector $\boldsymbol{\theta}=(\theta_1, \theta_2, \cdots, \theta_D)$, where D is the individual index of the parameter errors. The population in each generation G can be represented as a matrix $\boldsymbol{\Theta}_{i,\mathrm{G}} \in \mathcal{R}^{D\times NP}$, where $i=1,2, \cdots, NP$ is the population index defined by user. The flowchart of the DE algorithm is shown in Figure 9. The detailed parameter identification procedures of DE are discussed below.

1) Initialization

To start a DE optimization process, an initial population as a starting point must be created. One way to generate the initial population is to assign a random value for each parameter within its feasible boundaries

$$\theta_{j,i,G=1} = \theta^{L}_{j,i} + rand_j(0,1)\cdot(\theta^{U}_{j,i} - \theta^{L}_{j,i}), \tag{37}$$

where $j = 1,2,\cdots,D$ is the individual index for parameter errors; $i = 1,2,\cdots,NP$ is the population index; and $\theta^L_{j,i}$ and $\theta^U_{j,i}$ are the lower and upper boundaries of the parameter j respectively. After initialization, the population evolves with the operations of mutation, crossover, and selection.

2) Mutation

The main objective of the mutation operation is to keep a population robust and search for new territory. In the step of the DE mutation operation, the new parameter vectors are generated by adding a weighted difference vector between two different population members to the third member. For each vector $\boldsymbol{\theta}_{i,G}$, a mutant vector $\mathbf{m}_{i,G+1}$ is generated according to the formula of

$$\mathbf{m}_{i,G+1} = \boldsymbol{\theta}_{r1,G} + F\cdot(\boldsymbol{\theta}_{r2,G} - \boldsymbol{\theta}_{r3,G})\,. \tag{38}$$

The randomly selected integers have to satisfy the requirement of $r_1, r_2, r_3 \in \{1, 2, \cdots, NP\}$ and $r_1 \neq r_2 \neq r_3 \neq i$. The mutation scale factor $F > 0$.

3) Crossover

The aim of the crossover operation is to increase the diversity of the generated vectors. The trial vector is generated as follows

$$\begin{aligned} &\mathbf{u}_{i,G+1} = (u_{1,i,G+1}, u_{2,i,G+1}, \cdots, u_{D,i,G+1}), \\ &u_{j,i,G+1} = \begin{cases} m_{j,i,G+1}, & \text{if}(rand_j[0,1] < CR \quad or \quad j = j_r), \\ \theta_{j,i,G}, & \text{otherwise}, \end{cases} \end{aligned} \tag{39}$$

where $G = 1,2,\cdots,G_{max}$ is the generation index for maximum evolutionary generations; CR is a crossover rate in range [0, 1]; j_r is a random value chosen from the set $\{1,2,\cdots,D\}$. The use of j_r is to ensure that trail vector $\mathbf{u}_{i,G+1}$ can get at least one parameter from mutant vector $\mathbf{m}_{i,G+1}$.

4) **Selection**

In the selection operation of DE, trial vector $\mathbf{u}_{i,G+1}$ is compared to target vector $\boldsymbol{\theta}_{i,G}$ by evaluating the objective function values to determine whether the trial vector can become a member of the next generation. The vector, which has a smaller objective function value, is allowed to evolve to the next generation, i.e.

$$\boldsymbol{\theta}_{i,G+1} = \begin{cases} \mathbf{u}_{i,G+1}, \text{if } f(\mathbf{u}_{i,G+1}) \le f(\boldsymbol{\theta}_{i,G}), \\ \boldsymbol{\theta}_{i,G}, \ \text{otherwise.} \end{cases} \tag{40}$$

This selection procedure, guarantees that all individuals of the next generation are as good as or better than the individuals of the current population.

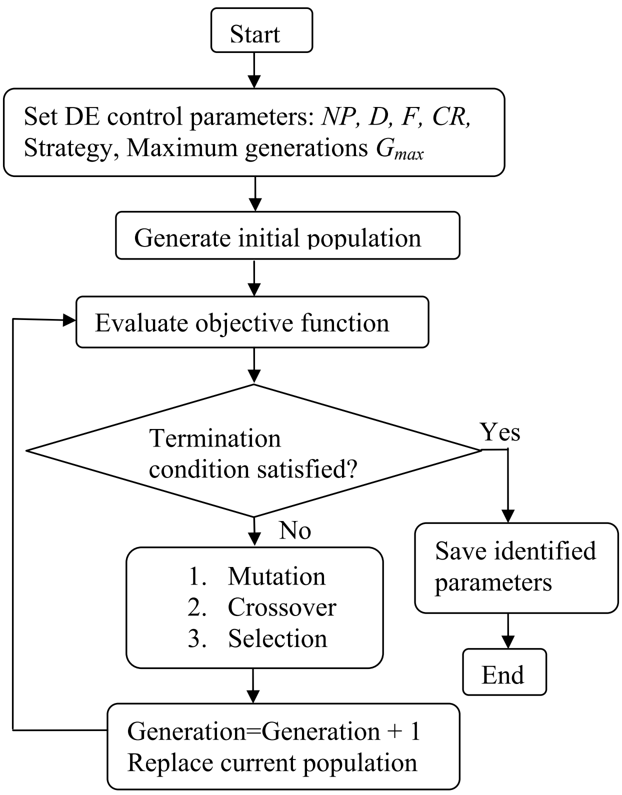


Figure 9. Flowchart of the DE algorithm.

3.4 Markov Chain Monte Carlo Parameter Identification Algorithms

This section presents a statistical parameter estimation method, the Markov Chain Monte Carlo (MCMC) approach, which basically employs the Metropolis-Hastings algorithm [70][71] and has many variants [72][73][74][75]. The basic idea of the MCMC methods is that a Markov chain can be constructed as a stationary distribution, which is also the joint posterior probability distribution of parameter errors. The parameter errors can be assigned with arbitrary initial guess and then the chain is simulated until it converges to a stationary distribution. Observations from the stationary chain can be used to estimate the joint posterior probabilities, and to analyze the identifiability of the parameter errors. Moreover, marginal distributions of parameters can also be easily obtained by monitoring the values of particular parameters in the stationary chain [76].

In terms of the nonlinear identification model in previous sections, let $\boldsymbol{\theta}$ be the vector of unknown parameter errors required of identification; let $\mathbf{y}$ be a vector of observed random variables, which defines a set of measured data. The posterior probability distribution of the parameter errors, $\boldsymbol{\theta}$, conditional on observations, $\mathbf{y}$, can be formulated as a Bayes formula

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}}, \tag{41}$$

where $p(\boldsymbol{\theta})$ is prior distribution (i.e., the probability distribution before examination of the data); $p(\mathbf{y}|\boldsymbol{\theta})$ is a likelihood function that gives the probability distribution of observations $\mathbf{y}$, given error parameter values $\boldsymbol{\theta}$. The most likely values of the parameters are those which give higher probability values for the posterior distribution $p(\boldsymbol{\theta}|\mathbf{y})$. The point estimates and the confidence intervals of $\boldsymbol{\theta}$ can be obtained from the posterior distribution in various ways. Generally, the mean values of the posterior distribution are commonly used as the point estimates of the parameter errors $\boldsymbol{\theta}$, and the α percent credible set is used as a confidence interval. Assuming independently and identically distributed Gaussian errors for n observations $\mathbf{y}_i$, we can have the likelihood function as

$$p(\boldsymbol{y}|\boldsymbol{\theta}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - f(c_i,\theta))^2}{2\sigma^2}} = \frac{1}{\left(\sqrt{2\pi\sigma^2}\right)^n} e^{-\frac{\sum_{i=1}^{n}(y_i - f(c_i,\theta))^2}{2\sigma^2}}. \tag{42}$$

Let $\mathrm{SS}_{\boldsymbol{\theta}} = \sum_{i=1}^{n}(\boldsymbol{y}_i - f(\boldsymbol{c}_i, \boldsymbol{\theta}))^2$, then the same form as the nonlinear identification model in the previous sections can be obtained; $\mathbf{y}_i$ represents the set of measured data; $f(\mathbf{c}_i, \boldsymbol{\theta})$ represents the function value predicted by the model; $\mathbf{c}_i$ represents the vector of known parameters.

In the past, Bayesian inference was largely limited to simple models because the normalizing constant requires integration over a high-dimensional space. An attractive feature of the MCMC methods is that the Markov chain only needs to calculate the ratios of the likelihood function. Therefore, the calculation of the normalizing constant ($\int p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}$) and prior distribution $p(\boldsymbol{\theta})$ in Equation (42) can be left out [77][78]. A typical MCMC algorithm employing the Metropolis rule to explore the posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{y})$ proceeds as follows:

1) **Initialize the vector of parameter errors $\boldsymbol{\theta}$**
   - Select an initial error parameter guess $\boldsymbol{\theta}_0$ by minimizing the objective function $\mathrm{SS}_{\boldsymbol{\theta}} = \sum_{i=1}^{n}(\boldsymbol{y}_i - f(\boldsymbol{c}_i, \boldsymbol{\theta}))^2$.

Generally, the initial guess can be selected arbitrarily for simple models, but for high nonlinear and/or high dimensional complex models, the arbitrarily selected initial values may slow down the simulation process or even make the simulation diverged. Therefore, the initial values should be selected as close to the actual values as possible. One way of doing this is through a suitable numerical optimization method generating a series of candidate values, but the "best" solution can only be determined by trial and error. In this work, the differential-evolution algorithm is employed to generate the initial guess for the MCMC sampling.

- Define the length of simulation chain N_{simu}.
- Generate a proposal distribution $q(\cdot\,|\,\boldsymbol{\theta}_{old})$ and set $SS_{old} = SS_{\boldsymbol{\theta}_0}$. The algorithm based on the LU-decomposition, which factorizes a covariance matrix as the product of a lower triangular matrix and an upper triangular matrix, is preferred to generate the proposal distribution since it is quite efficient in generating a large number of conditional realizations [79].

2) Simulation loop

- Generate $\boldsymbol{\theta}_{new}$ from the proposal distribution $q(\cdot\,|\,\boldsymbol{\theta}_{old})$ and compute SS_{new}.
- Calculate the acceptance probability α

$$\alpha = \min\left(1, \frac{p(\boldsymbol{\theta}_{new}|\mathbf{y})}{p(\boldsymbol{\theta}_{old}|\mathbf{y})}\right) \propto \min\left(1, \frac{p(\mathbf{y}|\boldsymbol{\theta}_{new})}{p(\mathbf{y}|\boldsymbol{\theta}_{old})}\right) \propto \min\left(1, e^{-\frac{1}{2\sigma^2}(SS_{new}-SS_{old})}\right). \tag{43}$$

- The new value is accepted if $SS_{new} < SS_{old}$ or $u < e^{-\frac{1}{2\sigma^2}(SS_{new}-SS_{old})}$, where u is a random number generated in the range of [0, 1].
- Repeat the simulation loop until N_{simu} samples have been created.

By using the DE-based parameter identification method, we can get only the point estimation of parameter errors, whereas by using the MCMC method, we can obtain both the point estimation results and the interval estimation results. Furthermore, the MCMC-based identification method can be used to analyze the correlations of parameter errors. The drawback of the MCMC-based identification method would be that the selection of initial guess is quite arbitrary; for a complex model, too big or too small initial values may lead to failure of the identification process. In addition, too large or too small proposal distributions would result in failure of the identification process. The proposal distribution should be chosen so that the 'sizes' of the proposal distribution and the target distributions suitably match [80].

CHAPTER 4

SIMULATION RESULTS FOR MODEL VALIDATIONS

This chapter presents some numerical simulation results to verify the effectiveness and robustness of the proposed error modeling and parameter identification methods. In the environment of Matlab® 7.0, two kinds of error modeling methods and two kinds of parameter identification methods are simulated on the same 10-DOF redundant serial-parallel hybrid robot as shown in Figure 7 (Section 3.1). The Markov Chain Monte Carlo (MCMC) algorithm and differential-evolution (DE) algorithm are used for the simulation of the Denavit-Hartenberg (DH) hybrid model considering parameter identifiability unknown. By using the MCMC algorithm, the mean values of posterior distribution can be used as the point estimates of parameter errors $\boldsymbol{\theta}$, and the correlations of parameter errors can also be observed from the stationary chain. From the correlation analysis, we can refine or eliminate redundant parameters to obtain a more accurate error model. Section 4.1 provides the simulation results of the DH-based hybrid model using the DE-based identification method. Section 4.2 simulates the MCMC-based method for parameter estimation and correlation analysis of the DH-based hybrid model. Section 4.3 presents the simulation results of the DE-based identification method for the POE-based model.

4.1 Denavit-Hartenberg Hybrid Model Using Differential-Evolution Identification Method

To verify the effectiveness of the identification algorithm in Section 3.1, the measurement device can be assumed to be perfect to ensure no measurement errors occur. In simulation, the open source Matlab® DE code [81] is employed. Based on the scheme of DE/rand-to-best/1 [53], the DE parameters in the simulation can be set as F=0.75, CR=0.95, D=54, NP=600, G_{max}=40,000. The simulations are implemented on a computer with an Intel® Core 2 Duo processor E8500 (3.16GHz) and a RAM (3.25 GB). Simulation procedures are as follows:

1) Randomly generate 100 end-effector measurement poses within the robot workspace to simulate the measured position (${}^{0}P_{5}^{m}$) and the measured orientation matrix (${}^{0}R_{5}^{m}$). Furthermore, the associated 100 actuated-joint displacements of the carriage are also randomly generated to simulate the measured joint displacements (Table 2). In the laboratory experimental test, the end-effector poses can be obtained by external measuring devices, and the joint displacements can be collected from the built-in sensor readings.

Table 2. Randomly generated end-effector poses and carriage-joint displacements (unit: mm for lengths and *rad.* for angles)

No.	d_1	d_2	θ_3	θ_4	${}^{0}P_{5x}^{m}$	${}^{0}P_{5y}^{m}$	${}^{0}P_{5z}^{m}$	${}^{0}\phi_{5}^{m}$	${}^{0}\theta_{5}^{m}$	${}^{0}\psi_{5}^{m}$
1	515.95	231.22	1.1084	1.0668	200.39	441.43	67.87	0.0403	0.1208	0.0719
2	229.76	205.73	1.2571	0.7626	155.3	255.94	255.75	0.2518	0.0288	0.0027
3	257.54	260.31	1.8784	0.0545	326.89	188.1	20.71	0.2294	0.2045	0.131
4	124.33	174.15	2.8636	0.1798	195.91	441.88	156.2	0.1279	0.1187	0.0108

No.	d_1	d_2	θ_3	θ_4	${}^{0}P_{5x}^{m}$	${}^{0}P_{5y}^{m}$	${}^{0}P_{5z}^{m}$	${}^{0}\phi_{5}^{m}$	${}^{0}\theta_{5}^{m}$	${}^{0}\psi_{5}^{m}$
5	310.13	187.83	0.4178	0.1204	59.66	122.88	160.88	0.1066	0.0778	0.0753
6	716.58	33.34	0.1325	0.9548	489.26	82	474.16	0.0331	0.0938	0.0947
7	710.21	64.10	1.2599	0.3667	259.12	329.3	103.67	0.2423	0.1263	0.1065
8	314.89	10.91	1.8715	0.2231	253.88	127.8	244.67	0.0015	0.1129	0.0761
9	540.36	133.36	2.8893	0.8328	162.51	78.8	222.33	0.0488	0.183	0.0141
10	201.77	97.97	0.9229	1.4613	98.63	138.3	410.42	0.0848	0.1767	0.0496
11	759.99	86.18	1.7123	0.969	431.44	149.35	8.74	0.0131	0.0018	0.0128
12	499.74	148.93	2.4557	1.0837	133.93	2.95	394.25	0.0378	0.0207	0.1274
13	165.55	54.51	1.0286	0.3682	269.34	114.32	270.39	0.1909	0.1206	0.0605
14	87.72	280.13	0.2993	0.3249	275.73	335.05	22.73	0.1263	0.2036	0.1138
15	452.67	282.05	1.3876	0.0587	481.62	490.83	256.92	0.0885	0.2138	0.0841
16	219.46	177.43	0.0913	1.042	134.78	128.07	222.81	0.062	0.1653	0.0668
17	56.93	0.2426	2.1218	0.7968	132.85	309.58	245.68	0.118	0.0955	0.0571
18	126.75	270.91	0.3506	0.7631	154.99	485.91	329.94	0.0486	0.2323	0.018
19	39.61	204.86	0.6272	1.3535	278.64	246.08	252.68	0.0849	0.0657	0.0646
20	544.71	22.15	1.6656	1.5166	189.09	150.51	370.92	0.0691	0.0173	0.0659
21	625.41	298.91	1.9709	1.1933	122.88	298.54	471.92	0.2173	0.1904	0.0841
22	645.89	94.33	1.1737	0.3875	338.08	271.69	399.66	0.1823	0.2007	0.1389
23	212.04	242.76	1.9052	1.0623	130.64	64.88	218.86	0.0873	0.2352	0.1207
24	716.72	138.43	2.659	1.1927	480.74	276.1	45.15	0.1519	0.2008	0.1483
25	486.38	176.14	2.2052	0.9208	248.78	74.53	466.78	0.0754	0.2479	0.0894
26	11.52	143.27	1.2342	1.2919	213.88	129.27	189.18	0.0691	0.1403	0.1492
27	275.22	165.73	1.7966	1.4559	283.89	264.35	379.46	0.068	0.2513	0.0812
28	426.93	52.595	0.7847	1.0015	306.51	406.68	401.89	0.1773	0.2561	0.0462
29	502.21	176.17	2.285	0.0297	403.49	359.26	402.11	0.1361	0.1367	0.1099
30	357.38	46.44	1.0786	0.2896	440.39	250.4	297.89	0.0201	0.2213	0.0982
31	648.88	133.63	0.247	0.042	79.98	446.99	162.49	0.0146	0.2351	0.0944
32	115.86	161.28	0.9495	1.4687	44.58	475.35	374.17	0.0677	0.2438	0.0686
33	779.86	138.57	0.9437	0.761	156.56	231.17	238.23	0.1152	0.1198	0.0734
34	667.00	250.23	0.073	1.2484	303.67	143.79	257.79	0.0744	0.1987	0.0239
35	272.08	298.85	0.5533	0.7176	189.36	463.18	240.04	0.1777	0.2458	0.051
36	492.99	291.67	2.7047	0.7763	462.77	248.16	212	0.2486	0.2122	0.0013
37	243.00	223.77	2.0607	0.6687	28.5	438.66	88	0.2026	0.2436	0.1628
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
100	699.01	4.7027	2.6965	0.3599	311.91	109.87	259.13	0.0547	0.1259	0.1069

2) Assume a set of errors for DH parameters, leg joint offset, and spherical joint coordinate parameters (Table 3).

3) Based on the assumed errors, nominal parameter values, generated end-effector poses, and the carriage joint displacements, calculate the simulated actual leg lengths $l_{i,j}^{m}$ according to Equation (29) in Section 3.1.

4) Take the 54 kinematic parameter errors as variables in Equation (30) to fit the predicted leg lengths $l_{i,j}^{p}$ to data. The simulation will terminate if either the maximum number of iterations (e.g. G_{max}=40,000) is reached or the objective function value is below the user-defined threshold (e.g. 10^{-23} in this simulation).

5) Reorganize the 100 randomly generated data into four different data sets with different number of measurement poses (e.g. 15-, 25-, 50- and 100-poses). Figure 10 shows the simulation results for these four different data sets.

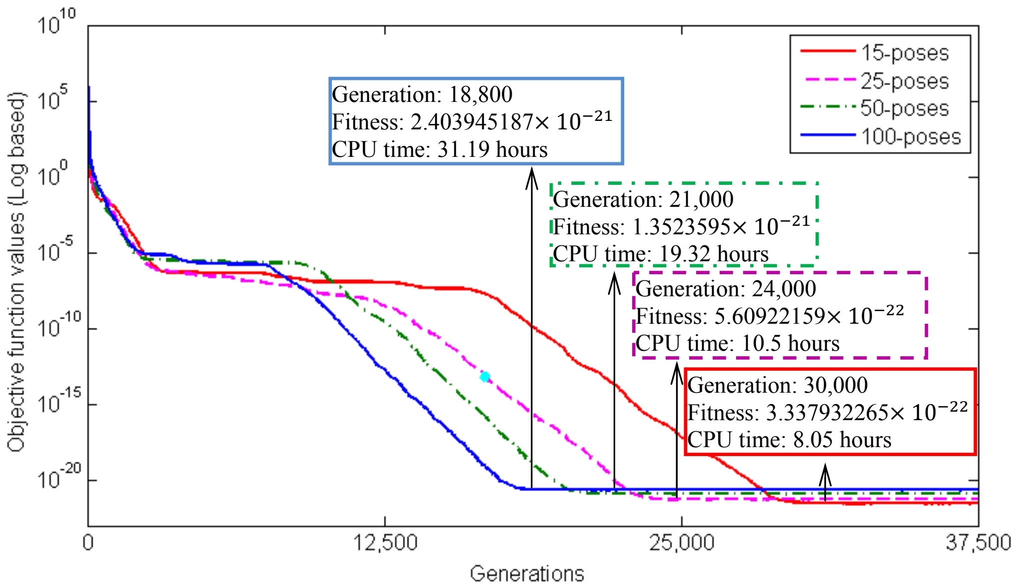


Figure 10. Simulation results of four different data sets.

From the results it can be seen that about 15 measurement poses are adequate for the objective function value converging to a very small value (10^{-22}). With the increase of measurement poses, the simulation time increases accordingly but the number of simulation generations decreases when the simulation has already converged. For instance, the convergence time is about 8.05 hours in the case of 15 poses, but 31.19 hours in the case of 100 poses. The increase of measurement poses cannot improve the convergence of the objective function values, but it can improve the robustness of the identified parameter errors from the whole workspace point of view.

6) To simulate the influence of the same number of measurement poses on different part of workspace, construct five data sets with same number of measurement poses (e.g.

data set 1: pose number 1 to number 15; data set 2: pose number 16 to number 30, etc., in Figure 11). The simulation results are demonstrated in Figure 11. It shows that all of the selected runs can converge to almost the same small fitness value (10^{-22}), but the simulation time and required generations is different. It implies that the runtime can be reduced by choosing an optimal number of measurement configurations. However, the determination of the optimal number of configurations for data acquisition, in order to perform a successful calibration, is still a research issue that remains to be addressed. Different criteria and opinions can be found in different literature presentations [82][83][84][85]. An overview of this problem can be referenced in the work of [41]

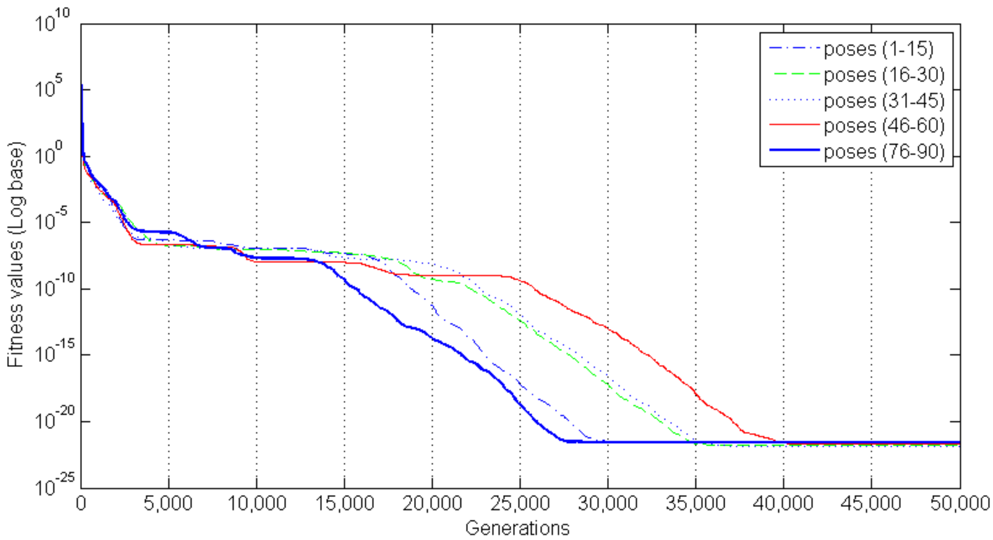


Figure 11. Simulation results of 15 measurement poses in five different data sets.

4.2 Denavit-Hartenberg Hybrid Model Using MCMC-Based Identification Method

By using the DE-based identification method, all parameter errors can be identified even when correlations exist. However, the redundant parameters which result in correlations have to be eliminated to obtain a more accurate error model [86]. To solve this problem, we propose the MCMC-based method for parameter correlation analysis as well as for statistical error parameter estimation. The task of simulation is to obtain a posterior distribution chain for parameter errors using the MCMC sampling methods. The MCMC toolbox for Matlab developed by Laine [87] is employed and the initial guess of the MCMC simulation are selected from the results of DE algorithm. The obtained MCMC simulation chain is a matrix of samples, which is commonly used to calculate and analyze the posterior means, standard deviations and correlations of parameter errors. In what follows, we first simulate the DH-based identification algorithms of Section 3.1 to check whether redundant parameters exist, and then we simulate the refined error model with reduced parameters under two experimental conditions (with measurement noise and without measurement noise). Set the length of chain, N_{simu}, to be 200,000 for the MCMC simulation, then a posterior distribution matrix of $200{,}000 \times 54$ can be achieved after the simulation. The mean values and standard deviations of each error parameter can be computed from this chain.

4.2.1 Results of 54 parameter errors without measurement noise

The identification results as well as the nominal values, assumed parameter errors, are listed in Table 3. From the table, it can be seen that the posterior mean values of independent parameters have been identified to be exactly the same as the assumed errors with high standard deviations (10^{-5} mm and 10^{-8} rad.), whereas the correlated or dependent parameters have not been correctly identified, so they have larger standard deviations (>10^{-3} mm).

Table 3. Nominal values, assumed errors, identified posterior mean values, and standard deviations for the 54-parameter model (without measurement noise)

No.	Symbols (nominal, errors)	Nominal Values (mm)	Assumed Errors (mm, ^{o})	Posterior mean (mm, ^{o})	Posterior Std. (rad.)
1	$\alpha_1,\delta\alpha_1$	π/2	0.0782°	0.07819°	2.352×10^{-8}
2	$\alpha_2,\delta\alpha_2$	π/2	0.0571°	0.0571°	3.1528×10^{-9}
3	$\alpha_3,\delta\alpha_3$	π/2	-0.048°	-0.048°	5.7552×10^{-9}
4	$\alpha_4,\delta\alpha_4$	-π/2	0.0417°	0.04173°	1.6735×10^{-5}
5	$a_3,\delta a_3$	252	-0.2164	-0.2164	2.2333×10^{-5}
6	$a_4,\delta a_4$	354	-0.4451	-0.4451	0.0048288
7	$d_3,\delta d_3$	422	0.1681	0.1681	2.7881×10^{-5}
8	$d_4,\delta d_4$	0	-0.3857	-0.38564	0.0073678
9	$\theta_1,\delta\theta_1$	0	0.0213°	0.0213°	2.6166×10^{-8}
10	$\theta_2,\delta\theta_2$	π/2	0.0794°	0.0794°	2.2173×10^{-8}
11	$\theta_3,\delta\theta_3$	0	0.0464°	0.0464°	4.8552×10^{-8}
12	$\theta_4,\delta\theta_4$	0	0.0345°	0.03449°	1.1837×10^{-5}
13	$a_{1x},\delta a_{1x}$	231.6663	-0.0654	-0.06538	0.0048255
14	$a_{1y},\delta a_{1y}$	-231.9022	0.0687	0.068645	0.0073711
15	$a_{1z},\delta a_{1z}$	0	0.0928	0.092879	0.0041492
16	$a_{2x},\delta a_{2x}$	316.663	0.0448	0.044815	0.0048282
17	$a_{2y},\delta a_{2y}$	-84.6778	-0.0942	-0.09425	0.0073716
18	$a_{2z},\delta a_{2z}$	0	-0.0731	-0.07301	0.0062136
19	$a_{3x},\delta a_{3x}$	85	0.0229	0.02291	0.004833
20	$a_{3y},\delta a_{3y}$	316.58	0.0133	0.013246	0.0073733
21	$a_{3z},\delta a_{3z}$	0	-0.0136	-0.01367	0.0080805
22	$a_{4x},\delta a_{4x}$	-85	-0.0752	-0.07518	0.0048307
23	$a_{4y},\delta a_{4y}$	316.58	-0.0976	-0.09765	0.007371
24	$a_{4z},\delta a_{4z}$	0	0.0167	0.016552	0.0080475

No.	Symbols (nominal, errors)	Nominal Values (mm)	Assumed Errors (mm, ^{o})	Posterior mean (mm, ^{o})	Posterior Std. (rad.)
25	a_{5x},δa_{5x}	-316.663	0.0576	0.057614	0.0048249
26	a_{5y},δa_{5y}	-84.6778	-0.0486	-0.04865	0.0073654
27	a_{5z},δa_{5z}	0	0.0329	0.03272	0.0061585
28	a_{6x},δa_{6x}	-231.6663	-0.0117	-0.01168	0.0048258
29	a_{6y},δa_{6y}	-231.9022	0.0676	0.067545	0.0073668
30	a_{6z},δa_{6z}	0	0.0273	0.027181	0.004128
31	b_{1x},δb_{1x}	32.5	0.0581	0.058101	2.8297×10^{-5}
32	b_{1y},δb_{1y}	-125.93	-0.0648	-0.0648	1.5322×10^{-5}
33	b_{1z},δb_{1z}	0	0.0717	0.0717	1.6098×10^{-5}
34	b_{2x},δb_{2x}	125.309	0.0847	0.084701	2.8356×10^{-5}
35	b_{2y},δb_{2y}	34.819	-0.0478	-0.04779	1.4938×10^{-5}
36	b_{2z},δb_{2z}	0	0.0324	0.0324	1.8154×10^{-5}
37	b_{3x},δb_{3x}	92.809	-0.0139	-0.01389	2.7064×10^{-5}
38	b_{3y},δb_{3y}	91.111	-0.0266	-0.02660	1.8706×10^{-5}
39	b_{3z},δb_{3z}	0	-0.0281	-0.02810	2.1123×10^{-5}
40	b_{4x},δb_{4x}	-92.809	-0.0594	-0.0594	2.897×10^{-5}
41	b_{4y},δb_{4y}	91.111	0.0375	0.0375	1.9324×10^{-5}
42	b_{4z},δb_{4z}	0	0.0088	0.008799	1.9959×10^{-5}
43	b_{5x},δb_{5x}	-125.309	0.0228	0.022802	3.211×10^{-5}
44	b_{5y},δb_{5y}	34.819	-0.0566	-0.0566	1.4835×10^{-5}
45	b_{5z},δb_{5z}	0	-0.0368	-0.0368	1.4931×10^{-5}
46	b_{6x},δb_{6x}	-32.5	-0.0638	-0.06379	3.043×10^{-5}
47	b_{6y},δb_{6y}	-125.93	-0.0087	-0.00869	1.305×10^{-5}
48	b_{6z},δb_{6z}	231.6663	-0.0736	-0.0736	1.1859×10^{-5}
49	l_1,δl_1	350+sensor val.	-0.3794	-0.3794	2.3128×10^{-5}
50	l_2,δl_2	350+sensor val.	-0.0895	-0.0895	2.8054×10^{-5}
51	l_3,δl_3	350+sensor val.	0.1650	0.165	4.7325×10^{-5}
52	l_4,δl_4	350+sensor val.	-0.3048	-0.3048	5.3156×10^{-5}
53	l_5,δl_5	350+sensor val.	0.3233	0.3233	2.8076×10^{-5}
54	l_6,δl_6	350+sensor val.	0.0774	0.0774	1.5764×10^{-5}

For further analysis of the dependent parameters, the two-dimensional (2D) posterior distributions and one-dimensional (1D) marginal density for some parameters are depicted in Figures 12-17.

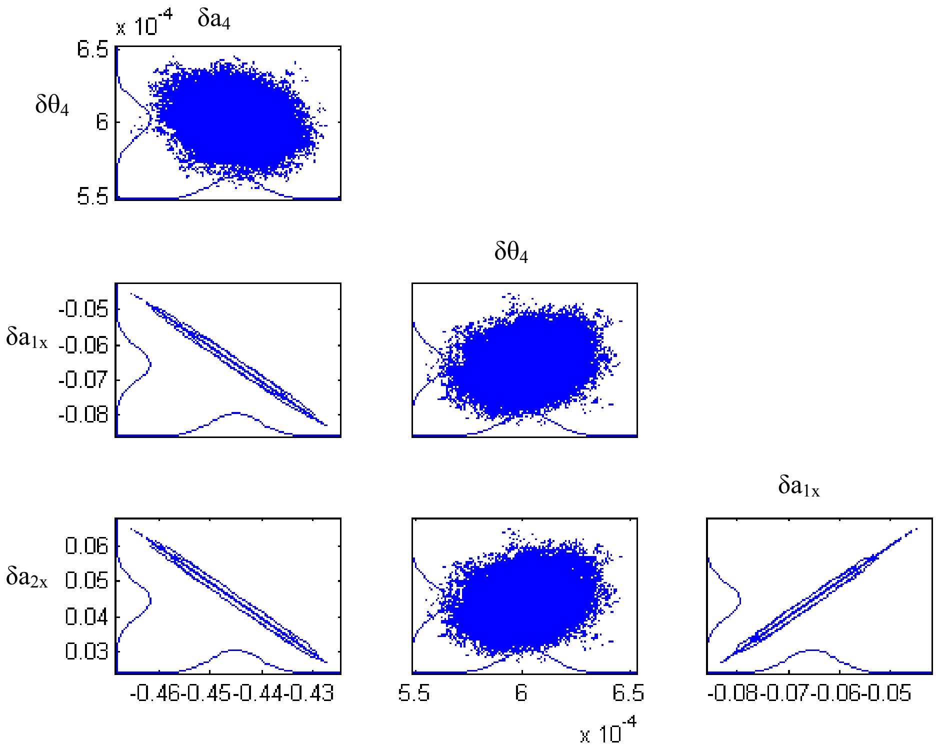


Figure 12. 2D marginal posterior distributions and 1D marginal density for parameters δa_4, $\delta\theta_4$, δa_{1x}, δa_{2x}.

From Figure 12, it can be seen that three parameter pairs (δa_4, δa_{1x}), (δa_4, δa_{2x}) and (δa_{1x}, δa_{2x}) are linearly correlated. Given more 2D posterior distributions (Figure 13), we can see that the parameter δa_4 is also linearly correlated with the rest of X direction parameters δa_{3x}, δa_{4x}, δa_{5x} and δa_{6x} as shown in Figure 13.

The same phenomenon can also be found in parameters between δd_4 and Y direction parameters δa_{1y}, δa_{2y}, δa_{3y}, δa_{4y}, δa_{5y} and δa_{6y} as shown in Figures 14-15.

The correlation phenomenon on Z direction is not so obvious. The parameter errors in Z direction are either correlated with $\delta\theta_4$ or $\delta\alpha_4$ (Figures 16-17). For instance, parameter $\delta\theta_4$ is linearly correlated with δa_{3z}, whereas δa_{1z} with δa_{2z} and $\delta\alpha_4$.

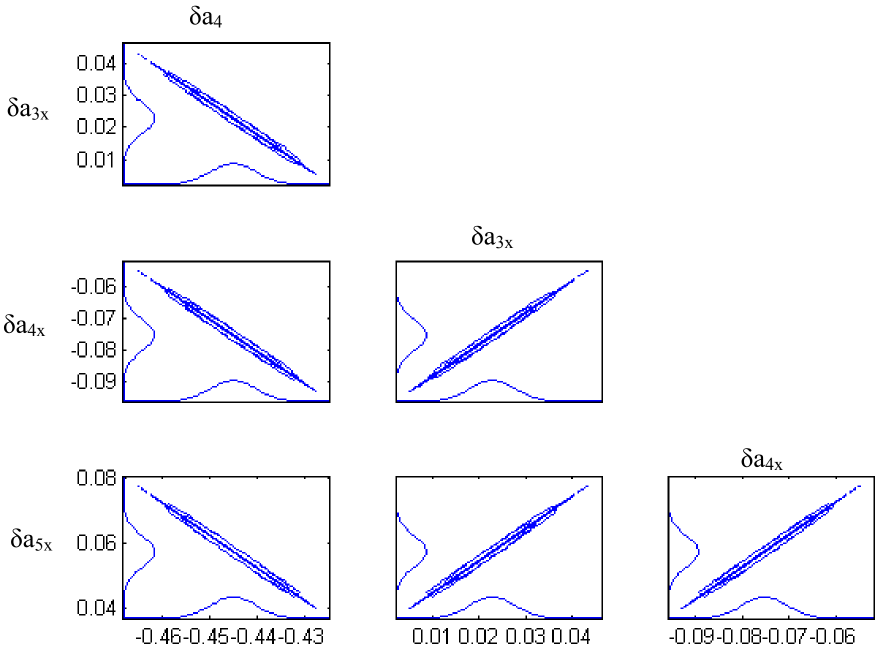


Figure 13. 2D marginal posterior distributions and 1D marginal density for parameters δa_4, δa_{3x}, δa_{4x}, δa_{5x}.

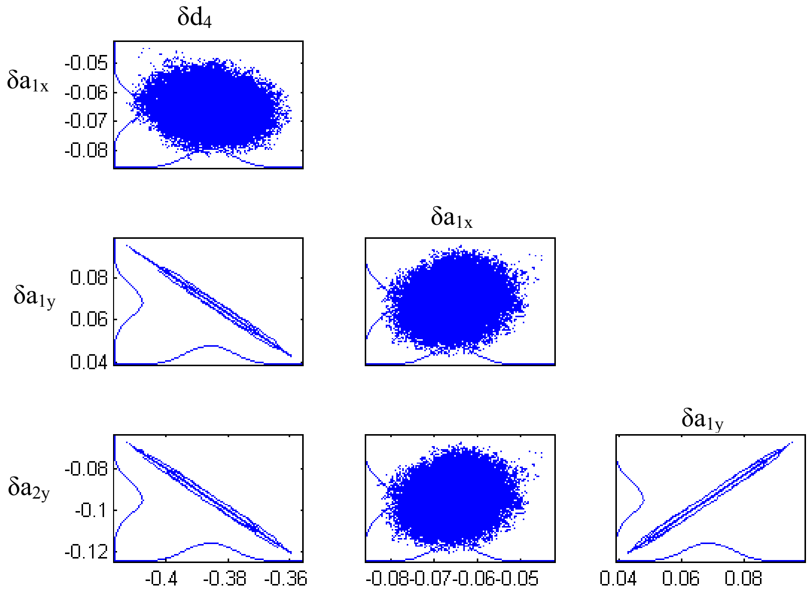


Figure 14. 2D marginal posterior distributions and 1D marginal density for parameters δd_d, δa_{1y}, δa_{2y}, δa_{1x}.

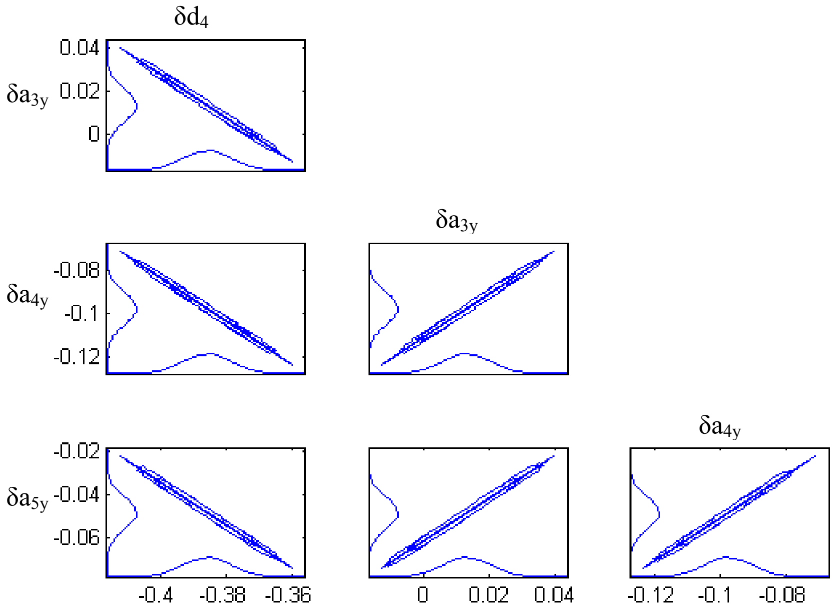


Figure 15. 2D marginal posterior distributions and 1D marginal density for parameters δd_4, δa_{3y}, δa_{4y}, δa_{5y}.

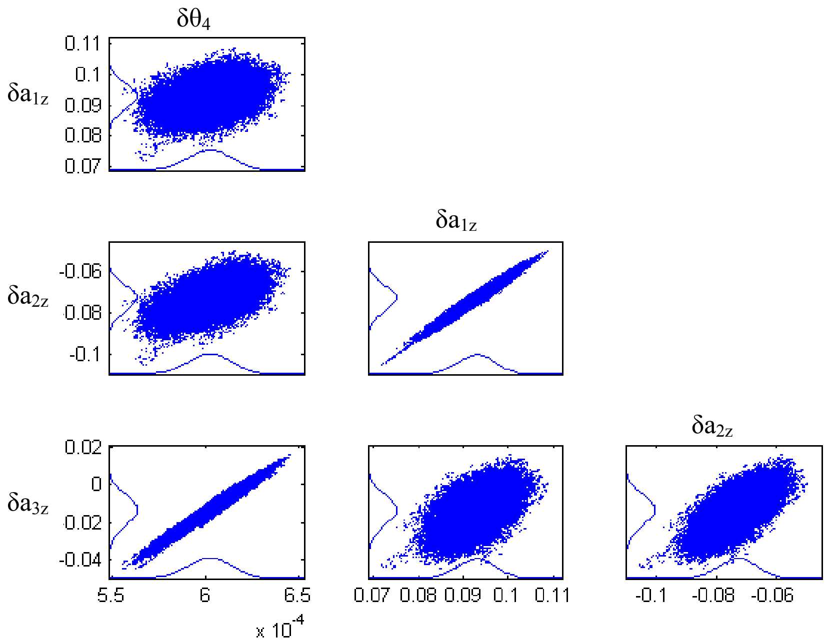


Figure 16. 2D marginal posterior distributions and 1D marginal density for parameters $\delta\theta_4$, δa_{1z}, δa_{2z}, δa_{3z}.

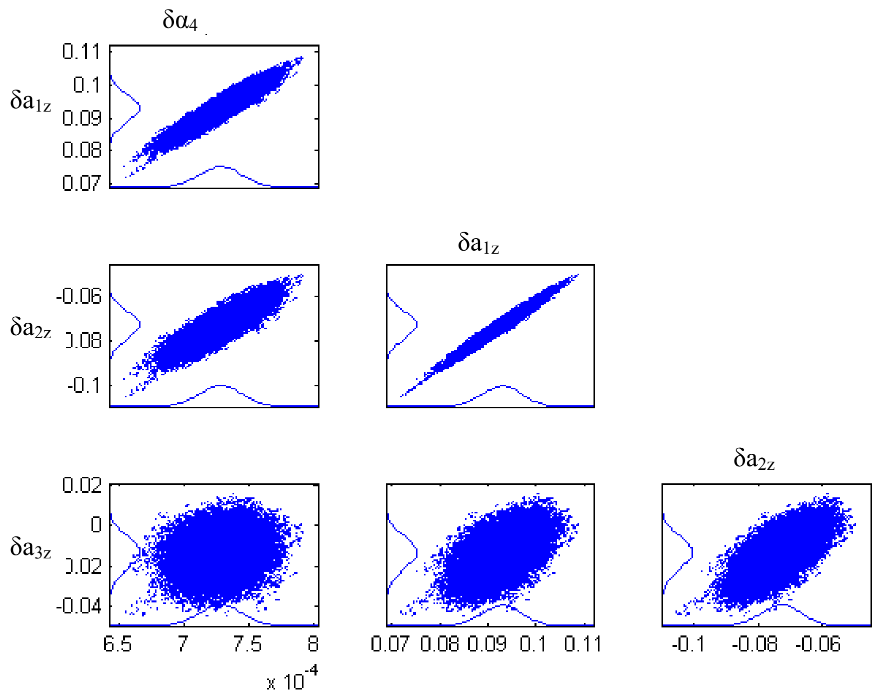


Figure 17. 2D marginal posterior distributions and 1D marginal density for parameters $\delta\alpha_4$, δa_{1z}, δa_{2z}, δa_{3z}.

Model refinement and re-parameterization can be made based on correlation analysis, but how to refine the model is still an unsolved problem and largely dependent on the specific model since the MCMC can only provide the graphical information of the parameter correlations. Basically, the model refinement could be achieved either by redeveloping a new model or just removing some of the correlated parameters in the model depending on the specific situations. In this work, we attempt to remove all of the parameter errors in spherical joint A_i to keep the remaining parameter errors identifiable and independent. Consequently, there are 36 independent and identifiable parameters left in the reduced model. It should be noted that this reduced model can only guarantee that the remaining parameters are identifiable and independent but cannot guarantee that the reduced model is the best model to satisfy the requirement of completeness and minimality simultaneously. Further model refinement work would be stressed on our future work.

4.2.2 Results of 36 parameter errors without measurement noise

Table 4 gives the simulation results of the posterior mean values and standard deviations of the 36 parameters. It can be seen that the parameter correlations have been successfully eliminated, and every parameter errors have been identified to be almost as the same as the assumed errors, and the standard deviations arrive at very high precision levels (10^{-6} mm and 10^{-9} *rad.*).

Table 4. Nominal values, assumed errors, identified posterior mean values, and standard deviations for the refined 36-parameter model (without measurement noise)

No.	Symbols (nominal, errors)	Nominal Values (mm)	Assumed Errors (mm, ^{o})	Posterior mean (mm, ^{o})	Posterior Std. (rad.)
1	$\alpha_1,\delta\alpha_1$	-90°	0.0782°	0.0782°	2.2777×10^{-9}
2	$\alpha_2,\delta\alpha_2$	90°	0.0571°	0.0571°	2.7999×10^{-9}
3	$\alpha_3,\delta\alpha_3$	90°	-0.048°	-0.048°	2.6991×10^{-9}
4	$\alpha_4,\delta\alpha_4$	90°	0.0417°	0.0417°	2.3243×10^{-9}
5	$a_3,\delta a_3$	252	-0.2164	-0.2164	8.8467×10^{-7}
6	$a_4,\delta a_4$	354	-0.4451	-0.4451	1.039×10^{-6}
7	$d_3,\delta d_3$	422	0.1681	0.1681	2.613×10^{-6}
8	$d_4,\delta d_4$	0	-0.3857	-0.3857	1.4035×10^{-6}
9	$\theta_1,\delta\theta_1$	0	0.0213°	0.0213°	2.6084×10^{-9}
10	$\theta_2,\delta\theta_2$	90°	0.0794°	0.0794°	1.2234×10^{-9}
11	$\theta_3,\delta\theta_3$	0°	0.0464°	0.0464°	3.2017×10^{-9}
12	$\theta_4,\delta\theta_4$	0°	0.0345°	0.0345°	1.6873×10^{-9}
13	$b_{1x},\delta b_{1x}$	32.5	0.0581	0.0581	2.5743×10^{-6}
14	$b_{1y},\delta b_{1y}$	-125.93	-0.0648	-0.0648	1.362×10^{-6}
15	$b_{1z},\delta b_{1z}$	0	0.0717	0.0717	1.3075×10^{-6}
16	$b_{2x},\delta b_{2x}$	125.309	0.0847	0.0847	2.5714×10^{-6}
17	$b_{2y},\delta b_{2y}$	34.819	-0.0478	-0.0478	1.3262×10^{-6}
18	$b_{2z},\delta b_{2z}$	0	0.0324	0.0324	1.1825×10^{-6}
19	$b_{3x},\delta b_{3x}$	92.809	-0.0139	-0.0139	2.6619×10^{-6}
20	$b_{3y},\delta b_{3y}$	91.111	-0.0266	-0.0266	1.4087×10^{-6}
21	$b_{3z},\delta b_{3z}$	0	-0.0281	-0.0281	9.956×10^{-7}
22	$b_{4x},\delta b_{4x}$	-92.809	-0.0594	-0.0594	2.8469×10^{-6}
23	$b_{4y},\delta b_{4y}$	91.111	0.0375	0.0375	1.3211×10^{-6}
24	$b_{4z},\delta b_{4z}$	0	0.0088	0.0088	9.9319×10^{-7}
25	$b_{5x},\delta b_{5x}$	-125.309	0.0228	0.0228	2.982×10^{-6}
26	$b_{5y},\delta b_{5y}$	34.819	-0.0566	-0.0566	1.0043×10^{-6}
27	$b_{5z},\delta b_{5z}$	0	-0.0368	-0.0368	1.0054×10^{-6}
28	$b_{6x},\delta b_{6x}$	-32.5	-0.0638	-0.0638	2.7734×10^{-6}
29	$b_{6y},\delta b_{6y}$	-125.93	-0.0087	-0.0087	1.03×10^{-6}
30	$b_{6z},\delta b_{6z}$	0	-0.0736	-0.0736	9.6405×10^{-7}
31	$l_1,\delta l_1$	350+sensor val.	-0.3794	-0.3794	1.584×10^{-6}
32	$l_2,\delta l_2$	350+sensor val.	-0.0895	-0.0895	1.372×10^{-6}
33	$l_3,\delta l_3$	350+sensor val.	0.1650	0.1650	1.3435×10^{-6}
34	$l_4,\delta l_4$	350+sensor val.	-0.3048	-0.3048	1.6412×10^{-6}
35	$l_5,\delta l_5$	350+sensor val.	0.3233	0.3233	1.5983×10^{-6}
36	$l_6,\delta l_6$	350+sensor val.	0.0774	0.0774	1.3004×10^{-6}

4.2.3 Results of 36 parameter errors with measurement noise

To simulate real experimental conditions, we assume that the position and orientation of the end-effector will be measured by using a high precision laser tracker. The position measurement accuracy is ±0.01 mm and the orientation measurement accuracy is ±0.00001 rad. Measurement noise is regarded as a Gaussian distribution, whose ranges obey the 3σ rule. The standard deviations of measurement noise for position and orientation are 0.003 mm and 0.000003 rad., respectively. Table 5 presents the simulation results of posterior mean values and standard deviations of the refined 36 parameters with pose measurement noise.

Table 5. Nominal values, assumed errors, identified posterior mean values, and standard deviations for the refined 36-parameter model (with measurement noise)

No.	Symbols (nominal, errors)	Nominal Values (mm)	Assumed Errors (mm, °)	Posterior mean (mm, °)	Posterior Std. (rad.)
1	$\alpha_1,\delta\alpha_1$	-90°	0.0782°	0.077859°	2.3614×10^{-8}
2	$\alpha_2,\delta\alpha_2$	90°	0.0571°	0.057234°	2.699×10^{-8}
3	$\alpha_3,\delta\alpha_3$	90°	-0.048°	-0.04781°	2.697×10^{-8}
4	$\alpha_4,\delta\alpha_4$	90°	0.0417°	0.041479°	2.3256×10^{-8}
5	$a_3,\delta a_3$	252	-0.2164	-0.21465	9.0571×10^{-6}
6	$a_4,\delta a_4$	354	-0.4451	-0.4450	1.0434×10^{-5}
7	$d_3,\delta d_3$	422	0.1681	0.17487	2.6985×10^{-5}
8	$d_4,\delta d_4$	0	-0.3857	-0.3856	1.4193×10^{-5}
9	$\theta_1,\delta\theta_1$	0	0.0213°	0.021623°	2.5993×10^{-8}
10	$\theta_2,\delta\theta_2$	90°	0.0794°	0.079532°	1.2297×10^{-8}
11	$\theta_3,\delta\theta_3$	0°	0.0464°	0.046937°	3.2475×10^{-8}
12	$\theta_4,\delta\theta_4$	0°	0.0345°	0.034722°	1.6977×10^{-8}
13	$b_{1x},\delta b_{1x}$	32.5	0.0581	0.061509	2.6428×10^{-5}
14	$b_{1y},\delta b_{1y}$	-125.93	-0.0648	-0.065029	1.3881×10^{-5}
15	$b_{1z},\delta b_{1z}$	0	0.0717	0.069514	1.2575×10^{-5}
16	$b_{2x},\delta b_{2x}$	125.309	0.0847	0.088151	2.6055×10^{-5}
17	$b_{2y},\delta b_{2y}$	34.819	-0.0478	-0.047026	1.2997×10^{-5}
18	$b_{2z},\delta b_{2z}$	0	0.0324	0.030068	1.1668×10^{-5}
19	$b_{3x},\delta b_{3x}$	92.809	-0.0139	-0.010355	2.6603×10^{-5}
20	$b_{3y},\delta b_{3y}$	91.111	-0.0266	-0.025498	1.4205×10^{-5}
21	$b_{3z},\delta b_{3z}$	0	-0.0281	-0.030009	9.9522×10^{-6}
22	$b_{4x},\delta b_{4x}$	-92.809	-0.0594	-0.056781	2.8328×10^{-5}
23	$b_{4y},\delta b_{4y}$	91.111	0.0375	0.038118	1.3413×10^{-5}
24	$b_{4z},\delta b_{4z}$	0	0.0088	0.007394	1.0058×10^{-5}
25	$b_{5x},\delta b_{5x}$	-125.309	0.0228	0.024621	2.9881×10^{-5}
26	$b_{5y},\delta b_{5y}$	34.819	-0.0566	-0.055924	1.0115×10^{-5}
27	$b_{5z},\delta b_{5z}$	0	-0.0368	-0.037492	1.0006×10^{-5}

28	$b_{6x},\delta b_{6x}$	-32.5	-0.0638	-0.060853	2.804×10^{-5}
29	$b_{6y},\delta b_{6y}$	-125.93	-0.0087	-0.008302	1.0275×10^{-5}
30	$b_{6z},\delta b_{6z}$	0	-0.0736	-0.07438	9.5351×10^{-6}
31	$l_1,\delta l_1$	350+sensor val.	-0.3794	-0.38249	1.5735×10^{-5}
32	$l_2,\delta l_2$	350+sensor val.	-0.0895	-0.092193	1.3539×10^{-5}
33	$l_3,\delta l_3$	350+sensor val.	0.1650	0.16235	1.3171×10^{-5}
34	$l_4,\delta l_4$	350+sensor val.	-0.3048	-0.3082	1.6392×10^{-5}
35	$l_5,\delta l_5$	350+sensor val.	0.3233	0.31997	1.6464×10^{-5}
36	$l_6,\delta l_6$	350+sensor val.	0.0774	0.074825	1.3081×10^{-5}

The results show that all of the parameter errors have successfully converged to the assumed errors with only a slight difference, and the standard deviations arrive at very high precisions (10^{-5} mm and 10^{-8} rad.).

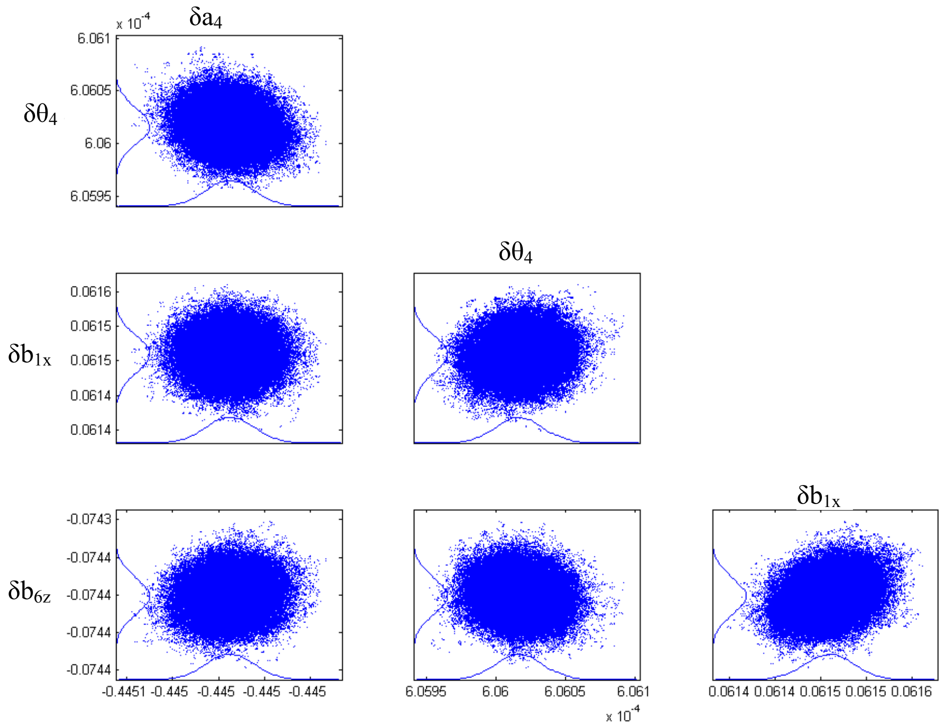


Figure 18. 2D marginal posterior distributions and 1D marginal density for parameters δa_4, $\delta\theta_4$, δb_{1x}, δb_{6z}.

From Table 5 and Figure 18, it can be seen that every parameter is independent of each other and identifiable. Measurement noises do have an influence on the identification results, but the MCMC-based identification method is able to lower the influences to the best extent.

4.3 Product-of-Exponential Model Using Differential-Evolution Identification Method

For our Product-of-Exponential (POE) error model, the Differential-Evolution (DE) identification method can be adopted to identify the parameter errors. The detailed kinematic parameter values are listed in Table 6.

Table 6. Kinematic parameters in the reference configuration

Symbols	Values (mm)		Symbols	Values (mm)
l_0	45		P_{3x}	0
l_1	330		P_{3y}	0
l_2	252		P_{3z}	-628
l_3	314		P_{4x}	0
l_4	116.84		P_{4y}	-313
			P_{4z}	-376

The simulation procedures are as follows:

1) Randomly generate 100 end-effector measurement poses g_{st}^{m} within the robot's workspace, and accordingly generate 100 joint displacements for the carriage actuators. The randomly generated end-effector poses, carriage joint displacements and the DE control parameters are the same as in Table 2, Section 4.1.
2) Assume a set of parameter errors for the carriage twist (Table 7). The assumed errors should meet the requirement of $\|\boldsymbol{\omega}_i+\delta\boldsymbol{\omega}_i\|=1$, $(\boldsymbol{\omega}_i+\delta\boldsymbol{\omega}_i)^T(\mathbf{v}_i+\delta\mathbf{v}_i)=0$ for revolute joint, and $\|\mathbf{v}_i+\delta\mathbf{v}_i\|=1$ for prismatic joint. The leg joint offset errors and the coordinate errors of spherical joints A_i and B_i are randomly generated at their tolerance limits (Table 8).
3) Based on the above assumed errors, generated poses, nominal kinematic values, and carriage joint displacements, we can calculate the actual leg lengths $d_{i,j}^{m}$ based on Equations (33) and (34) in Section 3.2. In reality, the leg lengths can be obtained from the built-in sensor readings.
4) Take the 60 parameter errors as decision variables in the objective function so as to calculate the predicted leg lengths $d_{i,j}^{p}$. Then the task of simulation is to employ the DE algorithm to search for an optimal combination of parameter errors to minimize the value of the objective function under some program terminal conditions.

As in the previous section, the simulation is also conducted under two different conditions:

a) An ideal experimental condition where measurement noises are not considered. This is to verify the effectiveness of the proposed calibration methods.

b) A real experimental condition where measurement noises are added into the randomly generated measurement poses. This is to verify the robustness of the proposed calibration methods.

The identification results of these two different conditions are listed in Table 7 and Table 8.

The results show that all of the error parameters have been successfully identified. Under ideal experimental conditions, we can expect almost the same identified parameter errors as the assumed ones. Under imperfect experimental conditions, the identified error parameters are also very close to the assumed ones.

Table 7. Nominal and identified parameters of carriage

No.	Symbols of twist element	Nominal twist values ξ_{si} (mm)	Assumed twist errors $\delta\xi_{si}$ (mm)	Identified twist errors without noise$\delta\xi_{si}$ (mm)	Identified twist errors with noise $\delta\xi_{si}$ (mm)
1	$\omega_{1x}, \delta\omega_{1x}$	0	-	-	-
2	$\omega_{1y}, \delta\omega_{1y}$	0	-	-	-
3	$\omega_{1z}, \delta\omega_{1z}$	0	-	-	-
4	$\upsilon_{1x}, \delta\upsilon_{1x}$	1	cos(0.02)-1	-1.99993×10^{-4}	-1.99418×10^{-4}
5	$\upsilon_{1y}, \delta\upsilon_{1y}$	0	sin(0.02)	0.0199987	0.0199996
6	$\nu_{1z}, \delta\upsilon_{1z}$	0	0	1.32017×10^{-17}	-9.3368×10^{-7}
7	$\omega_{2x}, \delta\omega_{2x}$	0	-	-	-
8	$\omega_{2y}, \delta\omega_{2y}$	0	-	-	-
9	$\omega_{2z}, \delta\omega_{2z}$	0	-	-	-
10	$\upsilon_{2x}, \delta\upsilon_{2x}$	0	0	-1.79319×10^{-15}	-7.5492×10^{-7}
11	$\upsilon_{2y}, \delta\upsilon_{2y}$	0	sin(0.02)	0.0199987	0.0199943
12	$\nu_{2z}, \delta\upsilon_{2z}$	1	cos(0.02)-1	-1.99993×10^{-4}	-1.99369×10^{-4}
13	$\omega_{3x}, \delta\omega_{3x}$	0	0	2.39192×10^{-13}	6.7108×10^{-5}
14	$\omega_{3y}, \delta\omega_{3y}$	1	cos(0.02)-1	-1.99993×10^{-4}	3.63863×10^{-4}
15	$\omega_{3z}, \delta\omega_{3z}$	0	sin(0.02)	0.0199987	0.0205498
16	$\upsilon_{3x}, \delta\upsilon_{3x}$	628	0.2	0.2	0.199999
17	$\upsilon_{3y}, \delta\upsilon_{3y}$	0	0	-9.2727×10^{-17}	1.08622×10^{-7}
18	$\nu_{3z}, \delta\upsilon_{3z}$	0	0	9.85174×10^{-17}	-4.4995×10^{-7}
19	$\omega_{4x}, \delta\omega_{4x}$	1	cos(0.02)-1	-1.99993×10^{-4}	4.29679×10^{-4}
20	$\omega_{4y}, \delta\omega_{4y}$	0	sin(0.02)	0.0199987	0.0192154
21	$\omega_{4z}, \delta\omega_{4z}$	0	0	-4.3303×10^{-15}	-0.001064
22	$\upsilon_{4x}, \delta\upsilon_{4x}$	0	0	-2.8376×10^{-16}	-5.53304×10^{-7}
23	$\upsilon_{4y}, \delta\upsilon_{4y}$	-376	0	2.79982×10^{-15}	-1.09392×10^{-6}
24	$\nu_{4z}, \delta\upsilon_{4z}$	313	0.2	0.2	0.200005

Table 8. Nominal and identified parameters of the Hexa-WH (unit: mm)

No.	Symbols (Nominal, errors)	nominal values	Assumed errors	Identified errors without noise	Identified errors with noise
1	$a_{51x}, \delta a_{51x}$	231.6	-0.0654	-0.0654	-0.0596501
2	$a_{51y}, \delta a_{51y}$	-231.9	0.0687	0.0687	0.0693789

No.	Symbols (Nominal, errors)	nominal values	Assumed errors	Identified errors without noise	Identified errors with noise
3	$a_{51z}, \delta a_{51z}$	0	0.0928	0.0928	0.0930406
4	$a_{52x}, \delta a_{52x}$	316.6	0.0448	0.0448	0.0479849
5	$a_{52y}, \delta a_{52y}$	-84.67	-0.0942	-0.0942	-0.0961375
6	$a_{52z}, \delta a_{52z}$	0	-0.0731	-0.0731	-0.0724037
7	$a_{53x}, \delta a_{53x}$	85	0.0229	0.0229	0.0192319
8	$a_{53y}, \delta a_{53y}$	316.58	0.0133	0.0133	0.0128637
9	$a_{53z}, \delta a_{53z}$	0	-0.0136	-0.0136	-0.0264649
10	$a_{54x}, \delta a_{54x}$	-85	-0.0752	-0.0752	-0.077656
11	$a_{54y}, \delta a_{54y}$	316.58	-0.0976	-0.0976	-0.100211
12	$a_{54z}, \delta a_{54z}$	0	0.0167	0.0167	0.0057957
13	$a_{55x}, \delta a_{55x}$	-316.6	0.0576	0.0576	0.0579317
14	$a_{55y}, \delta a_{55y}$	-84.67	-0.0486	-0.0486	-0.0487524
15	$a_{55z}, \delta a_{55z}$	0	0.0329	0.0329	0.0309625
16	$a_{56x}, \delta a_{56x}$	-231.6	-0.0117	-0.0117	-0.0118311
17	$a_{56y}, \delta a_{56y}$	-231.9	0.0676	0.0676	0.0727291
18	$a_{56z}, \delta a_{56z}$	0	0.0273	0.0273	0.02563
19	$b_{t1x}, \delta b_{t1x}$	32.5	0.0581	0.0581	0.0636969
20	$b_{t1y}, \delta b_{t1y}$	-125.9	-0.0648	-0.0648	-0.065743
21	$b_{t1z}, \delta b_{t1z}$	0	0.0717	0.0717	0.0688051
22	$b_{t2x}, \delta b_{t2x}$	125.3	0.0847	0.0847	0.0868672
23	$b_{t2y}, \delta b_{t2y}$	34.8	-0.0478	-0.0478	-0.0497133
24	$b_{t2z}, \delta b_{t2z}$	0	0.0324	0.0324	0.0325399
25	$b_{t3x}, \delta b_{t3x}$	92.8	-0.0139	-0.0139	-0.0171407
26	$b_{t3y}, \delta b_{t3y}$	91.1	-0.0266	-0.0266	-0.0277242
27	$b_{t3z}, \delta b_{t3z}$	0	-0.0281	-0.0281	-0.0392927
28	$b_{t4x}, \delta b_{t4x}$	-92.8	-0.0594	-0.0594	-0.0623811
29	$b_{t4y}, \delta b_{t4y}$	91.1	0.0375	0.0375	0.0326712
30	$b_{t4z}, \delta b_{t4z}$	0	0.0088	0.0088	0.000425282
31	$b_{t5x}, \delta b_{t5x}$	-125.3	0.0228	0.0228	0.0222086
32	$b_{t5y}, \delta b_{t5y}$	34.8	-0.0566	-0.0566	-0.0616722
33	$b_{t5z}, \delta b_{t5z}$	0	-0.0368	-0.0368	-0.0416378
34	$b_{t6x}, \delta b_{t6x}$	-32.5	-0.0638	-0.0638	-0.620393
35	$b_{t6y}, \delta b_{t6y}$	-125.9	-0.0087	-0.0087	-0.00470459
36	$b_{t6z}, \delta b_{t6z}$	231.6	-0.0736	-0.0736	-0.0752284
37	$d_1, \delta d_1$	350+sensor	-0.3794	-0.3794	-0.382394

No.	Symbols (Nominal, errors)	nominal values	Assumed errors	Identified errors without noise	Identified errors with noise
38	$d_2, \delta d_2$	350+sensor	-0.0895	-0.0895	-0.0897172
39	$d_3, \delta d_3$	350+sensor	0.165	0.165	0.16719
40	$d_4, \delta d_4$	350+sensor	-0.3048	-0.3048	-0.302824
41	$d_5, \delta d_5$	350+sensor	0.3233	0.3233	0.319244
42	$d_6, \delta d_6$	350+sensor	0.0774	0.0774	0.0781733

The set of 100 randomly generated data is reorganized into four subsets with different numbers of measurement poses; the simulation results of the objective function fitness values under imperfect experimental condition are plotted in Figure 19. It can be seen that with the increase of measurement poses, the final fitness value and the CPU time are also increased. At the generation 6000, the CPU time for the data set of 100 poses is 29 hours and the fitness value is about 0.00117mm, while the CUP time for the data set of 25 poses is only 7.3 hours and the fitness value is about 0.000264mm. It can also be seen that the more measurement poses used, the fewer generations needed to converge. To improve the robustness of identified parameter errors and reduce the effect of measurement noise, it is recommended to use as many measurement poses as possible to identify parameter errors, and the selected measurement pose configurations should cover the entire workspace, especially the workspace under extreme situations, such as the boundary of joint motion.

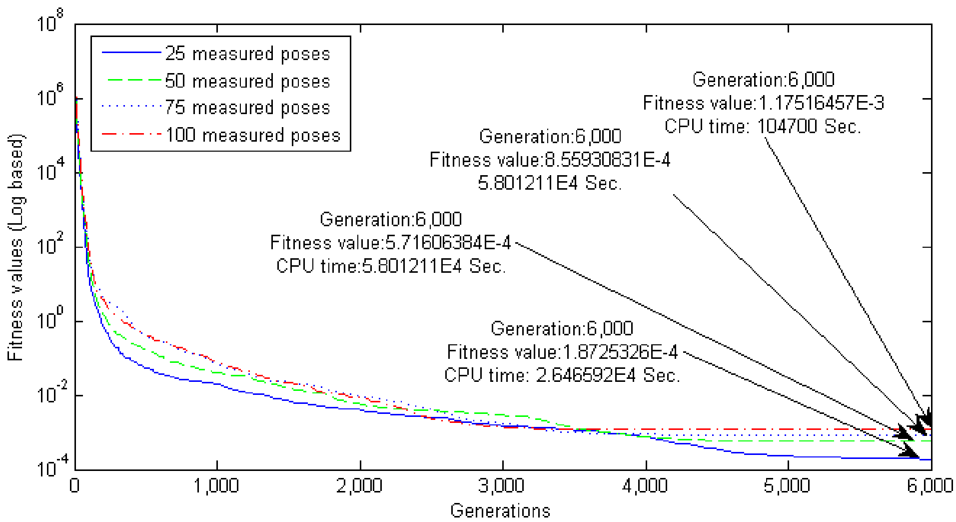


Figure 19. Fitness values of four different runs with four different measurement data sets.

To validate the identified parameter errors under imperfect experiment conditions, we can randomly generate another set of 25 joint-displacement vectors for carriage and the associated 25 leg-length vectors for the Hexa-WH, and then numerically solve the forward kinematics to find the end-effector pose values by the DE algorithm. The end-effector pose values after calibration can be obtained by including the identified parameter errors into the error model, and the values before calibration can be obtained without considering the identified parameter errors in the error model. Table 9 gives the root mean square (RMS) for position and orientation values, as well as the maximum position and orientation values of the 25 end-effector poses before and after calibration.

Table 9. Results of 25 end-effector poses before and after calibration

Errors type	Before calibration	After calibration
RMS position	0.3604 mm	0.001mm
RMS orientation	0.0316°	0.000248°
Max. position	3.797 mm	0.0098 mm
Max. orientation	0.4778°	0.0024°

Furthermore, position errors before and after calibration for the 25 end-effector pose configurations can be plotted (Figure 20 and Figure 21).

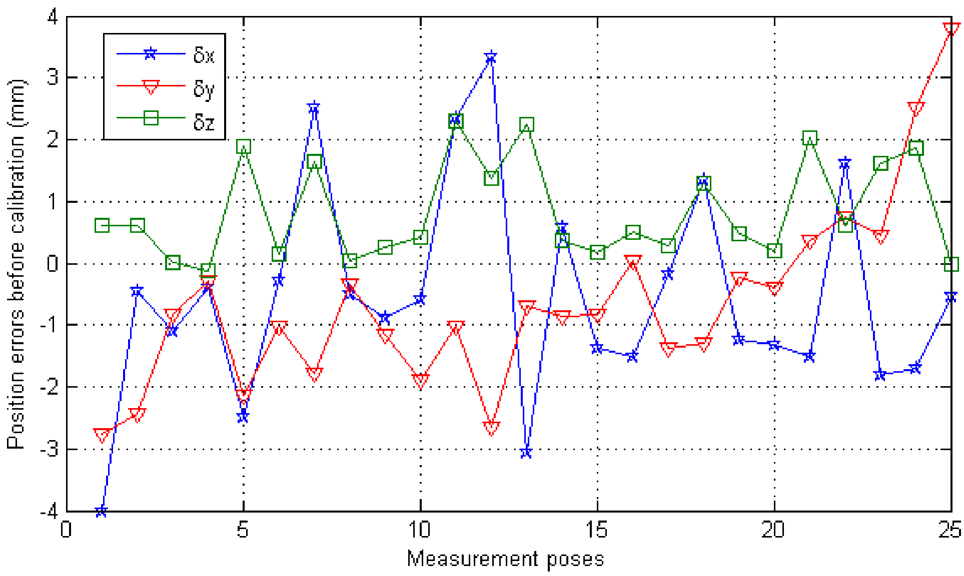


Figure 20. Position errors before calibration in the 25 end-effector pose configurations.

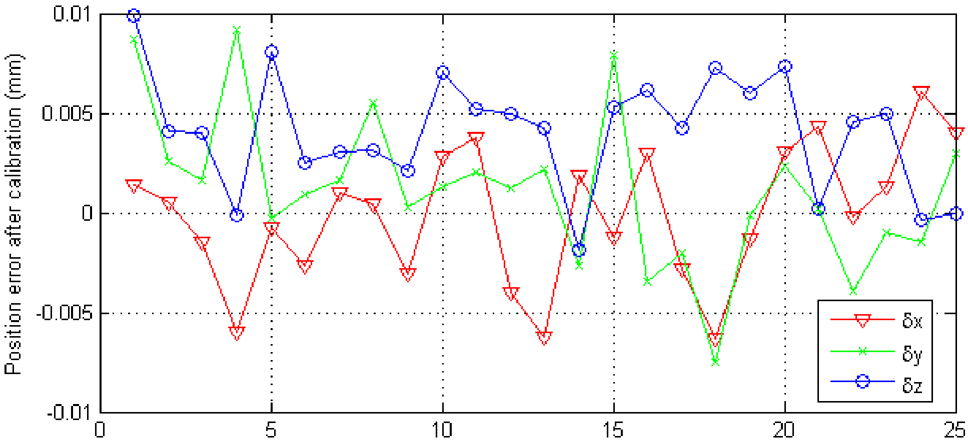


Figure 21. Position errors after calibration in the 25 end-effector pose configurations.

From the simulation results in Table 9 and Figures 20 and 21, it can be seen that the accuracy of the end-effector has improved and reaches the same precision level as the given external measurement device. The end-effector pose error before calibration is dependent on the assumed error parameter values, and the accuracy of the end-effector after calibration is dependent on the accuracy of the given measurement device system.

VALIDATION RESULTS BY USING SOLIDWORKS

This chapter introduces a validation method for the product-of-exponential (POE) calibration method by using the 3-2-1 wire-based pose measurement system [88][89][90] in the Solidworks environment, as demonstrated in Figure 22. The idea of this simulation method lies in the adjustment of the hexapod leg lengths, the carriage revolute angles and the slide displacements to form different pose configurations in the Solidworks environment. The 3-2-1 pose estimation method can be used to calculate the end-effector poses since a set of wire lengths can be measured for each pose configuration in the Solidworks environment. Unlike the numerical simulations in Chapter 4 where a set of randomly generated parameter errors and end-effector poses exist, the simulations in this chapter are much closer to the real working environment since the error parameter values are unknown and the end-effector poses are calculated by the 3-2-1 wire-based pose estimation system. Section 5.1 introduces the three-sphere-intersection algorithms which are the basis of the 3-2-1 wire-based pose measurement method. Section 5.2 presents a 3-2-1 wire-based pose estimation method for the hybrid IWR robot. Section 5.3 gives the simulation results and comments.

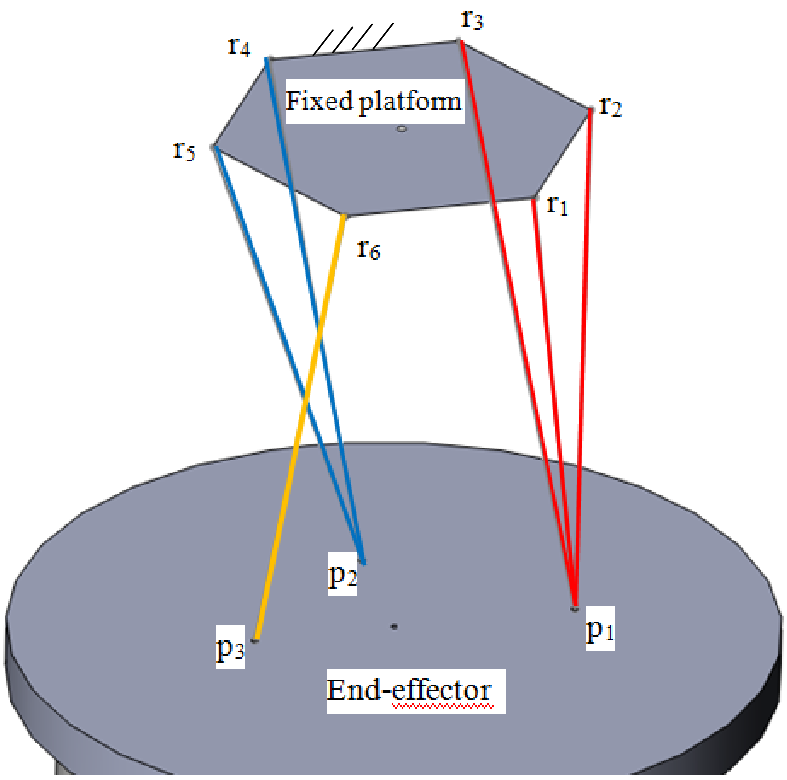


Figure 22. A scheme of 3-2-1 wire-based 3D pose estimation system.

5.1 Three Spheres Intersection Algorithm

For the intersection point of three given spheres (Figure 23), trilateration-based techniques can be used to determine the position vector of point (P) when the position vector of the three points (A_1, A_2, A_3) and three measured wire distances (r_1, r_2, r_3) are known.

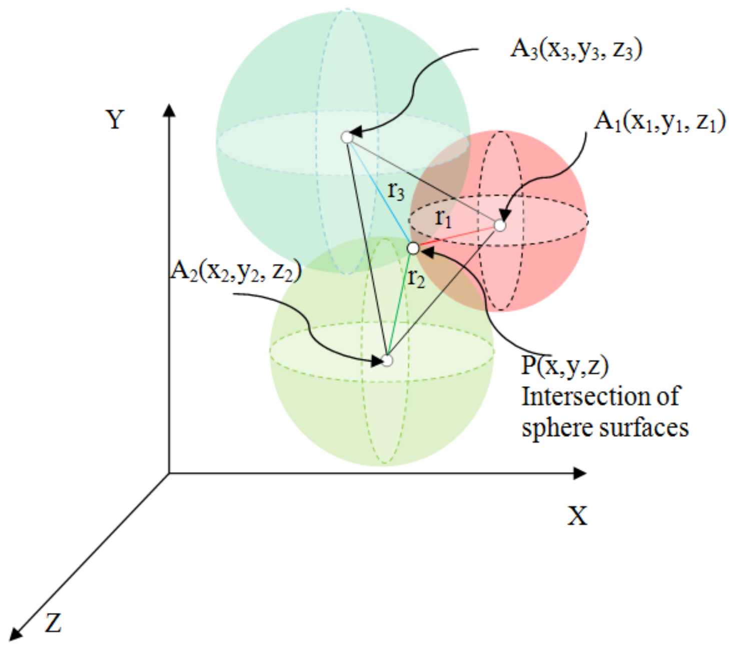


Figure 23. A scheme of trilateration method to determine the coordinates of point P.

Assume that three given spherical center vectors $\mathbf{A}_1 = \{x_1 \;\; y_1 \;\; z_1\}^T$, $\mathbf{A}_2 = \{x_2 \;\; y_2 \;\; z_2\}^T$, $\mathbf{A}_3 = \{x_3 \;\; y_3 \;\; z_3\}^T$ and radii r_1, r_2, and r_3 are known. The equations of the three spheres can be written as

$$(x-x_1)^2 + (y-y_1)^2 + (z-z_1)^2 = r_1^2 \,, \tag{44}$$

$$(x-x_2)^2 + (y-y_2)^2 + (z-z_2)^2 = r_2^2 \,, \tag{45}$$

$$(x-x_3)^2 + (y-y_3)^2 + (z-z_3)^2 = r_3^2 \,. \tag{46}$$

By subtracting Equation (46) from Equation (44) and Equation (46) from Equation (45) as the same principle used in [91], the squares of the unknowns can be eliminated. We obtain

$$c_{11}x + c_{12}y + c_{13}z = b_1 \,, \tag{47}$$

$$c_{21}x + c_{22}y + c_{23}z = b_2 \,, \tag{48}$$

with the following constant coefficients

$c_{11} = 2(x_3 - x_1)$, $\quad c_{21} = 2(x_3 - x_2)$, $\quad b_1 = r_1^2 - r_3^2 - x_1^2 - y_1^2 - z_1^2 + x_3^2 + y_3^2 + z_3^2$,

$c_{12} = 2(y_3 - y_1)$, $\quad c_{22} = 2(y_3 - y_2)$, $\quad b_2 = r_2^2 - r_3^2 - x_2^2 - y_2^2 - z_2^2 + x_3^2 + y_3^2 + z_3^2$,

$c_{13} = 2(z_3 - z_1)$, $\quad c_{23} = 2(z_3 - y_2)$.

Eliminating *z* from Equations (47) and (48) yields

$$x = f(y) = c_1 y + c_2, \tag{49}$$

where the coefficients

$$c_1 = \frac{c_{22} \cdot c_{13} - c_{12} \cdot c_{23}}{c_{11} \cdot c_{23} - c_{21} \cdot c_{13}},\ c_2 = \frac{b_1 \cdot c_{23} - b_2 \cdot c_{13}}{c_{11} \cdot c_{23} - c_{21} \cdot c_{13}}.$$

Substituting Equation (49) into Equation (47) to eliminate *x,* we can obtain

$$z = f(y) = c_3 y + c_4, \tag{50}$$

where the coefficients

$$c_3 = \frac{-c_{11}c_1 - c_{12}}{c_{13}}, \quad c_4 = \frac{b_1 - c_{11}c_2}{c_{13}}.$$

Now substituting Equations (49) and (50) into sphere Equation (44) to eliminate x and z, we achieve a single quadratic in y as

$$ay^2 + by + c = 0, \tag{51}$$

where the coefficients

$$a = c_1^2 + c_3^2 + 1,$$
$$b = 2c_1(c_2 - x_1) - 2y_1 + 2c_3(c_4 - z_1),$$
$$c = (c_2 - x_1)^2 + (c_4 - z_1)^2 + y_1^2 - r_1^2.$$

Two solutions of Equation (51) are

$$y_{\pm} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}. \tag{52}$$

To complete the intersection of the three sphere solution, substitute both positive value y_+ and negative value y_- in Equation (52) into Equations (49) and (50), we obtain

$$x_{\pm} = c_1 y_{\pm} + c_2, \tag{53}$$

$$z_{\pm} = c_3 y_{\pm} + c_4. \tag{54}$$

It should be noted that the singularity problem would happen when the centers of spheres 1 and 3 or spheres 2 and 3 have the same z coordinate, i.e. $z_1 = z_3$ or $z_2 = z_3$,

$$c_{13} = 2(z_3 - z_1) = 0,\ c_{23} = 2(z_3 - z_2) = 0.$$

In the case of the 3-2-1 wire-based pose estimation system in Figure 24, the singularity problem would occur when using the above algorithm to calculate the position value of P_{e1}, because they have the same z coordinate, i.e. $z_1 = z_2 = z_3$ at this configuration. Therefore, the above algorithm can only be used to calculate the position values of P_{e2} and P_{e3}. To solve the singularity problem and obtain the position value of P_{e1}, we can subtract Equation (44) from Equation (45) and Equation (45) from Equation (46) as the same principle used in [92]; then the squares of the unknowns can be eliminated and we obtain

$$a_{11}x + a_{12}y + a_{13}z = t_1, \tag{55}$$

$$a_{21}x + a_{22}y + a_{23}z = t_2, \tag{56}$$

where the constant coefficients are

$$a_{11} = 2(x_1 - x_2)\,, \quad a_{21} = 2(x_2 - x_3), \quad t_1 = r_2^2 - r_1^2 + x_1^2 + y_1^2 + z_1^2 - x_2^2 - y_2^2 - z_2^2,$$
$$a_{12} = 2(y_1 - y_2), \quad a_{22} = 2(y_2 - y_3), \quad t_2 = r_3^2 - r_2^2 + x_2^2 + y_2^2 + z_2^2 - x_3^2 - y_3^2 - z_3^2,$$
$$a_{13} = 2(z_1 - z_2), \quad a_{23} = 2(z_2 - y_3).$$

By eliminating x from Equations (55) and (56), we obtain

$$y = f(y) = a_1 z + a_2, \tag{57}$$

where the coefficients

$$a_1 = \frac{a_{23} \cdot a_{11} - a_{13} \cdot a_{21}}{a_{12} \cdot a_{21} - a_{22} \cdot a_{11}}, \qquad a_2 = \frac{a_{21} \cdot t_1 - a_{11} \cdot t_2}{a_{12} \cdot a_{21} - a_{22} \cdot a_{11}}.$$

Substitute Equation (57) into (55) to eliminate *y,* we obtain

$$x = f(z) = a_3 z + a_4, \tag{58}$$

where the coefficients

$$a_3 = \frac{-a_{12} \cdot a_1 - a_{13}}{a_{11}}, \quad a_4 = \frac{t_1 - a_{12} a_2}{a_{11}}.$$

Now substituting Equations (57) and (58) into Equation (46) to eliminate x and y, we obtain a single quadratic in z only

$$Az^2 + Bz + C = 0, \tag{59}$$

where the coefficients

$$A = a_3^2 + a_1^2 + 1,$$
$$B = 2(a_1 a_2 + a_3 a_4 - a_3 x_3 - a_1 y_3 - z_3)\,,$$
$$C = (a_4 - x_3)^2 + (a_2 - y_3)^2 + z_3^2 - r_3^2.$$

There are two solutions for *z*

$$z_{\pm} = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}. \tag{60}$$

To complete the intersection of the three sphere solution, we can substitute both z_+ and z_- from Equation (60) into Equations (57) and (58)

$$y_{\pm} = a_1 z_{\pm} + a_2, \tag{61}$$

$$x_{\pm} = a_3 z_{\pm} + a_4. \tag{62}$$

The sign ambiguities of the two above mirror solutions can be eliminated by observing the actual reference coordinate system in the real measurement system. This algorithm can only apply to calculating the position value of P_{e1} in our case.

5.2 Measurement Methodology

Similar to the 3-2-1 wire-based pose estimation system proposed in [93], some joints in the end-effector platform coincide as shown in Figure 23 (three wires, r_1, r_2 and r_3, intersect at point P_{e1}; two wires, r_4 and r_5, intersect at point P_{e2}; wire r_6 ended at the point P_{e3}). This configuration can greatly simplify the analysis of forward kinematics for the system. A closed-form forward pose solution can be obtained by solving three consecutive trilaterations according to the equations derived in the Section 5.1.

- Firstly, P_{w1}, P_{w2}, P_{w3} and P_{e1} can define the first tetrahedron; the position value of P_{e1} can be calculated based on Equations 60 through 62 when wire lengths r_1, r_2 and r_3 and the position values of P_{w1}, P_{w2} and P_{w3} are given.
- Secondly, P_{w4}, P_{w5}, P_{e1} and P_{e2} can define the second tetrahedron with another two known wire lengths, r_4, r_5, and the known edge length from P_{e1} to P_{e2}. The position value of P_{e2} can be obtained from Equations 52 through54.
- Finally, P_{w6}, P_{e1}, P_{e2} and P_{e3} can define the last tetrahedron with the known wire length r_6 and the known edge lengths $\overrightarrow{P_{e1}P_{e3}}$ and $\overrightarrow{P_{e2}P_{e3}}$. Under this situation, the position value of P_{e3} can be obtained from Equations (52) through (54). It should be noted that all of the obtained solutions are defined in the same fixed reference frame {w}, and the correct solutions can be obtained by choosing the negative sign in both Equations (52) and (60) for the setup (Figure 24).

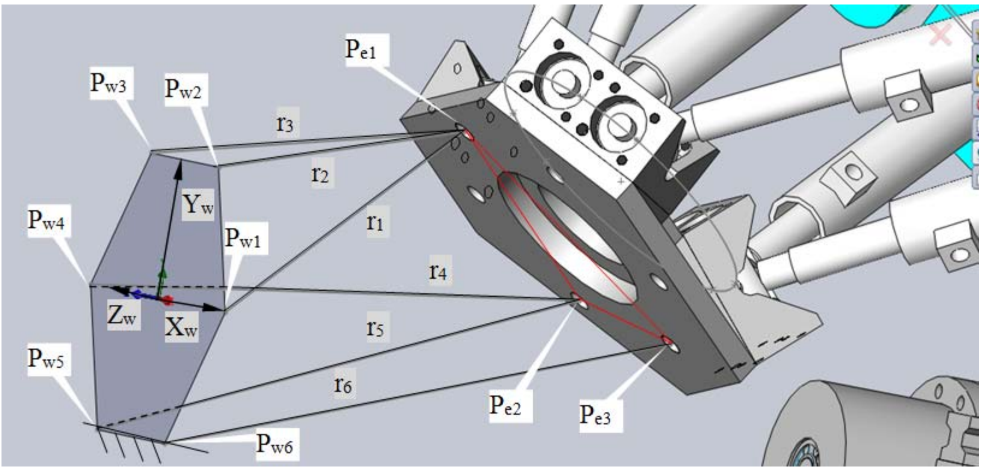


Figure 24. The 3-2-1 wire-based 3D pose estimation system at Solidworks environment.

Denote points P_{e1}, P_{e2} and P_{e2} with respect to the reference frame {w} as ${}^{w}P_{e1}$, ${}^{w}P_{e2}$ and ${}^{w}P_{e2}$, whereas in the end-effector frame {e} as ${}^{e}P_{e1}$, ${}^{e}P_{e2}$ and ${}^{e}P_{e2}$. Then the end-effector pose with respect to the reference frame {w} (denoted as ${}^{w}T_e$) can be calculated according to [94]

$$ {}^{w}T_e \cdot \begin{bmatrix} {}^{e}P_{ei} \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{w}P_{ei} \\ 1 \end{bmatrix},\ \text{i=1, 2, 3.} \tag{63} $$

Furthermore, we also have

$$ {}^{w}T_e \cdot \begin{bmatrix} ({}^{e}P_{e2} - {}^{e}P_{e1}) \times ({}^{e}P_{e3} - {}^{e}P_{e2}) \\ 1 \end{bmatrix} = \begin{bmatrix} ({}^{w}P_{e2} - {}^{w}P_{e1}) \times ({}^{w}P_{e3} - {}^{w}P_{e2}) \\ 1 \end{bmatrix}. \tag{64} $$

Combining Equations (63) and (64) brings

$$
{}^{w}\mathrm{T}_{e}\cdot\begin{bmatrix} {}^{e}p_{e1} & {}^{e}p_{e2} & {}^{e}p_{e3} & ({}^{e}p_{e2}-{}^{e}p_{e1})\times({}^{e}p_{e3}-{}^{e}p_{e2}) \\ 1 & 1 & 1 & 0 \end{bmatrix}=\begin{bmatrix} {}^{w}p_{e1} & {}^{w}p_{e2} & {}^{w}p_{e3} & ({}^{w}p_{e2}-{}^{w}p_{e1})\times({}^{w}p_{e3}-{}^{w}p_{e2}) \\ 1 & 1 & 1 & 0 \end{bmatrix}. \tag{65}
$$

Rewriting Equation (65), we can get the end-effector pose for the moving platform, ${}^{w}\mathrm{T}_{e}$, as

$$
{}^{w}\mathrm{T}_{e}=\begin{bmatrix} {}^{w}p_{e1} & {}^{w}p_{e2} & {}^{w}p_{e3} & ({}^{w}p_{e2}-{}^{w}p_{e1})\times({}^{w}p_{e3}-{}^{w}p_{e2}) \\ 1 & 1 & 1 & 0 \end{bmatrix}\begin{bmatrix} {}^{e}p_{e1} & {}^{e}p_{e2} & {}^{e}p_{e3} & ({}^{e}p_{e2}-{}^{e}p_{e1})\times({}^{e}p_{e3}-{}^{e}p_{e2}) \\ 1 & 1 & 1 & 0 \end{bmatrix}^{-1} \tag{66}
$$

From Equation (66) the measured position and orientation values of the end-effector can be calculated as follows

$$
\begin{cases} {}^{w}\emptyset_{e}^{m} = atan2(r_{21}, r_{11}), \\ {}^{w}\theta_{e}^{m} = \mathrm{atan2}(-r_{31}, r_{11}\cos({}^{w}\emptyset_{e}^{m}) + r_{21}\sin({}^{w}\emptyset_{e}^{m})), \\ {}^{w}\varphi_{e}^{m} = \mathrm{atan2}(r_{13}\sin({}^{w}\emptyset_{e}^{m}) - r_{23}\cos({}^{w}\emptyset_{e}^{m}), -r_{12}\sin({}^{w}\emptyset_{e}^{m}) + r_{22}cos({}^{w}\emptyset_{e}^{m})), \\ {}^{w}p_{ex}^{m} = r_{14}, \\ {}^{w}p_{ey}^{m} = r_{24}, \\ {}^{w}p_{ez}^{m} = r_{34}, \end{cases} \tag{67}
$$

where r_{ij} represents the elements of the ith row and jth column in the end-effector pose matrix ${}^{w}\mathrm{T}_{e}$; atan2(y, x) denotes the four quadrant arctangent of the real parts of elements x and y.

5.3 Simulation Results

In this section, experimental validations are simulated in the Solidworks environment. The measured data were obtained by manually adjusting the displacements of the 10 actuated-joints of the hybrid robot and measuring the corresponding 3-2-1 wire lengths for each configuration in the Solidworks environment. The default measurement precision settings (decimal places: 2) are used in our Solidworks CAD model, so the accuracy of the 3-2-1 pose estimation system will be in a range of ±0.1 mm. Furthermore, we allow a 1 mm assembly error along the X direction of the reference frame {w} for the second joint (q_2) of the hybrid IWR robot but keep the other geometric parameter values unchanged, which means there are no manufacturing errors and we only need to identify 60 parameter errors which are affected by one assembly error. The POE-based model in Section 3.2 and the DE-based identification method in Section 3.3 were employed to identify the 60 parameter errors. The detailed simulation procedures are as follows:

1) Set up a hexagonal platform as the world reference frame {w}. The coordinate values of the six hexagon vertex points (P_{w1}, P_{w2}, P_{w3}, P_{w4}, P_{w5} and P_{w6}) and the three end-effector points (P_{e1}, P_{e2} and P_{e3}) coordinate values are listed in Table 10.

Table 10. The hexagon vertex coordinate values with respect to the reference frame and three end-effector points coordinate values with respect to the moving platform

Symbols	x,y,z coordinate values (mm)		Symbols	x,y,z coordinate values (mm)
P_{e1}	$[0\ \ 120\ \ 0]^T$		P_{w3}	$[-65\ \ 112.58\ \ 0]^T$
P_{e2}	$[-130.92\ \ -60\ \ 0]^T$		P_{w4}	$[-130\ \ 0\ \ 0]^T$
P_{e3}	$[0\ \ -120\ \ 0]^T$		P_{w5}	$[-65\ -112.58\ \ 0]^T$
P_{w1}	$[130\ \ 0\ \ 0]^T$		P_{w6}	$[65\ \ -112.58\ \ 0]^T$
P_{w2}	$[65\ \ 112.58\ \ 0]^T$			

2) Randomly adjust the actuated-joint displacements in the Solidworks environment so as to form a set of 45 pose configurations; the obtained data in each pose configuration (Table 11) can be regarded as the measured transducer readings from actuated-joints.

Table 11. Measured actuated-joint displacements in the Solidworks environment

No. (*j*)	$q_{1,j}^m$ (mm)	$q_{2,j}^m$ (mm)	$q_{3,j}^m$ (°)	$q_{4,j}^m$ (°)	$d_{1,j}^m$ (mm)	$d_{2,j}^m$ (mm)	$d_{3,j}^m$ (mm)	$d_{4,j}^m$ (mm)	$d_{5,j}^m$ (mm)	$d_{6,j}^m$ (mm)
1	0	0	0	47	350	350	350	350	350	350
2	0	-30	0	47	350	350	350	350	350	350
3	0	0	0	47	396	396	350	350	350	350
4	0	-67	0	57	366	378	424	374	374	386
5	0	-67	0	57	396	378	424	374	374	386
6	0	-67	0	57	396	406	424	374	374	386
7	0	-67	0	57	396	406	424	374	406	386
8	0	-130	0	57	396	406	424	374	406	386
9	0	-130	0	54	396	406	424	374	406	386
10	0	-130	0	54	396	406	424	416	406	386
11	0	-130	0	54	396	406	424	416	436	386
12	0	-130	0	54	436	406	424	416	436	386
13	0	-130	0	54	466	406	424	416	436	386
14	0	-130	0	54	466	436	424	416	436	386
15	0	-130	0	54	466	466	424	416	436	386
16	0	-130	0	54	466	466	424	416	436	436
17	0	-130	0	54	466	466	424	446	436	436
18	0	-40	0	54	466	466	424	446	436	436
19	0	-40	0	54	496	466	424	446	436	436
20	0	-40	0	50	496	466	424	446	436	436
21	0	-40	0	50	496	506	424	446	436	436
22	0	-40	0	50	526	506	424	446	436	436
23	0	-40	0	50	526	506	424	446	436	476

No. (*j*)	$q_{1,j}^m$ (mm)	$q_{2,j}^m$ (mm)	$q_{3,j}^m$ (°)	$q_{4,j}^m$ (°)	$d_{1,j}^m$ (mm)	$d_{2,j}^m$ (mm)	$d_{3,j}^m$ (mm)	$d_{4,j}^m$ (mm)	$d_{5,j}^m$ (mm)	$d_{6,j}^m$ (mm)
24	0	-140	0	50	506	506	424	446	436	476
25	0	-140	0	50	546	506	424	446	436	476
Data from number1 to 25 are used for identification and the rest data are used for verification										
26	0	-80	0	45	486	506	424	416	466	476
27	0	-80	0	45	486	506	456	416	466	476
28	0	-80	0	45	486	506	456	451	466	476
29	0	-80	0	45	486	506	481	451	466	476
30	0	-80	0	45	486	506	481	451	506	476
31	0	-30	0	43	486	506	481	451	506	476
32	0	-30	0	43	516	506	481	451	506	476
33	0	-30	0	43	516	506	481	486	506	476
34	0	-30	0	43	516	536	481	486	506	476
35	0	-30	0	43	516	536	516	486	506	476
36	0	-30	0	43	516	536	516	486	531	476
37	0	-30	0	43	516	536	516	486	531	511
38	0	-30	0	43	541	536	516	486	531	511
39	0	-30	0	43	566	536	516	486	531	511
40	0	-30	0	43	566	536	516	486	556	511
41	0	-30	0	43	566	536	546	486	556	511
42	0	-55	0	43	566	536	546	486	556	511
43	0	-55	0	43	526	536	546	486	556	511
44	0	-55	0	43	526	536	511	486	556	511
45	0	-55	0	43	526	536	491	486	516	511

3) Furthermore, the wire lengths in each pose configuration are also recorded (Table 12). Based on these wire lengths, the 3-2-1 pose estimation method can be employed to calculate the end-effector poses, and results are listed in Table 12.

Table 12. Measured wire lengths in the Solidworks model and the corresponding calculated end-effector poses based on the 3-2-1 pose estimation method

r_1 (mm)	r_2 (mm)	r_3 (mm)	r_4 (mm)	r_5 (mm)	r_6 (mm)	${}^0\phi_e^m$ (rad.)	${}^w\theta_e^m$ (rad.)	${}^w\psi_e^m$ (rad.)	${}^wP_{ex}^m$ (mm)	${}^wP_{ey}^m$ (mm)	${}^wP_{ez}^m$ (mm)
411	317.67	317.26	419.56	479.55	495.81	0.00	0	0.82	-1.01	267.08	-378.58
429.68	341.49	341.11	446.83	503.58	521.61	0.00	0	0.82	-0.98	267.09	-408.57
394.12	312.59	312.17	425.33	473.99	496.74	0.00	0	0.83	-1.00	221.91	-402.24
326.61	274.27	285.1	439.85	477.86	541.09	0.06	0.12	1.31	4.95	182.89	-399.71
324.99	280.08	294.87	443.94	482.36	541.63	-0.10	0.15	1.27	15.67	167.68	-409.12
329.08	288.59	299.07	449.58	479.46	544.49	0.06	0.13	1.28	5.22	151.64	-417.64

r_1 (mm)	r_2 (mm)	r_3 (mm)	r_4 (mm)	r_5 (mm)	r_6 (mm)	${}^{0}\phi_e^m$ (rad.)	${}^{w}\theta_e^m$ (rad.)	${}^{w}\psi_e^m$ (rad.)	${}^{w}P_{ex}^m$ (mm)	${}^{w}P_{ey}^m$ (mm)	${}^{w}P_{ez}^m$ (mm)
340	296.27	303.09	427.98	458.59	533.3	0.07	0.25	1.15	-18.80	140.80	-423.00
395.34	358.42	364.09	490.58	517.5	593.19	0.07	0.25	1.15	-18.77	140.84	-485.99
399.72	352.64	358.39	480.34	513.73	584.97	0.05	0.26	1.10	-18.79	167.65	-476.80
401.31	351.92	360.33	476.54	502.24	550.82	0.06	0.12	0.90	8.97	153.46	-476.31
416.09	364.54	369.27	454.19	482.38	541.68	0.06	0.26	0.80	-18.06	146.72	-479.71
412.54	372.19	387.88	466.52	494.61	546.95	-0.16	0.25	0.76	7.47	130.10	-493.05
422.12	389.6	413.83	477.7	506.26	550.26	-0.34	0.24	0.68	28.60	120.46	-508.26
425.12	394.75	410.52	487.11	507.23	566.44	-0.15	0.27	0.76	8.23	102.15	-518.33
439.15	409.71	416.89	503.95	516.56	584.15	0.04	0.28	0.81	-12.21	86.44	-533.51
397.82	363.32	357.73	500.34	517.83	566.7	0.00	0.03	0.98	-19.81	94.23	-489.76
401.49	366.45	363.63	507.05	519.24	549.03	0.00	-0.10	0.87	5.09	86.61	-491.74
323.5	278.83	275.12	417.26	431.99	460.52	0.00	-0.10	0.87	5.10	86.60	-401.75
323.94	290.01	298.63	433.16	447.6	474.43	-0.18	-0.09	0.86	26.11	72.62	-418.75
328.55	277.99	286.98	413.38	437.27	454.37	-0.17	-0.10	0.79	26.13	107.15	-401.99
347.19	300.36	292.47	428.88	442.39	468.15	0.05	-0.11	0.80	-1.95	87.63	-418.66
346.33	310.39	315.14	446.33	459.22	484.15	-0.12	-0.10	0.80	19.89	73.21	-436.23
328.51	286.16	273.83	442.41	458.42	471.63	-0.11	-0.26	0.97	6.43	82.30	-401.80
414.31	375.83	360.58	530.23	543.83	557.97	0.01	-0.27	0.96	-6.70	94.05	-490.19
419.51	392.37	389.62	555.79	568.61	583.28	-0.24	-0.25	0.97	20.71	71.21	-516.29
The above data are used for identification purpose and the below data for verification											
387.55	314.41	282.17	412.86	451.32	486.41	0.11	-0.02	0.84	-68.22	153.53	-404.85
349.8	275.62	253.3	389.23	432.9	484.83	0.09	0.10	0.99	-58.13	166.16	-378.38
351.69	276.9	257.96	385.08	420.36	452.67	0.12	-0.02	0.83	-32.07	153.74	-377.94
324.12	247.8	239.38	366.5	406.45	451.45	0.10	0.08	0.94	-24.07	165.13	-357.44
344.91	268.07	253.13	343.77	390.34	448.93	0.06	0.27	0.82	-62.34	154.82	-364.85
325.82	231.27	213.78	291.42	352.03	402.13	0.05	0.27	0.78	-62.32	175.16	-309.09
304.67	220.29	218.25	298.62	357.01	403.22	-0.12	0.26	0.77	-42.15	157.69	-317.86
306.95	221.69	225.77	294.7	343.45	367.2	-0.08	0.12	0.63	-12.03	146.38	-315.22
320.09	238.57	224.94	301.67	339.22	375.36	0.09	0.13	0.63	-34.96	128.33	-329.79
282.55	198.47	204.97	274.78	319.71	371.99	0.05	0.27	0.78	-22.91	143.37	-301.57
298.59	216.68	217.26	261.57	311.63	375.6	0.03	0.42	0.71	-49.69	137.43	-308.70
290.31	192.16	171.1	260.24	319.05	366.91	0.05	0.22	0.81	-55.98	155.94	-274.89
270.32	181.56	176.92	266.54	323.01	368.48	-0.10	0.22	0.80	-38.61	140.34	-282.96
256.1	180.21	193.3	273.45	327.47	370.93	-0.25	0.23	0.77	-20.49	124.98	-294.41
268.62	195.57	203.59	260.7	320.67	376.68	-0.28	0.37	0.69	-46.27	120.77	-300.18
243.3	170.05	195.64	242.73	314.9	378.13	-0.37	0.50	0.78	-36.27	140.59	-276.60

r_1 (mm)	r_2 (mm)	r_3 (mm)	r_4 (mm)	r_5 (mm)	r_6 (mm)	${}^{0}\phi_e^m$ (rad.)	${}^{w}\theta_e^m$ (rad.)	${}^{w}\psi_e^m$ (rad.)	${}^{w}P_{ex}^m$ (mm)	${}^{w}P_{ey}^m$ (mm)	${}^{w}P_{ez}^m$ (mm)
260.62	194.02	216.8	265.96	333.14	398.92	-0.37	0.50	0.78	-36.28	140.60	-301.59
280.04	191.94	191.3	260.6	330.28	399.93	-0.11	0.46	0.88	-63.15	163.78	-284.94
315.59	229.82	209.7	280.71	338.54	396.32	-0.03	0.34	0.72	-77.76	141.95	-313.63
314.5	229.14	206.27	314.15	357.67	389.66	0.04	0.04	0.75	-41.98	140.41	-320.70

4) Take the 60 parameter errors in Equation (35) as decision variables, and substitute the end-effector poses from number 1 to 25 in Table 12 into Equation (36) in order to calculate the predicted leg lengths $d_{i,j}^{p}$. The identification task employs the DE algorithm to search for an optimal combination of parameter errors to minimize the differences between the measured leg lengths $d_{i,j}^{\mathrm{m}}$ and the predicted leg lengths $d_{i,j}^{p}$ under certain given program terminal conditions (e.g. maximum generations G_{max} and/or the objective function value). The identification results are listed in Table 13.

Table 13. Nominal and identified parameters of the hybrid IWR robot (unit: mm)

No.	Symbols	Nominal values	Identified values	No.	Symbols	Nominal values	Identified values
1	υ_{1x}	1	0.6043	31	a_{55x}	-316.6	-317.1
2	υ_{1y}	0	0.3021	32	a_{55y}	-84.67	-85.17
3	ν_{1z}	0	0.333	33	a_{55z}	0	-0.36
4	υ_{2x}	0	-0.0002	34	a_{56x}	-231.6	-232.1
5	υ_{2y}	0	0	35	a_{56y}	-231.9	-232.4
6	ν_{2z}	1	1.0003	36	a_{56z}	0	-0.5
7	ω_{3x}	0	0.3064	37	b_{t1x}	32.5	32.5363
8	ω_{3y}	1	0.7216	38	b_{t1y}	-125.9	-126.066
9	ω_{3z}	0	-0.4503	39	b_{t1z}	0	0.3286
10	υ_{3x}	628	627.5154	40	b_{t2x}	125.3	125.2117
11	υ_{3y}	0	0.2527	41	b_{t2y}	34.8	34.6732
12	ν_{3z}	0	-0.4971	42	b_{t2z}	0	0.4921
13	ω_{4x}	1	1.0007	43	b_{t3x}	92.8	92.9497
14	ω_{4y}	0	0.001	44	b_{t3y}	91.1	91.3804
15	ω_{4z}	0	0.0006	45	b_{t3z}	0	0.2824
16	υ_{4x}	0	-0.1154	46	b_{t4x}	-92.8	-92.7342
17	υ_{4y}	-376	-376.466	47	b_{t4y}	91.1	91.1808
18	ω_{4z}	0	-0.5	48	b_{t4z}	0	0.0575
19	a_{51x}	231.6	231.6594	49	b_{t5x}	-125.3	-125.374
20	a_{51y}	-231.9	-232.4	50	b_{t5y}	34.8	34.8061
21	a_{51z}	0	-0.5	51	b_{t5z}	0	-0.225

No.	Symbols	Nominal values	Identified values	No.	Symbols	Nominal values	Identified values
22	a_{52x}	316.6	316.1	52	b_{t6x}	-32.5	-32.631
23	a_{52y}	-84.67	-85.17	53	b_{t6y}	-125.9	-125.857
24	a_{52z}	0	-0.5	54	b_{t6z}	231.6	231.4647
25	a_{53x}	85	85.4793	55	δd_1	0	0.0235
26	a_{53y}	316.58	316.4412	56	δd_2	0	-0.0201
27	a_{53z}	0	-0.5	57	δd_3	0	-0.5
28	a_{54x}	-85	-84.6441	58	δd_4	0	-0.5
29	a_{54y}	316.58	316.1839	59	δd_5	0	-0.5
30	a_{54z}	0	-0.5	60	δd_6	0	-0.2872

5) After the parameter errors are identified, we can use the rest of the end-effector poses from number 26 to 45 in Table 12 to verify the validity of identified results. Firstly, the leg length values before calibration are calculated by using these end-effector poses under an ideal condition where the identified parameter errors are not included. Secondly, the same end-effector data set is used to calculate a set of actual leg lengths by considering the identified parameter errors. The calculated leg lengths before and after calibration are listed in Table 14. Leg errors, before and after calibration, are shown in Figures 25 and 26.

Table 14. Leg lengths before calibration (superscript *b* denotes 'before') and after calibration (superscript *a* denotes 'after')

$d^b_{1,j}$ (mm)	$d^b_{2,j}$ (mm)	$d^b_{3,j}$ (mm)	$d^b_{4,j}$ (mm)	$d^b_{5,j}$ (mm)	$d^b_{6,j}$ (mm)	$d^a_{1,j}$ (mm)	$d^a_{2,j}$ (mm)	$d^a_{3,j}$ (mm)	$d^a_{4,j}$ (mm)	$d^a_{5,j}$ (mm)	$d^a_{6,j}$ (mm)
485.56	505.60	422.66	414.62	465.65	475.66	485.98	505.97	424.12	416.05	466.08	476.07
485.47	505.48	454.63	414.60	465.66	475.61	485.97	505.99	456.14	416.06	466.12	476.11
485.45	505.45	454.60	449.57	465.53	475.53	485.99	505.96	456.08	451.02	466.12	476.10
485.37	505.36	479.59	449.57	465.52	475.49	485.96	505.98	481.11	451.05	466.12	476.12
485.50	505.42	479.63	449.65	505.66	475.63	485.98	505.96	481.11	451.07	506.14	476.12
485.51	505.42	479.66	449.69	505.70	475.67	485.98	505.96	481.12	451.08	506.15	476.13
515.47	505.37	479.63	449.69	505.68	475.60	515.99	505.98	481.12	451.08	506.12	476.10
515.42	505.38	479.61	484.65	505.53	475.50	515.98	505.97	481.07	486.04	506.10	476.09
515.49	535.49	479.68	484.67	505.54	475.58	515.95	535.96	481.07	486.03	506.09	476.09
515.42	535.36	514.64	484.66	505.57	475.54	515.97	536.01	516.10	486.05	506.13	476.13
515.53	535.38	514.66	484.72	530.67	475.66	515.99	535.97	516.07	486.06	531.16	476.13
515.38	535.38	514.69	484.69	530.57	510.55	515.93	535.97	516.15	486.08	531.09	511.11
540.39	535.35	514.66	484.69	530.59	510.52	540.96	536.00	516.14	486.08	531.09	511.10
565.42	535.31	514.62	484.69	530.62	510.51	566.01	535.99	516.12	486.08	531.11	511.10
565.44	535.31	514.64	484.73	555.67	510.57	565.96	535.96	516.12	486.09	556.09	511.06
565.41	535.21	544.62	484.73	555.68	510.56	565.97	535.99	546.12	486.11	556.10	511.08
565.40	535.21	544.63	484.73	555.68	510.55	565.96	535.99	546.13	486.12	556.11	511.08

$d^b_{1,j}$ (mm)	$d^b_{2,j}$ (mm)	$d^b_{3,j}$ (mm)	$d^b_{4,j}$ (mm)	$d^b_{5,j}$ (mm)	$d^b_{6,j}$ (mm)		$d^a_{1,j}$ (mm)	$d^a_{2,j}$ (mm)	$d^a_{3,j}$ (mm)	$d^a_{4,j}$ (mm)	$d^a_{5,j}$ (mm)	$d^a_{6,j}$ (mm)
525.44	535.28	544.66	484.75	555.70	510.61		525.99	536.00	546.13	486.13	556.16	511.14
525.49	535.41	509.70	484.75	555.67	510.64		525.96	535.96	511.13	486.10	556.12	511.10
525.43	535.43	489.67	484.66	515.56	510.55		525.98	535.94	491.11	486.09	556.13	511.11

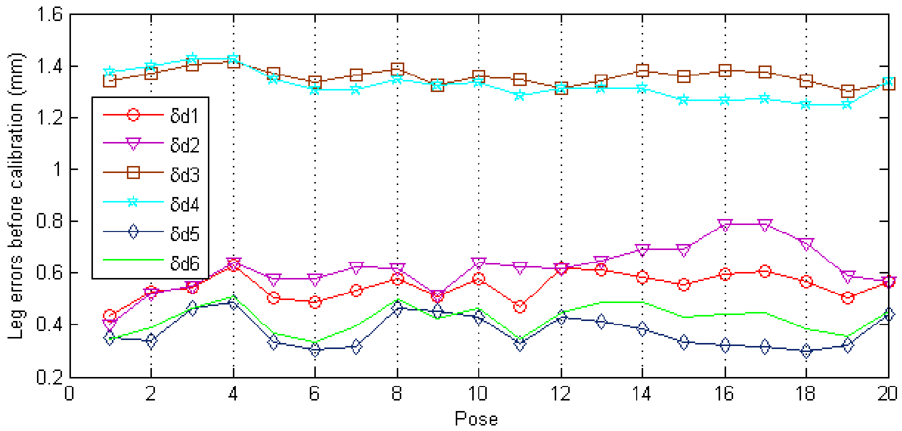


Figure 25. Leg errors before calibration in 20 pose configurations.

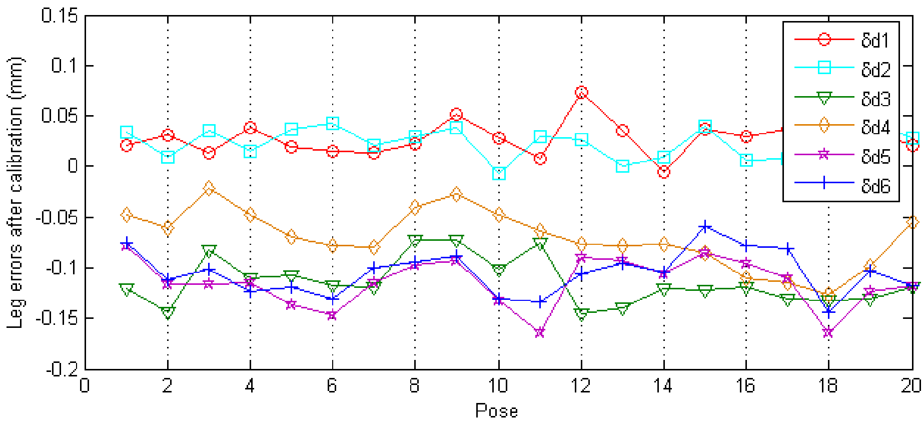


Figure 26. Leg errors after calibration in 20 pose configurations.

The results of leg errors before and after calibration in Figures 25 and 26 show that the errors can be reduced by at least one order of magnitude, i.e., from about 1.5 mm to less than 0.15 mm.

To accurately show improvement in the end-effector poses, we can assume the measured end-effector values, which should be achieved by the control program, to be the desired ones, and compare these values with the numerically calculated end-effector poses to get the orientation errors and the position errors under two different conditions (with or without identified parameter errors). The POE error model in Section 3.2 and the DE identification method in Section 3.3 can be used to calculate the end-effector poses, but the 60 error parameter variables should be replaced by the three orientation Euler angles and the three position vectors of the end-effector pose. The end-effector pose values after calibration can be obtained by including the identified parameter errors in the error model, whereas the values before calibration can be obtained without considering the identified parameter errors in the error model. Now the task of parameter identification is to search for a set of end-effector poses $\theta = \left({}^{w}\phi_e, {}^{w}\theta_e, {}^{w}\varphi_e, {}^{w}P_{ex}, {}^{w}P_{ey}, {}^{w}P_{ez}\right)$ to minimize

$$SS_\theta = \sum_{i=1}^{6}\left(d_{i,j}^{m} - d_{i,j}^{p}\right)^2, \tag{68}$$

where $d_{i,j}^{m}$ is the measured leg length as in Table 11; $d_{i,j}^{p}$ is the leg length predicted by Equations (33) and (34); the measured end-effector pose g_{st}^{m} is replaced by a homogeneous transformation matrix which involves variables of the three Z-Y-X Euler angles and the three position vectors

$${}^{w}T_{\mathrm{e}} = \begin{bmatrix} \cos(\phi)\cos(\theta) & \cos(\phi)\sin(\theta)\sin(\psi) - \sin(\phi)\cos(\psi) & \cos(\phi)\sin(\theta)\cos(\psi) + \sin(\phi)\sin(\psi) & p_x \\ \sin(\phi)\cos(\theta) & \sin(\phi)\sin(\theta)\sin(\psi) + \cos(\phi)\cos(\psi) & \sin(\phi)s(\theta)\cos(\psi) + \cos(\phi)\sin(\psi) & p_y \\ -\sin(\theta) & \cos(\theta)\sin(\psi) & \cos(\theta)\cos(\psi) & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{69}$$

Numerical solutions of the end-effector pose before and after calibration are listed in Table 15. Comparing these results with the measured end-effector poses, we can get the orientation errors before and after calibration (Figures 27-28), and the position errors before and after calibration (Figures 29-30).

Table 15. End-effector poses before calibration (superscript *b* denotes 'before') and after calibration (superscript *a* denotes 'after')

${}^{w}\phi_e^b$ (rad.)	${}^{w}\theta_e^b$ (rad.)	${}^{w}\psi_e^b$ (rad.)	${}^{w}P_{ex}^b$ (mm)	${}^{w}P_{ey}^b$ (mm)	${}^{w}P_{ez}^b$ (mm)		${}^{w}\phi_e^a$ (rad.)	${}^{w}\theta_e^a$ (rad.)	${}^{w}\psi_e^a$ (rad.)	${}^{w}P_{ex}^a$ (mm)	${}^{w}P_{ey}^a$ (mm)	${}^{w}P_{ez}^a$ (mm)
0.11	-0.02	0.84	-67.22	153.10	-403.8		0.11	-0.02	0.84	-68.21	153.46	-405.0
0.09	0.10	0.99	-57.12	165.73	-377.3		0.09	0.10	0.99	-58.08	166.06	-378.6
0.12	-0.02	0.83	-31.06	153.32	-376.9		0.12	-0.02	0.83	-32.00	153.67	-378.1
0.10	0.08	0.94	-23.08	164.69	-356.4		0.10	0.08	0.94	-24.00	165.02	-357.6
0.06	0.27	0.82	-61.32	154.39	-363.8		0.06	0.26	0.82	-62.28	154.73	-365.0
0.05	0.27	0.78	-61.32	174.76	-308.1		0.05	0.27	0.78	-62.27	175.06	-309.3
-0.12	0.26	0.77	-41.15	157.29	-316.8		-0.12	0.26	0.77	-42.12	157.61	-318.0
-0.08	0.12	0.63	-11.03	145.99	-314.2		-0.08	0.12	0.63	-11.97	146.31	-315.4
0.09	0.13	0.63	-33.96	127.93	-328.7		0.09	0.13	0.63	-34.88	128.24	-330.0
0.05	0.27	0.78	-21.90	142.98	-300.5		0.05	0.27	0.78	-22.80	143.29	-301.8
0.03	0.42	0.71	-48.68	137.05	-307.6		0.03	0.42	0.71	-49.59	137.36	-308.9
0.05	0.22	0.81	-55.00	155.51	-273.8		0.05	0.22	0.81	-55.95	155.78	-275.1
-0.10	0.22	0.80	-37.61	139.94	-281.9		-0.10	0.22	0.80	-38.58	140.23	-283.2
-0.25	0.23	0.77	-19.48	124.61	-293.4		-0.25	0.23	0.77	-20.47	124.91	-294.6

${}^{w}\phi_e^b$ (rad.)	${}^{w}\theta_e^b$ (rad.)	${}^{w}\psi_e^b$ (rad.)	${}^{w}P_{ex}^b$ (mm)	${}^{w}P_{ey}^b$ (mm)	${}^{w}P_{ez}^b$ (mm)	${}^{w}\phi_e^a$ (rad.)	${}^{w}\theta_e^a$ (rad.)	${}^{w}\psi_e^a$ (rad.)	${}^{w}P_{ex}^a$ (mm)	${}^{w}P_{ey}^a$ (mm)	${}^{w}P_{ez}^a$ (mm)
-0.28	0.37	0.69	-45.27	120.35	-299.2	-0.28	0.37	0.69	-46.28	120.68	-300.4
-0.37	0.50	0.78	-35.28	140.19	-275.6	-0.37	0.50	0.78	-36.27	140.51	-276.8
-0.37	0.50	0.78	-35.28	140.19	-300.5	-0.37	0.50	0.78	-36.26	140.51	-301.8
-0.11	0.46	0.88	-62.14	163.40	-283.9	-0.11	0.46	0.88	-63.08	163.69	-285.2
-0.03	0.34	0.72	-76.77	141.55	-312.6	-0.03	0.34	0.72	-77.74	141.83	-313.9
0.04	0.04	0.75	-40.96	140.01	-319.6	0.04	0.04	0.75	-41.92	140.30	-320.9

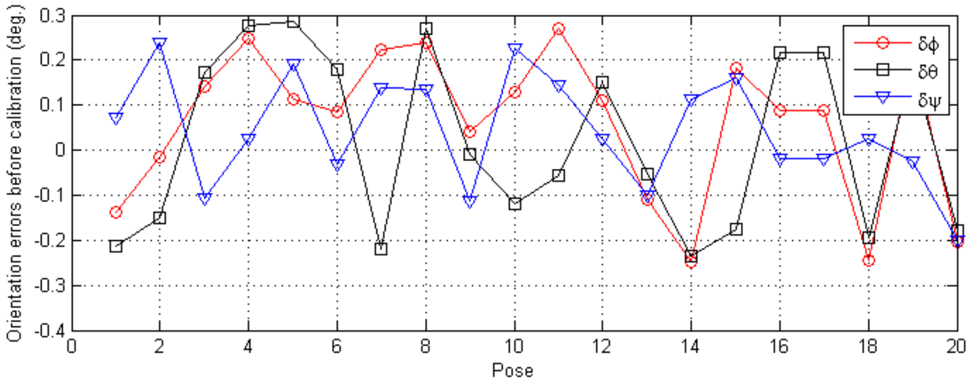


Figure 27. Orientation errors before calibration in 20 pose configurations.

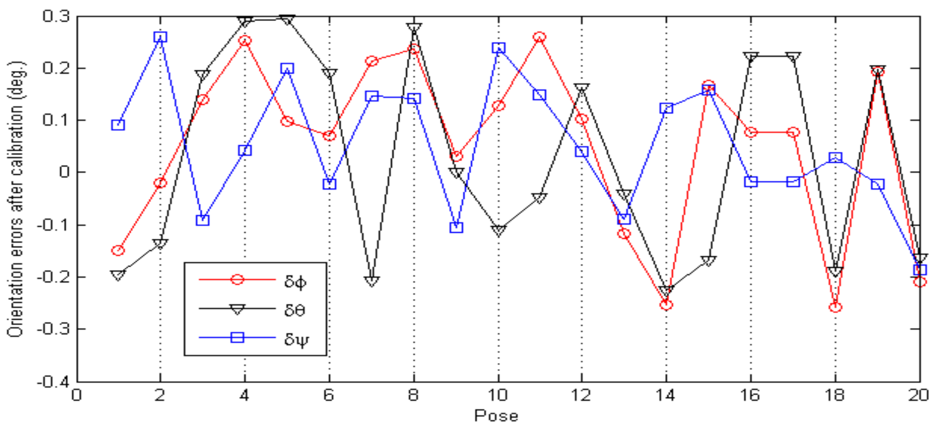


Figure 28. Orientaion errors after calibration in 20 pose configurations.

It can be seen from Figures 27 and 28 that the orientation errors are not reduced after calibration. The simulation results have perfectly reflected our actual error settings where only a 1 mm assembly error along the x-coordinate of world frame {w} is realized in the second joint (q_2) of the hybrid IWR robot. This arrangement guaranteed that the orientation values of the end-effector are not affected by the translational movement of the second joint of the IWR robot. The orientation errors in Figures 27 and 28 are only influenced by the precision level of the 3-2-1 pose estimation system.

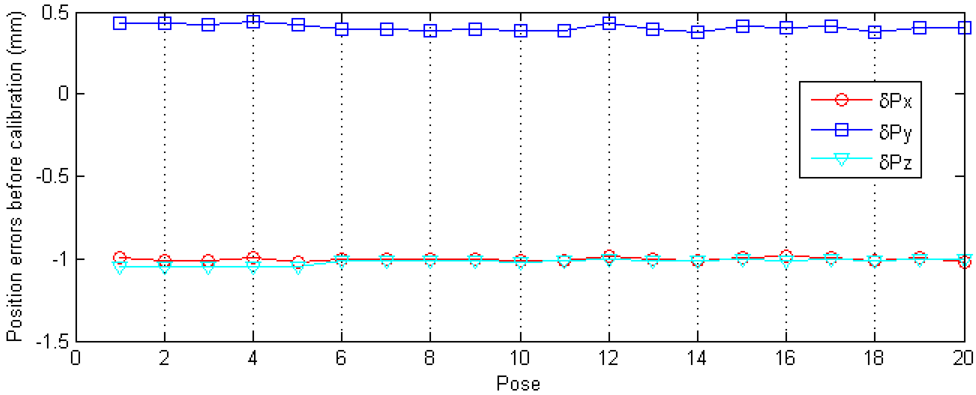


Figure 29. Position errors before calibration in 20 pose configurations.

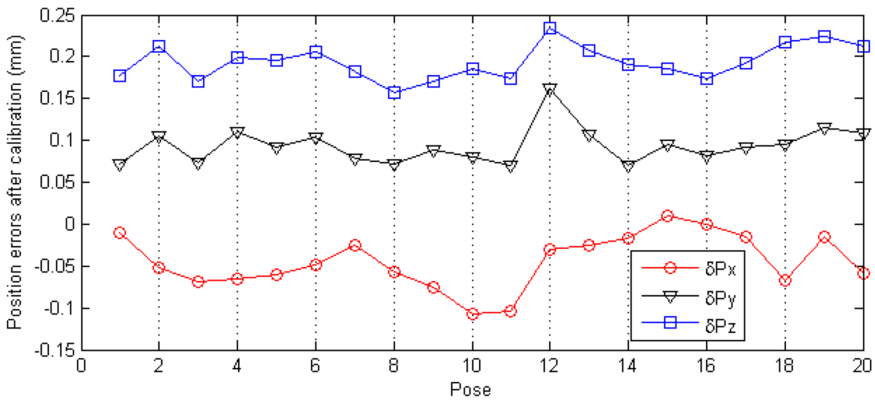


Figure 30. Position errors after calibration in 20 pose configurations

From Figures 29 and 30, it can be seen that the improved accuracy for position errors is assessed and shown to be better by almost one order of magnitude. Before calibration, the biggest position error is about -1.2 mm; but after calibration, the biggest position error is reduced to 0.25 mm, which reaches the precision level as that given by the 3-2-1 measurement system (±0.1 mm).

Compared with the numerical simulation in Chapter 4, the Solidworks simulation in Chapter 5 is closer to real applications. In the Solidworks environment, the different accuracy of the 3-2-1 pose measurement system can be realized by setting different decimal places for the Solidworks measuring precision. Moreover, manufacturing and assembly errors can also be easily realized in the Solidworks CAD model. By comparing the simulation results in Chapters 4 and 5, we can draw the same conclusion, that is, the higher the accuracy of the measurement system we use, the better the identification results we obtain.

CHAPTER 6

CONCLUSIONS

The main purpose of this study is to develop an effective calibration method to improve the accuracy of a redundant 10-DOF serial-parallel robot. To accomplish this, two kinds of error modeling methods and two kinds of parameter identification methods are proposed. Both of the error-modeling methods take into account the geometrical error sources that basically result from machining and assembly processes. The two methods can be regarded as a hybrid calibration method as they integrate both the traditional forward calibration for serial mechanisms and the inverse calibration for parallel mechanisms into one.

For the Denavit-Hartenberg (DH) hybrid model, the DH modeling method is employed to predict a forward solution for a serial mechanism, while the vector chain analytical method is used to develop an inverse solution for the parallel manipulator and the hybrid mechanism. The advantage of this method is that the forward solution of the serial mechanism can be used as a prediction value to fit into the final error model. Therefore, the full pose measurement of the end-effector for the hybrid robot can meet the calibration requirement effectively, while the pose measurement of the end-tip for a serial mechanism is unnecessary. The identification of unknown parameter errors involves using a powerful global optimization method – the differential-evolution (DE) algorithm. Computer simulations of the serial-parallel IWR robot demonstrate that all of the 54 geometrical parameter errors can be successfully identified. The simulation results show that the DE-based parameter identification algorithm has a very strong stochastic searching ability. It is very robust and effective and can easily be employed to identify multi-dimensional parameter errors for high nonlinear kinematic models. The simulation is also helpful to find out the most suitable DE algorithm control parameters and termination conditions before carrying out an experimental test. By using the DE-based identification method, all the parameter errors can be identified even if correlations between parameter errors exist.

However, to get a more accurate error model, redundant parameters which result in correlations have to be eliminated. To solve this problem, the MCMC-based method has been proposed for parameter correlation analysis as well as for parameter estimation in a statistical way. The simulation results for the reduced error model with measurement noise show that all the independent and identifiable parameters have successfully converged to assumed errors with only a slight difference and the standard deviations arrive at very high precisions (10^{-5} mm and 10^{-8} rad.). Another advantage of using the MCMC approach is that it is able to lower the influence of measurement noise to as small as possible. The limitation of the MCMC approach is time-consuming, thus a powerful CPU processor and RAM are required for the simulation computer.

For the product-of-exponential (POE) error model, solving forward kinematics of the Hexa-WH parallel manipulator is also a very difficult problem due to its high dimensional (60 error parameters) and sophisticated constraints. However, the inverse kinematics problem for this parallel manipulator is very simple. The solution is based on the geometry of the manipulator and it can also be derived in a manner similar to that used for solving subproblems in [95]. In our POE-based calibration model, the forward kinematics for a serial mechanism and the inverse kinematics for a parallel mechanism are integrated together to form a hybrid calibration model. The parameter errors derived by the POE modeling method are independent and identifiable, so the DE-based identification method can satisfy the requirements of identification effectively. Simulation results show that the accuracy of the end-effector can be improved to the same precision level as that given by the external measurement device. The higher accuracy the measurement system has, the better the identification results that can be achieved. For instance, the simulation in Section 4.3 shows that the RMS position error after calibration is about 0.001 mm which matches the assumed laser tracker precision (position accuracy: ±0.01 mm, standard deviation: 0.003 mm), while the simulation in Section 5.3 also demonstrates that the orientation error (0.3^o) and position error (0.12 mm) after calibration reached the simulated precision of the 3-2-1 pose estimation system in Solidworks, i.e. ±0.1 mm for position accuracy, $\pm 0.3^o$ for orientation accuracy.

Finally, the Solidworks CAD prototype model and a 3-2-1 wire-based pose measurement system are employed to simulate the working environment of the hybrid IWR robot. Calibration results for leg length errors show that kinematic errors can be reduced by at least one order of magnitude. Calibration results for end-effector poses also show that position errors can be reduced by about one order of magnitude. To prove its feasibility, our future work will focus on experimentally validating the proposed methods on the current IWR robotic system, and extending the proposed method to other serial-parallel robots or parallel robots.

REFERENCES

[1] http://www.iter.org/industry. (Retrieved on Aug. 25, 2012).

[2] http://www.iter.org/mach/vacuumvessel. (Retrieved on Aug. 25, 2012).

[3] H. Wu, H. Handroos, P. Pessi, J. Kilkki and L. Jones. Development and control towards a parallel water hydraulic weld/cut for machining processes in ITER Vacuum Vessel. *International Journal of Fusion Engineering and Design,* 75−79, pp. 625−631, 2005.

[4] Mohamed Abderrahim, Alla Khamis, Santiago Garrido, Luis Moreno. Accuracy and calibration issues of industrial manipulators. Available from: http://cdn.intechopen.com/pdfs/259/InTech-Accuracy_and_calibration_issues_of_industrial_manipulators.pdf (Retrieved on Oct. 25, 2012).

[5] C. Mavroidis, S. Dubowsky, P. Drouet, J. Hintersteiner and J. Flanz. A systematic error analysis of robotic manipulators: application to a high performance medical robot. *Proceedings of the 1997 International Conference in Robotics and Automation,* Albuquerque, New Mexico, pp. 980-985, April, 1997.

[6] A. Y. Elatta, Li Pei Gen, Fan liang Zhi et al. An overview of robot calibration. *Information Technology Journal 3 (1)*: pp.74-78, 2004

[7] Edited by R. Bernhardt, S. L. Albright. Robot calibration. (Chapman and Hall , London,1993)

[8] B. W. Mooring, Z. S. Roth and M. Driels, The fundamentals of manipulator calibration (John Wiley & Sons, New York. 1991).

[9] H. Zhuang. Self-calibration of parallel mechanisms with a case study on Stewart platform. *IEEE Transactions on Robotics and Automation*, 13(3): pp.387-397, 1997.

[10] G. Yang, I. M. Chen, W.K. Lim, and S.H. Yeo. Self-calibration of three-legged modular reconfigurable parallel robots based on leg-end distance errors. *Robotica*, Vol. 19, pp.187-198, 2001.

[11] H. Zhuang, L. Liu, O. Masory. Autonomous calibration of hexapod machine tools. *ASME transactions on Journal of Manufacturing Science and Engineering,* Vol. 122, pp.140−148, Feb. 2000.

[12] M. Gao, T. Li, W. Yin. Calibration method and experiment of Stewart platform using a laser tracker. *Proc. of IEEE International Conference on Systems, Man and Cybernetics,* Vol. 3, pp. 2797–2802, Oct. 2003.

[13] S. Besnard, W. Khalil, Calibration of parallel robots using two inclinometers. *Proc. of the 1999 IEEE International Conference on Robotics & Automation*, Detroit, Michigan, pp.1758−1763, 1999.

[14] P. Renaud and N. Andreff et al, Vision-based kinematic calibration of a H4 parallel mechanism. *Proc. Of IEEE International Conference on Robotics and Automation,* Taipei, Taiwan, pp.1191−1196, 2003.

[15] L. J. Everett, Morris Driels, and B. W. Mooring. Kinematic modeling for robot calibration. *In Proceedings of 1987 IEEE International Conference on Robotics and Automation*, Texas, USA, pp. 183-189, Mar. 1987.

[16] K. Schr¨oer, S. Albright, and M. Grethlein. Complete, minimal and model continuous kinematic models for robot calibration. *Robot. Comput.-Integr. Manuf.,* vol. 13, no. 1, pp. 73–85, 1997.

[17] W. Khalil, M. Gautier. Calculation of the identifiable parameters for robot calibration. *9th IFAC/IFOR Symp. on Identif. and Sys*. Parameter Estimation, pp. 888-892, Budapest, Hungaria, 1991.

[18] S.Besnard, W. Khalil. Identifiable Parameters for Parallel Robots Kinematic Calibration. *Proceedings of the 2001 IEEE International Conference on Robotics & Automation,* pp. 2859-2865, Seoul, Korea, May 21-26, 2001.

[19] P. Vischer. Improving the Accuracy of Parallel Robots. *PhD thesis,* No.1570 (Ecole Polytechnique fédérale de Lausanne EPFL, 1996)

[20] R. Paul. Robot Manipulators: Mathematics, Programming, and Control. (Cambridge, MA: MIT Press, 1981).

[21] S. Hayati, M. Mirmirani, Improving the absolute positioning accuracy of robot manipulators, *Journal of Robotic Systems,* 2(4), pp. 397-413, 1985.

[22] W. K. Veitschegger, C-H Wu, Robot accuracy analysis based on kinematics, *IEEE Journal of Robotics and Automation, RA-2(3),* September, pp.171-179, 1986.

[23] H. W. Stone, A. C. Sanderson, A prototype arm signature identification system, *in Proc. IEEE Int. Conf. Robotics and Automation*, pp.175-182, Mar. 1987.

[24] B. W. Mooring, G. R. Tang. An improved method for identifying the kinematic parameters in a six axis robot. *Proceedings of the 1983 ASME Computers in Engineering Conference*, Las Vegas, Nevada, pp.79-84, 1983.

[25] K. Okamura and F. C. park, Kinematic Calibration Using the Product of Exponentials Formula, *Robotica*, Vol. 14, pp. 415-421, 1996.

[26] Koichiro Okamura, F. C. Park. Kinematic calibration using the product of exponentials formula. *Robotica,* Vol. 14. pp. 415-421, 1996.

[27] B. Paden, S. S. Sastry, Optimal kinematic design of 6r manipulators, *The International Journal of Robotics Research,* 7 (2), pp. 43 – 61, 1988.

[28] F. C. Park, D. J. Pack. Motion control using the product-of-exponentials kinematic equations*. IEEE International Conference on Robotics and Automation,* pp.2204-2209, April, 1991.

[29] Scott R. Ploen, Frank C. Park. A Lie group formulation of the dynamics of cooperation robot systems. *Journal of Robotics and Autonomous Systems*, Vol.21, pp. 279-287, 1997.

[30] J. Loncaric. Normal forms of stiffness and compliance matrices, *IEEE Journal of Robotics and Automation*. 3 (6), pp. 567 – 572, 1987.

[31] Irene Fassi, Giovanni Legnani, Diego Tosi, Alberto Omodei. Calibration of serial manipulators: theory and applications. *In: Industrial robotics: programming, simulation and applications*. Proliteratur Verlag, Mammendorf, Germany, pp.147-170, 2007.

[32] M. A. Meggiolaro, S. Dubowsky. An analytical method to eliminate the redundant parameters in robot calibration. *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*, San Francisco, CA, pp.3609-3615, April, 2000.

[33] I-Ming Chen, Guilin Yang, Chee Tat Tan, Song Huat Yeo. Local POE model for robot kinematic calibration. *Mechanism and Machine Theory*, 36, pp.1215-1239, 2001.

[34] I-Ming Chen, Guilin Yang. Kinematic calibration of modular reconfigurable robots using product-of-exponentials formula. *Journal of Robotic Systems,*pp.807-821*, 1997*.

[35] I-Ming Chen, Guilin Yang, Wee Kiat Lim, Song Huat Yeo. Self-calibration of three-legged modular reconfigurable parallel robots based on measurement residues. *Technical report, School of Mechanical & Production Engineering, Nanyang Technological University,* 1999.

[36] R. B. He, Y. J. Zhao, S. N. Yang, and S. Z. Yang, Kinematic-parameter Identification for Serial-robot Calibration Based on POE Formula, *IEEE Trans. on Robotics,* Vol. 26, No. 3, pp. 411-423, Jun. 2010.

[37] Tian Huang, Derek G. Chetwynd, David J. Whitehouse, Jinsong Wang. A general and novel approach for parameter identification of 6-DOF parallel kinematic machines. *Int. J. of Mechanism an Machine Theory,* 40, pp. 219-239, 2005.

[38] David Daney, Ioannis Z. Emiris. Identification of parallel robot kinematic parameters under uncertainties by using algebraic methods. *Proceedings of the 11th World Congress in Mechanism and Machine Science,* April 1-4, Tianjin, China, 2004

[39] Fan LiangZhi, A.Y. Elatta, Li XiaoPing. Kinematic calibration for a hybrid 5DOF manipulator based on 3- RPS in-actuated parallel manipulator. *Int. J. of Adv. Manuf .Technol,* 25, pp. 730-734, 2005.

[40] A.Y. Elatta, Li Pei Gen, Fan Liang Zhi, Yu Daoyuan and Luo Fei, An overview of robot calibration, *Information Technology Journal 3(1),*2004, pp. 74-78.

[41] Ana C. Majarena, Jorge Santolaria, David Samper and Junan J. Aguilar. An overview of kinematic and calibration models using internal/external sensors or constraints to improve the behaviour of spatial parallel mechanisms. *Journal of sensors, 10*, pp. 10256-10297*,* 2010.

[42] Pierre Renaud, Nicolas Andreff, Jean-Marc Lavest, and Michel Dhome. Simplifying the kinemaic calibration of parallel mechanisms using vision-based metrology. *IEEE trans. Rob.* Vol. 22. pp.12-22, 2006.

[43] Tian Huang, Derek G. Chetwynd, David J. Whitehouse, Jinsong Wang. A general and novel approach for parameter identification of 6-DOF parallel kinematic machines. *Mechanism and Machine Theory*, Vol. 40, pp. 219-239, 2005.

[44] David Daney, Loria, Inria Lorraine, Villers-Les-Nancy. Optimal measurement configurations for Gough platform calibration. *In Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, DC, USA, pp. 147-152, May 2002.

[45] X. Zhong, J. Lewis and F. Nagy. Inverse robot calibration using artificial neural networks. *International Journal of Engineering Applications of Artificial Intelligence, 9(1),* pp. 83–93, 1996.

[46] J. U. Dolinsky, I. D. Jenkinson and G.J. Colquhoun. Application of genetic programming to the calibration of industrial robots. *International Journal of Computers in Industry, 58*, pp. 255–264 *,* 2007.

[47] G. Alici, R. Jagielski, Y. Ahmet and B. Shirinzadeh. Prediction of geometric errors of robot manipulators with Particle Swarm Optimisation method. *Journal of Robotics and Autonomous Systems, 54,* pp. 956-966, 2006.

[48] Y. Liu, B. Liang, C. Li, L. Xue, S. Hu and Y. Jiang. Calibration of a Steward parallel robot using genetic algorithm. *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation,* Harbin, China, pp. 2495–2500, Aug. 5–8, 2007.

[49] H. Wu, H. Handroos. Utilization of differential evolution in inverse kinematics solution of a parallel redundant manipulator. *Fourth international conference on knowledge-based intelligent engineering systems & allied technologies,* Brighton, UK, pp. 812-815, Aug. 30 - Sept. 1, 2000.

[50] J. Vesterstrom, R. Thomsen. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. *International Journal of Evolutionary Computation, 2,* pp.1980−1987, 2004.

[51] J. Tvrdik, Competitive differential evolution and genetic algorithm in GA-DS toolbox, Available from: *http://dsp.vscht.cz/konference_matlab/MATLAB06/prispevky/tvrdik/tvrdik.pdf* *(Retrieved on Apr. 19, 2012).*

[52] S. Paterlini, T. Krink. Differential evolution and particle swarm optimisation in partitional clustering. *Journal of Computational Statistics & Data Analysis*, 50, pp. 1220-1247, 2006.

[53] R. Storn and K. Price. Differential evolution – A practical approach to global optimization. (Berlin: Springer, 2005).

[54] J. Liu, J. Lampinen, A fuzzy adaptive differential evolution algorithm, *International Journal of Soft Computing − A Fusion of Foundations, Methodologies and Applications, 9(6), pp.* 448−462, Jun. 2005.

[55] Faraz M. Mirzaei. A Kalman Filter-Based Algorithm for IMU Camera Calibration: Observability Analysis and Performance Evaluation. *IEEE transactions on robotics,* 24(5), pp.1143-1156, 2008.

[56] Alberto Omodei, Giovanni Legnani, Riccardo Adamini. Three Methodologies for the Calibration of Industrial Manipulators: Experimental Results on a SCARA Robot. *Journal of Robotic Systems*, 17(6), pp. 291-307, 2000.

[57] Julier, S.J.; Uhlmann, J.K. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, Vol. 92, No.3, pp. 401–422, Mar. 2004.

[58] Sy-Miin Chow, Emilio Ferrer, John R. Nesselroade. An Unscented Kalman Filter Approach to the Estimation of nonlinear dynamical systems models. http://www.ocgy.ubc.ca/~yzq/books/paper5_KalmanFilter/04_LorenzKalman.pdf (retrieved on Sept. 13, 2012)

[59] J. P. Merlet. Parallel Robot, 2nd Edition. (Springer. ISBN 978-1-4020-4132-7).

[60] L. W. Tsai. Robot Analysis: The Mechanics of Serial and Parallel Manipulators. (John Wiley & Sons, 1999).

[61] Chunhe Gong, Jingxia Yuan, Jun Ni. Nongeometric error identification and compensation for robotic system by inverse calibration. *International Journal of Machine tools & Manufacture*, 40 (2000), pp. 2119-2137, 2000.

[62] Gao Meng, Li Tiemin, Yin Wensheng. Calibration method and experiment of Stewart Platform using a laser tracker. *In: IEEE International Conference on System, Man and Cybernetics*, Washington DC, Vol. 3, pp. 2797-2802, 2003.

[63] Xiao-Dong Ren, Zu-Ren Feng, Cheng-Ping Su. A new calibration method for parallel kinematics machine tools using orientation constraints. *International Journal of Machine tools & Manufacture, Vol.* 49, pp. 708-721, 2009.

[64] Jian Wang, Oren Masory. On the accuracy of a Stewart platform – Part I: the effect of manufacturing tolerances. *Proceedings of IEEE International Conference on Robotics and Automation.* Florida, USA, Vol. 1, pp. 114-120, 1993.

[65] Hanqi Zhuang, Jiahua yan, Oren Masory. Calibration of Stewart platform and other parallel manipulators by minimizing inverse kinematic residuals. *Journal of Robotic Systems, 15(7)*, pp.395-405, 1998.

[66] Dayong Yu, Hongren Li, Weifang Chen. Kinematic calibration of parallel robots for docking mechanism motion simulation. *International journal of Advanced Robotic System,* Vol. 2. No.4, pp.152-165, 2011.

[67] Yu Liu, Hong Liu, Fenglei Ni, Wenfu Xu, Feng Han. Self-calibration of a Stewart parallel robot with a laserranger. *Intelligent Robotics and Applications, Lecture Notes in Computer Science,* Vol. 6424, pp. 570-581, 2010.

[68] Frank C. Park. Computational aspects of the product-of-exponentials formula for robot kinematics. *IEEE Transactions on Automatic Control, V*ol. 39, No. 3, pp. 643-647, 1994.

[69] Amit J. Patel, Kornel F. Ehmann. Calibration of a hexapod machine tool using a redundant leg. *International Journal of Machine Tools & Manufacture, Vol. 40*, pp. 489-512, 2000.

[70] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics,* 21(6) . pp.1087 – 1092, 1953.

[71] W. K. Hastings, Monte Carlo sampling methods using Markov chains and their applications. *Biometrika,*vol. 57, No.1. pp. 97 – 109, 1970.

[72] W. R. Gilks, S. Richardson , D. J. Spiegalhalter. Markov Chain Monte Carlo in Practice. (Chapman & Hall, London, 1996).

[73] Heikki Haario, Marko Laine, Antonietta Mira. DRAM: efficient adaptive MCMC. *Journal of Stat Comut,* 16, pp. 339-354, 2006.

[74] Olli Malve, Marko Laine, Heikki Haario, Teija Kirkkala, Jouko Sarvala. Bayesian Modelling of Algal Mass Occurrences – Using Adaptive MCMC methods with a lake water quality model. *Int. J. of Environmental Modelling & Software*, 22, pp. 966-977, 2007.

[75] Olli Malve, Marko Laine, Heikki Haario. Estimation of winter respiration rates and prediction of oxygen regime in a lake using Bayesian inference. *Journal of Ecological Modelling*, 182, pp.183-197, 2005.

[76] Kerry Gallagher, Karl Charvin, Soren Nielsen, Malcolm Sambridge, John Stephenson. Markov Chain Monte Carlo (MCMC) sampling methods to determine optimal models, model resolution and model choice for Earth Science problems. *Journal of Marine and Petroleum Geology*, 26, pp.525-535, 2009.

[77] Robert E. Kass and Adrian E. Raftery. Bayes factors. *Journal of the American Statistical Association*, Vol. 90, No. 430, pp.773-795, 1995.

[78] Junhong Liu, Huapeng Wu, Heikki Handroos, and Heikki Haario. Parameter estimation of an electrohydraulic servo system using a Markov Chain Monte Carlo method. *Journal of Dynamic Systems, Measurement, and Control*, Vol. 135, No.1, January 2013.

[79] M. W. Davis. Production of conditional simulations via the LU triangular decomposition of the covariance matrix. *Mathematical Geology*, 19(2), pp.91-98, 1987.

[80] A. Solonen and H. Haario. Model-based process optimization in the presence of parameter uncertainty. *Engineering Optimization,* Vol. 44, No.7, pp.875-894, 2007.

[81] Webpage: http://www1.icsi.berkeley.edu/~storn/code.html. (Retrieved on Aug. 10, 2011).

[82] David Daney, Yves Papegay, Blaise Madeline. Choosing measurement poses for robot calibration with the local convergence method and Tabu search. *The International Journal of Robotics Research*, Vol. 24, No. 6, pp. 501-518, 2005.

[83] Hanqi Zhuang, Kuanchih Wang and Zvi S. Roth. Optimal selection of measurement configurations for robot calibration using simulated annealing. *Proceedings of 1994 IEEE International Conference on Robotics and Automation,* Florida, USA, Vol.1, pp. 393-398, 1994.

[84] Jin-Hwan Born and Chia-Hsiang Meng. Determination of optimal measurement configurations for robot calibration based on observability measure. *The International Journal of Robotics Research,* Vol. 10, No. 1, pp.51-63, 2011.

[85] Chenhua Huang, Cunxi Xie, Tie Zhang. Determination of optimal measurement configurations for robot calibration based on a hybrid optimal method. *Proceedings of the 2008 IEEE International Conference on Information and Automation,* Zhangjiajie, China, pp.789-793, 2008.

[86] M. A. Meggiolaro and S. Dubowsky. Improving the positioning accuracy of powerful manipulators with application in nuclear maintenance. *Proceedings of COBEM 2001, Robotics and Control*, Vol. 15, pp.210-219, 2001.

[87] M. Laine. Adaptive MCMC methods with applications in environmental and geophysical models. (Finnish Meteorological Institute Contributions 69. ISBN 978-951-697-662-7).

[88] Z. Geng and L. S. Haynes. A 3-2-1-kinematic configuration of a Stewart platform and its application to six degree of freedom pose measurements. *Robotics Computer-Integrated Manufacturing,* Vol. 11, No.1, pp.23-34, 1994.

[89] Federico Thomas, Erika Ottaviano, Lluis Ros, and Marco Ceccarelli. Performance analysis of a 3-2-1 pose estimation device. *IEEE Transactions on Robotics,* Vol. 21, No.3, pp. 288-297, 2005.

[90] Federico Thomas, Erika Ottaviano, Lluis Ros, and Marco Ceccarelli. Coordinate-free formulation of a 3-2-1 wire-based tracking device using Cayley-Menger determinants. *Proceedings of the 2003 IEEE International Conference on Robotics & Automation,* Taipei, Taiwan, pp.355-361, 2003.

[91] Robert L. Williams II, James S. Albus and Roger V. Bostelman. 3D Cable-Based Cartesian Metrology System. *Journal of Robotic Systems,*21(5): 237-257, 2004.

[92] Cristina Tavolieri, Erika Ottaviano, Marco Ceccarelli. Pose Determination for a Rigid Body by Means of CatraSys II (Cassino Tracking System). *Proceedings of EUCoMeS, the first European Conference on Mechanism Science Obergurgl*, Austria, February 21-26, 2006.

[93] E. E. Hernandez-Martinez, L. Conghui, G. Carbone, M. Ceccarelli, C. S. Lopez-Cajun. Experimental and Numerical Characterization of CaPaMan 2bis Operation. *Journal of Applied Research and Technology,* Vol. 8, No.1, pp. 101-119, April 2010.

[94] G. Yang, I-M Chen, W-K Lim, S-H Yeo. Design and kinematic analysis of modular reconfigurable parallel robots. *Proceedings of the IEEE International Conference on Robotics and Automation*, Detroit, USA, pp. 2501-2506, 1999.

[95] R. M. Murray, Z. X. Li, and S. S. Sastry, A Mathematical Introduction to Robotic Manipulation. (Boca Ration, Florida: CRC Press, 1994).

APPENDIX A

- **POE representation for Robot kinematics**

To facilitate the error modeling of the studied robot, some related mathematic concepts are summarized in this section. For more details please refer to [36] [95].

a) The Lie Group *SO(3)*, or the Special Orthogonal Group, also referred as the rotation group, has the form of

$$SO(3) = \{\mathbf{R} \in \Re^{3\times 3} : \mathbf{R}\mathbf{R}^T = \mathbf{I}, \det \mathbf{R} = 1\}. \tag{A.1}$$

Every rigid body rotation about a fixed axis can be expressed as an $R \in SO(3)$.

b) The Lie Group *SE(3)*, or the Special Euclidean Group, also known in the robotics literature as the homogeneous transformation matrix, has the form of

$$SE(3) = \left\{ g = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} : \mathbf{R} \in SO(3), \mathbf{p} \in \Re^{3\times 1} \right\}. \tag{A.2}$$

SE(3) represents the group of general rigid body motions including rotation and translation.

c) The Lie algebra of *SO(3)*, denoted by *so(3)*, is a vector space of the skew-symmetric matrices, such that

$$so(3) = \{\hat{\boldsymbol{\omega}} \in \Re^{3\times 3} : \hat{\boldsymbol{\omega}}^T = -\hat{\boldsymbol{\omega}}\}, \quad \hat{\boldsymbol{\omega}} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}, \tag{A.3}$$

where the vector $\boldsymbol{\omega}=(\omega_x, \omega_y, \omega_z)^T \in R^{3\times 1}$,which correspondents to the axis of a rigid body rotation. The rotation can be represented in an exponential form as $\mathbf{R} = e^{\hat{\omega} q}$, where q represents the angle of the rotation.

d) The Lie algebra of *SE(3)*, denoted by *se(3)*, is defined as

$$se(3) = \{\hat{\boldsymbol{\xi}} \in \begin{bmatrix} \hat{\boldsymbol{\omega}} & \mathbf{v} \\ \mathbf{0} & 0 \end{bmatrix} : \hat{\boldsymbol{\omega}} \in so(3), \mathbf{v} \in \Re^{3\times 1}\}, \tag{A.4}$$

where $\hat{\boldsymbol{\xi}}$ admits a six-dimentional vector presentation: $\boldsymbol{\xi}=(\boldsymbol{\omega}, \mathrm{v})^T$, termed as twist. The twist $\boldsymbol{\xi}$ represents the line coordinate of the screw axis of a general rigid body motion. $\boldsymbol{\omega}$ is the unit directional vector of the axis, **v** is the position of the axis with respect to the origin. In the exponential form, $g = e^{\hat{\xi} q} \in SE(3)$,where $q \in R$ is joint variable which represents the angle or displacement of a joint motion. For revolute joint, if $\boldsymbol{p} \in R^{3\times 1}$ is an arbitary point on the axis, then $\mathbf{v}=-\boldsymbol{\omega}\times\mathbf{p}$. For prismatic joint, $\boldsymbol{\omega}=0$, **v** represents the unit directional vector of the axis.

e) Adjoint transformation, is a 6×6 matrix which transforms twists from one coordinate frame to another, written as Ad(g). Thus, given $g \in SE(3)$, Ad(g) can be expressed as

$$Ad(g) = \begin{bmatrix} \mathbf{R} & \mathbf{0}_{3\times 3} \\ \hat{\mathbf{b}}\mathbf{R} & \mathbf{R} \end{bmatrix}, \tag{A.5}$$

where $\hat{\mathbf{b}}$ is the skew-symmetric matrix of vector b.

f) Exponential of *se(3)*, presents an important connection between a Lie Group *SE(3)* and its Lie algebra *se(3)*. Given $\hat{\boldsymbol{\xi}} \in se(3)$, $\boldsymbol{\xi}=(\boldsymbol{\omega}, \mathrm{v})^{\mathrm{T}}$ and $\|\boldsymbol{\omega}\| = \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2}$, then

$$e^{\hat{\xi}q} = \begin{bmatrix} e^{\hat{\boldsymbol{\omega}}q} & (\mathbf{I}_3 - e^{\hat{\boldsymbol{\omega}}q})(\boldsymbol{\omega}\times\mathbf{v}) + \boldsymbol{\omega}\boldsymbol{\omega}^T\mathbf{v}q \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{b} \\ \mathbf{0} & 1 \end{bmatrix}, \tag{A.6}$$

where if $\|\boldsymbol{\omega}\|=1$, then

$$\begin{aligned} \mathbf{R} = e^{\hat{\boldsymbol{\omega}}q} &= \mathbf{I}_3 + \sin(q)\hat{\boldsymbol{\omega}} + (1-\cos(q))\hat{\boldsymbol{\omega}}^2 \\ &= \begin{bmatrix} \omega_x^2 v_q + c_q & \omega_x\omega_y v_q - \omega_z s_q & \omega_x\omega_z v_q + \omega_y s_q \\ \omega_x\omega_y v_q + \omega_z s_q & \omega_y^2 v_q + c_q & \omega_y\omega_z v_q - \omega_x s_q \\ \omega_x\omega_z v_q - \omega_y s_q & \omega_y\omega_z v_q + \omega_x s_q & \omega_z^2 v_q + c_q \end{bmatrix}, \end{aligned} \tag{A.7}$$

here c_q, s_q are abbreviations for *cos*(q) and *sin(q)* respectively, and $v_q=1-c_q$.

If $\|\boldsymbol{\omega}\| \neq 1$,

$$\mathbf{R} = \mathbf{I}_3 + \frac{\sin(\|\boldsymbol{\omega}\|q)}{\|\boldsymbol{\omega}\|}\hat{\boldsymbol{\omega}} + \frac{1-\cos(\|\boldsymbol{\omega}\|q)}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}^2, \tag{A.8}$$

and

$$\mathbf{b} = \left(q\mathbf{I}_3 + \frac{1-\cos(\|\boldsymbol{\omega}\|q)}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}} + \frac{\|\boldsymbol{\omega}\|q - \sin(\|\boldsymbol{\omega}\|q)}{\|\boldsymbol{\omega}\|^3}\hat{\boldsymbol{\omega}}^2 \right)\mathbf{v}. \tag{A.9}$$

If $\|\boldsymbol{\omega}\|=0$, which means the joint is prismatic, then

$$\mathbf{R}=\mathbf{I}_3, \quad \mathbf{b}=q\mathbf{v}. \tag{A.10}$$

g) Forward kinematics using POE formular

Combining the individual joint motions, the forward kinematics for an *n-degree-of -freedom* serial robot is given by

$$g_{st}(\mathbf{q}) = e^{\hat{\xi}_1 q_1} e^{\hat{\xi}_1 q_1} \cdots e^{\hat{\xi}_n q_n} g_{st}(0). \tag{A.11}$$

where $g_{st}(0)$ represents the rigid body transformation between tool frame *T* and base frame *S* when the manipulator is in its reference configuration ($\mathbf{q}$=0). We can define any configuration of the manipulator as the reference configuration. One natural choice is to let the base frame be coincident with the tool frame in reference configuration, then $g_{st}(0)=\mathbf{I}$. The twist coordinates ξ_i for the individual joints of a manipulator depend on the choice of reference configuration (as well as base frame) and so the reference configuration is usually chosen such that the kinematic analysis is as simple as possible.

- **POE representation for Robot error modeling**

According to the error model of He [36], if let the base frame coincident with the tool frame in the reference configuration, and assuming no errors in $g_{st}(0)$ and $\mathbf{q}$, then a POE based error model can be expressed in an explicit form as:

$$\begin{aligned} \left[\delta g g^{-1}\right]^{\vee} &= (\delta e^{\hat{\xi}_1 q_1} \cdot e^{-\hat{\xi}_1 q_1})^{\vee} \\ &+ Ad(e^{\hat{\xi}_1 q_1})(\delta e^{\hat{\xi}_2 q_2} \cdot e^{-\hat{\xi}_2 q_2})^{\vee} \\ &+ \cdots + Ad(\prod_{i=1}^{n-1} e^{\hat{\xi}_i q_i})(\delta e^{\hat{\xi}_n q_n} \cdot e^{-\hat{\xi}_n q_n})^{\vee} \end{aligned}, \tag{A.12}$$

where

$$(\delta e^{\hat{\xi}_i q_i} \cdot e^{-\hat{\xi}_i q_i})^{\vee} = \mathbf{A}_i \delta \boldsymbol{\xi}_i ,$$

and

$$\begin{aligned} \mathbf{A}_i = q_i \mathbf{I} &+ \frac{4 - \theta_i \sin(\theta_i) - 4\cos(\theta_i)}{2\|\boldsymbol{\omega}\|^2} \boldsymbol{\Omega}_i + \frac{4\theta_i - 5\sin(\theta_i) + \theta_i \cos(\theta_i)}{2\|\boldsymbol{\omega}\|^3} \boldsymbol{\Omega}_i^2 \\ &+ \frac{2 - \theta_i \sin(\theta_i) - 2\cos(\theta_i)}{2\|\boldsymbol{\omega}\|^4} \boldsymbol{\Omega}_i^3 + \frac{2\theta_i - 3\sin(\theta_i) + \theta_i \cos(\theta_i)}{2\|\boldsymbol{\omega}\|^5} \boldsymbol{\Omega}_i^4 \end{aligned},$$

and

$$\begin{aligned} &\boldsymbol{\Omega}_i = \begin{bmatrix} \hat{\boldsymbol{\omega}}_i & \mathbf{0}_{3\times 3} \\ \hat{v}_i & \hat{\boldsymbol{\omega}}_i \end{bmatrix}, \quad \theta_i = \|\boldsymbol{\omega}_i\| q_i \\ &\|\boldsymbol{\omega}_i\| = \sqrt{\omega_{xi}^2 + \omega_{yi}^2 + \omega_{zi}^2} \end{aligned}.$$

PART II: PUBLICATIONS

PUBLICATION 1

Wang ,Y.B. & Wu, H.P. & Handroos, H. (2012)

Error Modelling and Differential-Evolution-Based Parameter Identification Method for Redundant Hybrid Robot

The paper has been published in the *International Journal of Modelling and Simulation, vol.32, No. 4*, 2012, p.255-264

International Journal of Modelling and Simulation, Vol. 32, No. 4, 2012

ERROR MODELLING AND DIFFERENTIAL-EVOLUTION-BASED PARAMETER IDENTIFICATION METHOD FOR REDUNDANT HYBRID ROBOT

Yongbo Wang,* Huapeng Wu,* and Heikki Handroos*

Abstract

This paper focuses on the geometrical error modelling and parameter identification of a 10 degree-of-freedom (DOF) redundant serial–parallel hybrid intersector welding/cutting robot (IWR). The proposed hybrid robot consists of a kinematically redundant 4-DOF serial mechanism to enlarge workspace and a 6-DOF Stewart parallel robot to improve the end-effector accuracy. Due to its redundant degrees of freedom and the serial–parallel structure, the traditional error modelling and identification methods which tailored for pure serial robot or pure parallel robot cannot be directly used. In this paper, a hybrid error modelling method for redundant serial–parallel hybrid robot is presented by combining both the traditional forward calibration and inverse calibration method. Furthermore, because of the high nonlinear and multi-modal characteristics of the derived hybrid error model, the traditional iterative linear least-square algorithm cannot be utilized to identify the error parameters. In this paper, an easy-to-use and powerful evolutionary global optimization algorithm named differential evolution (DE) is employed to search for a set of optimum combination of all error parameters in the error model to minimize the discrepancies of measured and predicted leg lengths. Numerical simulation and analysis are conducted by generating random manufacturing and assembly errors within the real error parameter tolerance range. Meanwhile, different measurement poses of the end-effector and the corresponding joint displacements of the serial mechanism are also randomly generated in the workspace to simulate the real physical behaviours. The simulation results show that the DE-based parameter identification method is robust and reliable, and all of the preset errors can be successfully recovered. The simulation also shows that the hybrid calibration method can avoid the external pose measurement of the connecting point between serial and parallel mechanism, and the pose measurement of the end-effector of serial–parallel robot can satisfy the calibration purpose effectively.

* Faculty of Technology, The Laboratory of Intelligent Machines, Lappeenranta University of Technology, Finland; e-mail: {yongbo.wang, Huapeng.wu, heikki.handroos}@lut.fi

Recommended by Dr. J. Shen
(DOI: 10.2316/Journal.205.2012.4.205-5750)

Key Words

Error modelling, differential evolution, calibration, parameter identification, hybrid robot, accuracy

1. Introduction

Practically, kinematic errors in robot manipulators which originated from manufacturing and assembly processes are inevitable thus have to be compensated to a certain value to meet a specified accuracy requirement. However, after the robot is assembled, it would be very difficult but still possible to measure the geometrical errors. Alternatively, the most cost-effective way for improving robot accuracy is to formulate an accurate mathematical error model according to the designed geometric characteristics and then use numerical method to identify the unknown error parameters in the error model and compensate for these errors in the controller. Over recent decades, a number of different modelling methods have been proposed for kinematic modelling of serial robot manipulators. The most popular model is the DH model which was developed by Denavit and Hartenberg (DH) [1], but it suffers from singularity problem when two consecutive joints are parallel or near parallel. To avoid singularity of DH convention, many modelling methods have been proposed. For instance, Hayati and Mirmirani [2] established a modified DH model; Veitschegger and Wu [3] developed a linear and a second-order error modelling methods for serial robot; Stone and Sanderson [4] proposed a S-model which uses six parameters for each link and these parameters are converted to DH parameters; Mooring [5], [6] presented the zero-reference model and it does not rely on the DH formalism and adopts only two coordinate systems: one is reference coordinate system fixed in the work space and another one is end-effector coordinate system attached to the end-effector of the robot. For parallel manipulators, due to its closed-loop kinematic structure characteristics, the vector chain analytical method is commonly adopted for kinematic modelling. In terms of hybrid robot connected by serial and parallel mechanisms, to the best of our knowledge, there are no generic modelling and identification method available. This paper presents a hybrid error modelling method for redundant serial–parallel robot; it is a combination of DH modelling method for serial mechanism and vector chain analytical method for parallel mechanism. If this hybrid calibration method employs DH model or modified DH model to predict a forward solution for serial mechanism, then vector chain analytical method are used to develop an inverse solution for parallel mechanism. The advantage of this method is that the external pose measurement of the connection point between serial and parallel mechanism is avoided. Therefore, if the two hybrid parts do not need to calibrate separately and can be regarded as a whole, then the pose measurement of the end-effector can fulfil the calibration purpose effectively. Once the most suitable calibration model has been selected for the mechanism, the next step is to select a suitable optimization method to find out a set of optimum solutions in the error model to minimize the derived objective functions. Generally, the optimization method in this step can be divided into two categories. One is iterative linearization method which linearizes the derived error model, obtains a corresponding identification Jacobian matrix and then recursively solves the linear system until the average error approaches a stable minimum. The advantage of this method is less computation time to converge, but the identification Jacobian may suffer from numerical problems of ill-conditioning. To overcome this problem, the Levenberg and Marquardt (LM) minimization techniques can be used [7], [8], but for complex models, LM algorithms may converge to local minimum. Another one is nonlinear optimization method which minimizes the sum of square errors between the measured and predicted values based on the Euclidean norm to search a set of optimum error parameters in the predicted error model. This method is commonly used in the high nonlinear and complex systems where the identification Jacobian matrix is not easy but still possible to derive. Based on the error model, some global optimization algorithms such as Markov Chain Monte Carlo methods [9], artificial neural networks [10], genetic programming [11], particle swarm optimization (PSO) [12], genetic algorithms (GA) [13] and differential evolution (DE) [14] have been successfully employed to calibrate the specific serial or parallel robots. The comparison of these global optimization methods for benchmark or real-world applications can be found in some literatures [15]–[17]. The benchmark comparison of DE, GA, PSO, evolutionary algorithms (EAs) in [15], [16] demonstrated that DE algorithms are more reliable and easy-to-use than other optimization algorithms. The comparison in [17] shows that DE is clearly and consistently superior to GAs and PSO in terms of precision as well as robustness of the results for hard clustering problems. In general, DE is a simple but effective EA to solve nonlinear and global optimization problems [18], [19]. The DE-based identification method is a nonlinear optimization method and is purely stochastic; it avoids problems in defining search direction, and whether the initial values are close to the optimum solution or not is insignificant. Therefore, the development of identification matrix is not necessary and the numerical

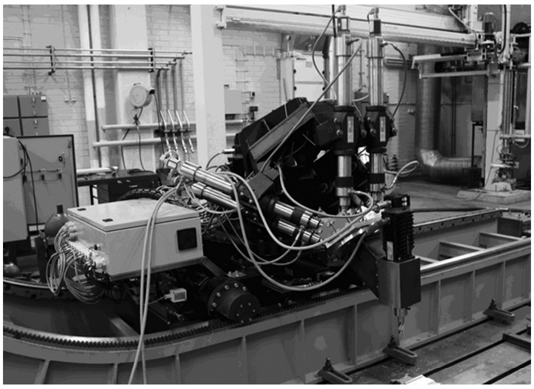


Figure 1. Experimental prototype developed in LUT.

problem of ill-conditioning of identification matrix can be avoided. Due to the outstanding performance of DE and the complicated error model of the proposed hybrid robot, the DE algorithm will be employed in this paper to search a set of optimum solutions globally in the predicted error model to minimize the position error of the end-effector.

The remainder of the article is organized as follows. The kinematic and identification models of the robot are derived in Section 2. Section 3 presents the implementation of DE algorithm. Simulation results are given in Section 4, and conclusions are drawn in Section 5.

2. Error Modelling

The prototype of the hybrid serial–parallel robot, as shown in Fig. 1, is composed of a 4-degree-of-freedom (DOF) serial mechanism (carriage) and a 6-DOF hexapod parallel mechanism (Hexa-WH). The aim is to make a compromise between large workspace of serial manipulators and high stiffness of parallel manipulators. The robot is designed for machining and assembling the vacuum vessel of ITER [20]. To simplify the analysis, the kinematic and identification model of the two parts will be derived separately in Section 2.1 and then be integrated together to get the hybrid model [21], [22].

2.1 Kinematic Model

The first step of the calibration procedure is to develop a suitable mathematic model to specify the relationship between the outputs of the joint displacement transducers and the pose of the end-effector. In the following sections, modelling of the carriage, Hexa-WH and intersector welding/cutting robot (IWR) will be discussed in detail.

2.1.1 Kinematic Model for the Carriage

Based on the routine of Denavit–Hartenberg (DH) coordinate system from Paul [23], the related coordinate systems are established as shown in Fig. 2 and the corresponding kinematic parameters are listed, as given in Table 1.

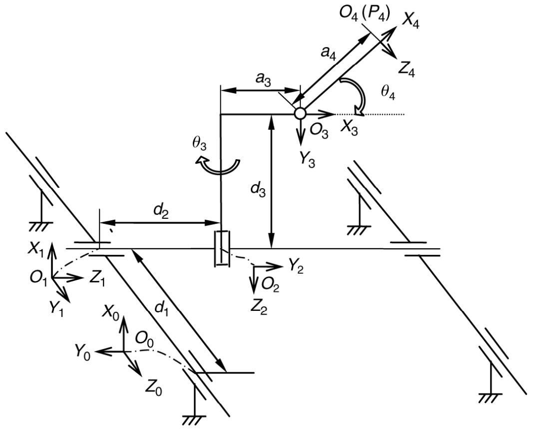


Figure 2. Coordinate system of the carriage.

Table 1
DH Parameters of the Carriage

Link No.	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	d_1 (variable)	0
2	$\pi/2$	0	d_2 (variable)	$\pi/2$
3	$\pi/2$	a_3	d_3	θ_3 (variable)
4	$-\pi/2$	a_4	0	θ_4 (variable)

Substituting the DH link parameters into (1), we obtain the DH homogeneous transformation matrices ${}^0\mathbf{A}_1$, ${}^1\mathbf{A}_2$, ${}^2\mathbf{A}_3$, ${}^3\mathbf{A}_4$ and the nominal forward kinematics of the carriage ${}^0\mathbf{T}_4$. In the following equations, the sine and cosine are abbreviated as s and c.

$$ {}^{i-1}\mathbf{A}_i = \begin{bmatrix} c\theta_i & -c\alpha_i s\theta_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & c\alpha_i c\theta_i & -s\alpha_i c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1} $$

$$ \begin{aligned} {}^0\mathbf{T}_4 &= {}^0\mathbf{A}_1{}^1\mathbf{A}_2{}^2\mathbf{A}_3{}^3\mathbf{A}_4 \\ &= \begin{bmatrix} s\theta_4 & 0 & c\theta_4 & a_1 + d_3 + a_4 s\theta_4 \\ -s\theta_3 c\theta_4 & -c\theta_3 & s\theta_3 s\theta_4 & -d_2 - a_3 s\theta_3 - a_4 s\theta_3 c\theta_4 \\ c\theta_3 c\theta_4 & -s\theta_3 & -c\theta_3 s\theta_4 & d_1 + a_3 c\theta_3 + a_4 c\theta_3 c\theta_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} {}^0\mathbf{R}_4 & {}^0\mathbf{P}_4 \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{2} $$

2.1.2 Kinematic Model for Hexa-WH

A schematic diagram of the hexapod parallel mechanism is shown in Fig. 3. Two Cartesian coordinate systems, frame

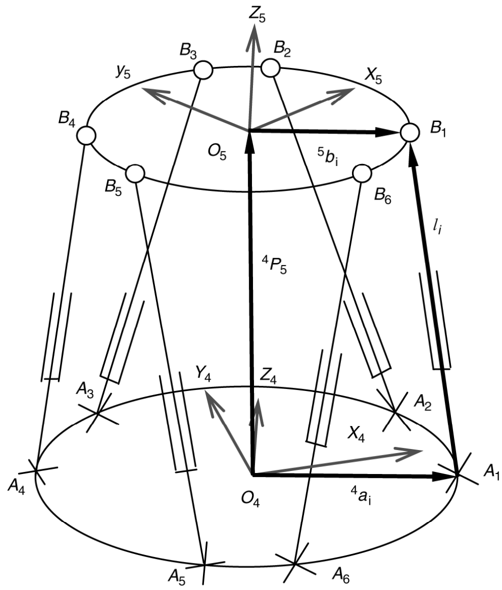


Figure 3. Coordinate system of Hexa-WH.

$O_4(X_4, Y_4, Z_4)$, and frame $O_5(X_5, Y_5, Z_5)$ are attached to the base platform and the end-effector, respectively. Six hydraulic actuated variable legs are connected to the base platform by universal joints and to the task platform by spherical joints.

For the nominal kinematic parameters of Hexa-WH, let $\mathbf{l}_i$ be the unit vector of the direction from $\mathbf{A}_i$ to $\mathbf{B}_i$ and l_i the magnitude of the leg vector $\rightharpoonup \mathbf{A}_i\mathbf{B}_i$. Then the inverse kinematics of the ith leg of the parallel manipulator can be expressed by the following vector-loop equation:

$$ l_i\mathbf{l}_i = {}^4\mathbf{P}_5 + {}^4\mathbf{R}_5{}^5\mathbf{b}_i - {}^4\mathbf{a}_i, \quad i = 1, 2, \ldots, 6 \tag{3} $$

where ${}^4\mathbf{P}_5$ is the position vector of the task frame $\{5\}$ related to the connecting frame $\{4\}$; ${}^4\mathbf{a}_i$ and ${}^5\mathbf{b}_i$ are the position vectors of the universal joint A_i in frame $\{4\}$ and spherical joint B_i in frame $\{5\}$; ${}^4\mathbf{R}_5$ is the Z–Y–X Euler transformation matrix which represents the orientation of frame $\{5\}$ with respect to frame $\{4\}$:

$$ {}^4\mathbf{R}_5 = \begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\lambda - s\alpha c\lambda & c\alpha s\beta c\lambda + s\alpha s\lambda \\ s\alpha c\beta & s\alpha s\beta s\lambda + c\alpha c\lambda & s\alpha s\beta c\lambda + c\alpha s\lambda \\ -s\beta & c\beta s\lambda & c\beta c\lambda \end{bmatrix} \tag{4} $$

2.1.3 Kinematic Model for IWR

Combining the two parts together we can get a schematic diagram for the redundant hybrid manipulator which consists of the carriage and Hexa-WH mechanisms as shown in Fig. 4. The coordinate frame $\{4\}$ of one platform of the

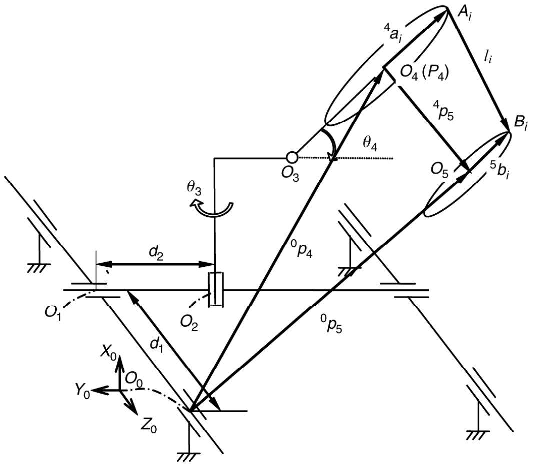


Figure 4. Schematic diagram of IWR.

Hexa-WH is coincident with the end-tip frame of carriage. The fixed reference frame $\{0\}$ is placed at the left rail of carriage. Based on this hybrid structure, we can obtain a vector-loop equation as:

$$\begin{aligned} {}^0\mathbf{P}_5 &= {}^0\mathbf{P}_4 + {}^0\mathbf{R}_4{}^4\mathbf{P}_5 = {}^0\mathbf{P}_4 + {}^0\mathbf{R}_4\left(l_i\mathbf{l}_i + {}^4\mathbf{a}_i - {}^4\mathbf{R}_5{}^5\mathbf{b}_i\right) \\ &= {}^0\mathbf{P}_4 + {}^0\mathbf{R}_4 l_i\mathbf{l}_i + {}^0\mathbf{R}_4{}^4\mathbf{a}_i - {}^0\mathbf{R}_5{}^5\mathbf{b}_i \end{aligned} \tag{5}$$

From (5), the inverse solution of the hybrid robot, *i.e.*, the nominal leg lengths can be derived as:

$$\begin{aligned} l_i\mathbf{l}_i &= \left({}^0\mathbf{R}_4\right)^{-1}\left({}^0\mathbf{P}_5 - {}^0\mathbf{P}_4 - {}^0\mathbf{R}_4{}^4\mathbf{a}_i + {}^0\mathbf{R}_5{}^5\mathbf{b}_i\right), \\ & i = 1, 2, \ldots, 6 \end{aligned} \tag{6}$$

where ${}^0\mathbf{R}_5$ and ${}^0\mathbf{P}_5$ are the orientation matrix and position vector of the end-effector frame $\{5\}$ with respect to the fixed reference frame $\{0\}$.

2.2 Identification Model

2.2.1 Identification Model for the Carriage

According to the approaches proposed by Veitschegger and Wu [3], if small errors occur in the DH parameters θ_i, d_i, a_i and α_i, it will lead to a differential change $d^{i-1}\mathbf{A}_i$ between two successive joint coordinates, and the predicted relationship between the two consecutive joint coordinates can be expressed as:

$$^{i-1}\mathbf{A}_i^p = {}^{i-1}\mathbf{A}_i + d^{i-1}\mathbf{A}_i \tag{7}$$

where ${}^{i-1}\mathbf{A}_i$ is homogeneous transformation matrix which has four nominal DH link parameters that can express the relationship between the joint coordinates i and $i-1$; $d^{i-1}\mathbf{A}_i$ is differential change due to the errors from link parameters and the joint offset errors from actuators. The differential change can be approximated as a linear function by Taylor's series:

$$\begin{aligned} d^{i-1}\mathbf{A}_i &= \frac{\partial^{i-1}\mathbf{A}_i}{\partial\theta_i}\Delta\theta_i + \frac{\partial^{i-1}\mathbf{A}_i}{\partial d_i}\Delta d_i \\ &+ \frac{\partial^{i-1}\mathbf{A}_i}{\partial a_i}\Delta a_i + \frac{\partial^{i-1}\mathbf{A}_i}{\partial\alpha_i}\Delta\alpha_i \end{aligned} \tag{8}$$

where $\Delta\theta_i$, Δd_i, Δa_i and $\Delta\alpha_i$ are small errors in the DH parameters; the partial derivatives are calculated by the nominal geometrical link parameters. From (1), taking the partial derivative with respect to θ_i, d_i, a_i and α_i respectively, we can easily establish the matrices of $\frac{\partial^{i-1}\mathbf{A}_i}{\partial\theta_i}$, $\frac{\partial^{i-1}\mathbf{A}_i}{\partial d_i}$, $\frac{\partial^{i-1}\mathbf{A}_i}{\partial a_i}$, and $\frac{\partial^{i-1}\mathbf{A}_i}{\partial\alpha_i}$. Let $d^{i-1}\mathbf{A}_i = {}^{i-1}\mathbf{A}_i\delta^{i-1}\mathbf{A}_i$ and $\delta^{i-1}\mathbf{A}_i = \mathbf{D}_{\theta_i}\Delta\theta_i + \mathbf{D}_{d_i}\Delta d_i + \mathbf{D}_{a_i}\Delta a_i + \mathbf{D}_{\alpha_i}\Delta\alpha_i$, then by expanding it into matrix form we obtain:

$$\delta^{i-1}\mathbf{A}_i = \begin{bmatrix} 0 & -c\alpha_i\Delta\theta_i & s\alpha_i\Delta\theta_i & \Delta a_i \\ c\alpha_i\Delta\theta_i & 0 & -\Delta\alpha_i & a_i c\alpha_i\Delta\theta_i + s\alpha_i\Delta d_i \\ -s\alpha_i\Delta\theta_i & \Delta\alpha_i & 0 & -a_i s\alpha_i\Delta\theta_i + c\alpha_i\Delta d_i \\ 0 & & 0 & 0 \end{bmatrix} \tag{9}$$

The above expression gives the general differential translation and orientation vectors for joints which are not parallel or near parallel as the function of four DH kinematic errors. In the case of 4-DOF carriage, the predicted forward solution which including kinematic errors can be expressed as:

$${}^0\mathbf{T}_4^p = {}^0\mathbf{T}_4 + d^0\mathbf{T}_4 = \prod_{i=1}^{4}\left({}^{i-1}\mathbf{A}_i + d^{i-1}\mathbf{A}_i\right) = \begin{bmatrix} {}^0\mathbf{R}_4^p & {}^0\mathbf{P}_4^p \\ 0 & 1 \end{bmatrix} \tag{10}$$

Expanding (10) and ignoring the second and higher-order differential errors, the relationship between the differential change in the carriage end-tip point and the change in the link parameters can be expressed as:

$$d^0\mathbf{T}_4 = \delta\mathbf{T}^1 * {}^0\mathbf{T}_4,\ \delta\mathbf{T}^1 = \sum_{i=1}^{4}\left(\left[{}^0\mathbf{A}_i\right] * \delta^{i-1}\mathbf{A}_i * \left[{}^0\mathbf{A}_i\right]^{-1}\right) \tag{11}$$

where $\delta\mathbf{T}^1$ is the first-order error transformation matrix in the fixed reference frame. According to Paul [23], it has the following form:

$$\delta\mathbf{T} = \begin{bmatrix} 0 & -\delta\theta_z & \delta\theta_y & \delta d_x \\ \delta\theta_z & 0 & -\delta\theta_x & \delta d_y \\ -\delta\theta_y & \delta\theta_x & 0 & \delta d_z \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{12}$$

From (10), the predicted orientation matrix ${}^0\mathbf{R}_4^p$ and position vector ${}^0\mathbf{P}_4^p$ of frame $\{4\}$ with respect to frame $\{0\}$ can be formulated, and the unknown constant error parameters $\Delta\theta_i$, Δd_i, Δa_i and $\Delta\alpha_i$ will be taken as identification variables in the final fitness function (15). The DH convention from Paul [23] shows that for a revolute joint whose axis Z_i is a line in space, all four error parameters including the kinematic parameters and joint offset errors have to be calibrated. For a prismatic joint whose Z_i is a free vector, only two parameters describing its orientation ($\Delta\alpha_i$ and $\Delta\theta_i$) are required and the other two must be set to a value of 0. Since the carriage consists of two prismatic joints and two revolute joints, the number of identification parameters for the serial part is 12.

2.2.2 Identification Model for Hexa-WH

For the Hexa-WH, when manufacturing and assembly errors are introduced, the vectors ${}^4\mathbf{a}_i$ and ${}^5\mathbf{b}_i$ will deviate from the nominal values and have constant error parameters $\delta^4\mathbf{a}_i$ and $\delta^5\mathbf{b}_i$. Leg length l_i will have an initial offset δl_i. Then the error model of Hexa-WH can be written as:

$$l_i^p = (l_i + \delta l_i)\mathbf{l}^p = {}^4P_5^m + {}^0R_5^m({}^5\mathbf{b}_i + \delta^5\mathbf{b}_i) - ({}^4\mathbf{a}_i + \delta^4\mathbf{a}_i),\quad i = 1, 2, \ldots, 6 \tag{13}$$

Since in each leg we have 7 error parameters, 3 coordinate error parameters for joint $\mathbf{A}_i$, 3 coordinate error parameters for and $\mathbf{B}_i$ and 1 error parameters for the leg joint offset, the number of identification variables for the Hexa-WH is 42.

2.2.3 Identification Model for IWR

Integrating the above derived error model of serial and parallel part together, we can obtain the final error model of the hybrid robot as:

$$l_i^p = (l_i + \delta l_i)\mathbf{l}^p = ({}^0R_4^p)^{-1}\left[{}^0P_5^m - {}^0P_4^p - {}^0R_4^p({}^4\mathbf{a}_i + \delta^4\mathbf{a}_i) + {}^0R_5^m({}^5\mathbf{b}_i + \delta^5\mathbf{b}_i)\right] \tag{14}$$

where ${}^0\mathbf{P}_5^m$ and ${}^0\mathbf{R}_5^m$ denote the measured position vector and orientation matrix of end-effector and can be obtained *via* accurate measurement instrument; ${}^0\mathbf{P}_4^p$ and ${}^0\mathbf{R}_4^p$ denote the predicted carriage end-tip position vector and the orientation matrix which includes the identification error parameters. Therefore, the error residuals of the measured leg length l_i^m from linear actuator inner sensors and the predicted leg lengths l_i^p in (14) can be adopted to express the objective function of DE algorithm as:

$$\text{Min}\quad f(\Delta\mathbf{k}_c, \delta\mathbf{k}_h) = \sum_{j=1}^{N}\sum_{i=1}^{6}(l_{i,j}^m - l_{i,j}^p)^2 \tag{15}$$

In (15), N is the number of pose measurement points, $l_{i,j}^p$ is the predicted leg length from (14) and $l_{i,j}^m$ is the measured value of the ith leg in the jth measurement configuration; $\Delta\mathbf{k}_c$ and $\delta\mathbf{k}_h$ are the identification parameter vectors from the carriage and Hexa-WH. The total number of these variables is 54, of which 12 are from the carriage while the remaining 42 variables are from Hexa-WH.

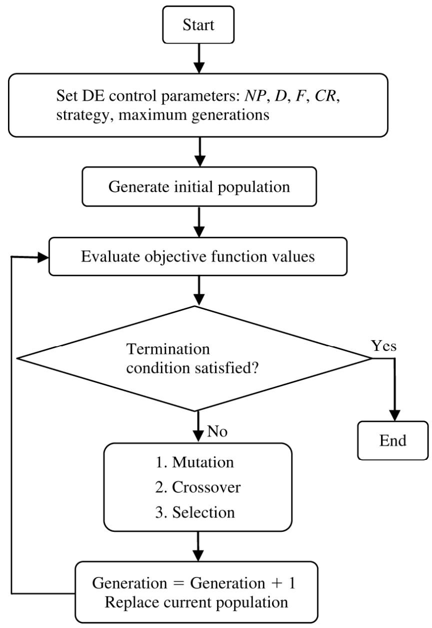


Figure 5. Flowchart of DE algorithm.

3. Application of Differential Evolution for Parameter Identification of Hybrid Robot

DE algorithm is a promising candidate for minimizing real-valued, multi-modal, and nonlinear objective functions [18]. It belongs to the class of EAs and utilizes mutation, crossover and selection operations, as shown in the flowchart in Fig. 5. The number of the identification variables in the objective function is equal to 54. The variables can be represented in DE as an individual vector $\mathbf{x} = (x_1, x_2, \ldots, x_D)$, where D is the individual index. For each generation G, the population can be represented as a matrix $\mathbf{X}_{i,G} \in \Re^{D\times Np}$, where $i = 1, 2, \ldots, Np$ is the population index. The detailed algorithm steps of DE for parameter identification of hybrid robot are discussed below.

3.1 Initialization

To start a DE optimization process, an initial population for a starting point must be created. The natural way to generate the initial population is to assign a random value for each parameter within its feasible boundaries:

$$x_{j,i,G=0} = x_{j,i}^L + rand_j(0,1)\cdot(x_{j,i}^U - x_{j,i}^L) \tag{16}$$

where $j = 1, 2, \ldots, D$ is the individual parameter index, $i = 1, 2, \ldots, NP$ is the population index and $x_{j,i}^L$ and $x_{j,i}^U$ are the lower and upper boundaries of the jth parameter, respectively. After initialization, the population evolves with the operations of mutation, crossover, and selection.

Table 2
Twelve Nominal and Identified Parameters of the Carriage (15 Measurement Poses from 76–90)

No.	Symbols	Nominal Values	Preset Errors	Identified Errors
1	$\alpha_1, \delta\alpha_1$	$\pi/2$	$0.0782°$	0.0781999999993
2	$\alpha_2, \delta\alpha_2$	$\pi/2$	$0.0571°$	0.0570999999997
3	$\alpha_3, \delta\alpha_3$	$\pi/2$	$-0.048°$	-0.048000000001
4	$\alpha_4, \delta\alpha_4$	$-\pi/2$	$0.0417°$	0.0416999991195
5	$a_3, \delta a_3$	252	-0.2164	-0.2164000000026
6	$a_4, \delta a_4$	354	-0.4451	-0.445099998348
7	$d_3, \delta d_3$	422	0.1681	0.1681000000063
8	$d_4, \delta d_4$	0	-0.3857	-0.385700004978
9	$\theta_1, \delta\theta_1$	0	$0.0213°$	0.0213000000003
10	$\theta_2, \delta\theta_2$	$\pi/2$	$0.0794°$	0.0794000000014
11	$\theta_3, \delta\theta_3$	0	$0.0464°$	0.0463999999999
12	$\theta_4, \delta\theta_4$	0	$0.0345°$	0.0345000005016

3.2 Mutation

The main objective of mutation operation is to keep a population robust and search new territory. In the step of DE mutation operation, the new parameter vectors are generated by adding a weighted difference vector between two different population members to the third member. For each vector $\mathbf{x}_{i,G}$, a mutant vector $\mathbf{m}_{i,G+1}$ is generated according to the formula:

$$\mathbf{m}_{i,G+1} = \mathbf{x}_{r1,G} + F \cdot (\mathbf{x}_{r2,G} - \mathbf{x}_{r3,G}) \tag{17}$$

The randomly selected integers have to satisfied the requirement of $r1, r2, r3 \in \{1, 2, \ldots, NP\}$ and $r1 \neq r2 \neq r3 \neq i$. The mutation scale factor $F > 0$.

3.3 Crossover

The aim of crossover operation is to increase the diversity of the generated vectors. The trial vector is generated as follows:

$$\mathbf{u}_{i,G+1} = (u_{1,i,G+1}, u_{2,i,G+1}, \ldots, u_{D,i,G+1})$$
$$u_{j,i,G+1} = \begin{cases} m_{j,i,G+1}, & \text{if } (rand_j[0,1) < CR \vee j = j_r) \\ x_{j,i,G}, & \text{otherwise} \end{cases} \tag{18}$$

where $G = 1, 2, \ldots, G_{\max}$ is generation index. j_r is chosen randomly from the set $\{1, 2, \ldots, D\}$, the use of j_r is to ensure that vector $u_{j,i,G+1}$ gets at least one parameter from $\mathbf{m}_{i,G+1}$. CR is a crossover rate; it is a parameter defined by users in the range of [0,1].

3.4 Selection

In the selection operation of DE, the trial vector $\mathbf{u}_{i,G+1}$ is compared to the target vector $\mathbf{x}_{i,G}$ by evaluating the objective function to decide whether the trial vector can become a member of the next generation or not. The vector, which has a smaller objective function value, is allowed to evolve to the next generation, *i.e.*:

$$\mathbf{x}_{i,G+1} = \begin{cases} \mathbf{u}_{i,G+1}, & \text{if } f(\mathbf{u}_{i,G+1}) \leq f(\mathbf{x}_{i,G}) \\ \mathbf{x}_{i,G}, & \text{otherwise} \end{cases} \tag{19}$$

By using this selection procedure, it can be guaranteed that all individuals of the next generation are as good as or better than the individuals of the current population.

4. Simulation Results

In this section, some numerical simulations for identifying the kinematic error parameters of a novel redundant hybrid robot are conducted to verify the validity and effectiveness of the DE-based method. The detailed nominal and identified geometrical parameters of the carriage and Hexa-WH are listed in Tables 2 and 3. The simulation procedures are as follows:

1. Randomly generate 100 end-effector measurement poses $({}^0\mathbf{P}_5^m, {}^0\mathbf{R}_5^m)$ within the robot workspace. And also randomly generate 100 joint displacements of the carriage actuators within the real motion range, which can be the representative of the nominal joint displacements of the carriage actuators in real case. In practice, the end-effector poses are obtained by the external measuring devices and the joint displacements are collected from the actuator sensor readings.

Table 3
Forty-Two Nominal and Identified Parameters of the Hexa-WH (15 Measurement Poses from 76–90)

No.	Symbols	Nominal Values	Preset Errors	Identified Errors
1	$a_{1x}, \delta a_{1x}$	231.6663	−0.0654	−0.06540000167417
2	$a_{1y}, \delta a_{1y}$	−231.9022	0.0687	0.068699995022313
3	$a_{1z}, \delta a_{1z}$	0	0.0928	0.092799991314147
4	$a_{2x}, \delta a_{2x}$	316.663	0.0448	0.044799998331845
5	$a_{2y}, \delta a_{2y}$	−84.6778	−0.0942	−0.09420000497952
6	$a_{2z}, \delta a_{2z}$	0	−0.0731	−0.07310000716682
7	$a_{3x}, \delta a_{3x}$	85	0.0229	0.022899998344534
8	$a_{3y}, \delta a_{3y}$	316.58	0.0133	0.013299995030753
9	$a_{3z}, \delta a_{3z}$	0	−0.0136	−0.01359999896447
10	$a_{4x}, \delta a_{4x}$	−85	−0.0752	−0.07520000166834
11	$a_{4y}, \delta a_{4y}$	316.58	−0.0976	−0.09760000496367
12	$a_{4z}, \delta a_{4z}$	0	0.0167	0.016700002517118
13	$a_{5x}, \delta a_{5x}$	−316.663	0.0576	0.057599998340029
14	$a_{5y}, \delta a_{5y}$	−84.6778	−0.0486	−0.04860000497178
15	$a_{5z}, \delta a_{5z}$	0	0.0329	0.032899998377918
16	$a_{6x}, \delta a_{6x}$	−231.6663	−0.0117	−0.01170000167045
17	$a_{6y}, \delta a_{6y}$	−231.9022	0.0676	0.06759999502538
18	$a_{6z}, \delta a_{6z}$	0	0.0273	0.02729999537018
19	$b_{1x}, \delta b_{1x}$	32.5	0.0581	0.058100000005788
20	$b_{1y}, \delta b_{1y}$	−125.93	−0.0648	−0.064800000023
21	$b_{1z}, \delta b_{1z}$	0	0.0717	0.071699999997256
22	$b_{2x}, \delta b_{2x}$	125.309	0.0847	0.084699999999777
23	$b_{2y}, \delta b_{2y}$	34.819	−0.0478	−0.04779999999904
24	$b_{2z}, \delta b_{2z}$	0	0.0324	0.03239999999749
25	$b_{3x}, \delta b_{3x}$	92.809	−0.0139	−0.0139000000058
26	$b_{3y}, \delta b_{3y}$	91.111	−0.0266	−0.02659999996617
27	$b_{3z}, \delta b_{3z}$	0	−0.0281	−0.02809999998549
28	$b_{4x}, \delta b_{4x}$	−92.809	−0.0594	−0.05939999998726
29	$b_{4y}, \delta b_{4y}$	91.111	0.0375	0.037499999987263
30	$b_{4z}, \delta b_{4z}$	0	0.0088	0.0088000000136
31	$b_{5x}, \delta b_{5x}$	−125.309	0.0228	0.022799999997413
32	$b_{5y}, \delta b_{5y}$	34.819	−0.0566	−0.05659999999247
33	$b_{5z}, \delta b_{5z}$	0	−0.0368	−0.03680000000364
34	$b_{6x}, \delta b_{6x}$	−32.5	−0.0638	−0.0638000000001
35	$b_{6y}, \delta b_{6y}$	−125.93	−0.0087	−0.00870000001196
36	$b_{6z}, \delta b_{6z}$	231.6663	−0.0736	−0.07359999999731
37	$l_1, \delta l_1$	350	−0.3794	−0.37939999998198
38	$l_2, \delta l_2$	350	−0.0895	−0.08950000000639
39	$l_3, \delta l_3$	350	0.1650	0.164999999968428
40	$l_4, \delta l_4$	350	−0.3048	−0.30479999997541
41	$l_5, \delta l_5$	350	0.3233	0.323299999988304
42	$l_6, \delta l_6$	350	0.0774	0.077400000010887

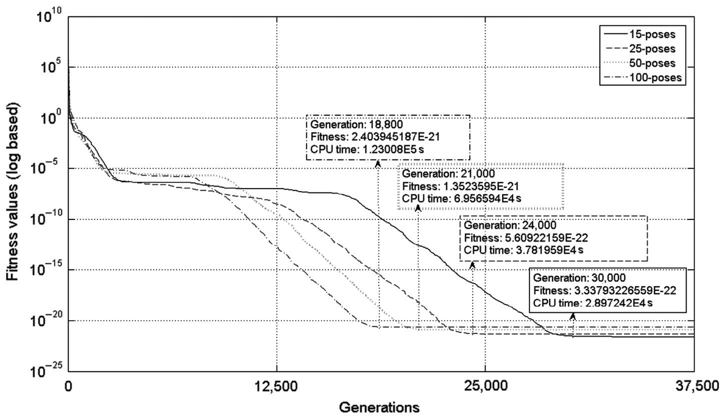


Figure 6. Fitness values versus simulation generations with different number of measurement poses.

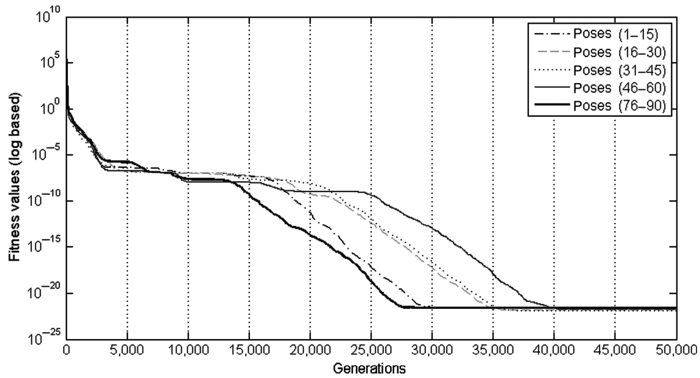


Figure 7. Fitness values versus simulation generations with same number of measurement poses.

2. Assuming some preset errors for the DH error parameters, the leg joint offset error parameters and the coordinate error parameters of joint $\mathbf{A}_i$ and $\mathbf{B}_i$, these preset errors can represent corresponding real physical manufacturing and assembly errors within the designed tolerance range (see Tables 2 and 3).
3. Based on the above nominal kinematic values, the generated poses, the carriage joint displacements and the preset errors, we can calculate the actual leg lengths $l^m_{i,j}$ according to (14). In reality, the leg lengths can be obtained from the linear actuator sensor readings.
4. Take the 54 kinematic error parameters as the identification variables in the fitness function (15) to calculate the predicted leg lengths $l^p_{i,j}$. Then the task of the simulation is to employ DE algorithm to search for an optimal combination of error parameters to minimize the value of the fitness function under some program terminal conditions.

To validate the identification algorithm, we assume that the measurement device is perfect and the measurement noises are omitted. The DE control parameters can be selected according to the scheme of DE/rand-to-best/1 [18]. In the simulation, the open source Matlab® code of DE from [24] is employed, the DE control parameter are set to be $F = \lambda = 0.75$, $CR = 0.95$, $D = 54$, $Np = 600$ and the error bound range is $[-0.5, 0.5]$, the termination conditions of maximum generation $G_{\max} = 40,000$ and the minimum objective function threshold are set to be 10^{-23}. Since in every measurement pose we have only 6 equations for calculating the leg lengths, to identify 54 variables we need at least 9 measurement poses. The simulations were implemented on a computer with an Intel® Core 2 Duo processor E8500, 3.16 GHz and 3.25 GB of RAM. Figure 6 shows the results of calibration simulation with different number of measurement poses. From the results we can see that about 15 measurement poses are adequate for the calibration results to stabilize. With the increase of measurement poses, the simulation time is increased but the number of simulation generations is decreased when the calibration results are stabilized. The increase of measurement poses cannot significantly improve the stabilized objective function values.

To simulate the influence of the same number of measurement poses in different pose configurations, we select 5 sections from 100 measurement poses and let each section has 15 measurement poses. The simulation results are demonstrated in Fig. 7. It shows that all of the selected runs can converge to almost the same stabilized values and the simulation time can be reduced by suitably arrange the measurement configurations.

Tables 2 and 3 also present the preset geometrical parameter errors and the identified parameter errors after one of the termination conditions has been satisfied when use measurement poses 76–90. From the identification results it can be seen that all of the preset variable values have been successfully recovered and the precision of the final optimum objective function value approaches to the scale of 10^{-22}.

5. Conclusions and Future Work

An error modelling and parameter identification method for redundant serial–parallel hybrid robots is presented. The proposed hybrid modelling method takes into account the geometrical errors which are originated from machining and assembly processes. The method is a combination of the traditional forward calibration method for serial mechanism and inverse calibration method for parallel mechanism. DH model or modified DH model is employed to predict a forward solution for the serial mechanism and vector chain analytical method is used to develop an inverse solution for the parallel and the whole mechanism. The advantage of this method is that the forward solution of the serial mechanism can be used as a prediction value to fit into the final error model. Therefore, the full pose measurement of the end-effector of hybrid robot can meet the calibration requirement effectively and the pose measurement of the end-tip of serial mechanism is not necessary. The identification of the unknown error parameters involves the using of a powerful evolutionary global optimization method, DE algorithm. Computer simulation of the serial–parallel IWR robot demonstrated that all of the 54 geometrical error parameters of the hybrid robot can be successfully identified. The simulation results show that the DE-based parameter identification algorithm has a very strong stochastic searching ability; it is very robust and effective and can be easily employed to identify multi-dimensional error parameters for high nonlinear kinematic models. The simulation is also helpful to find the most suitable DE algorithm control parameters and termination conditions before carrying out experimental test. Our future work will focus on the experimental validation of our method for the current robotic system and extend the proposed method to other serial–parallel robot to verify its practicability.

Acknowledgements

This work, supported by the European communities under the contract of association between EURATOM and Finnish Tekes, was carried out within the framework of the European Fusion development Agreement. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

[1] J. Denavit and R.S. Hartenberg, Kinematic modelling for robot calibration, *Transactions of ASME Journal of Applied Mechanics*, *22*, 1955, 215–221.

[2] S. Hayati and M. Mirmirani, Improving the absolute positioning accuracy of robot manipulators, *Journal of Robotic Systems*, *2*(4), 1985, 397–413.

[3] W.K. Veitschegger and C.-H. Wu, Robot accuracy analysis based on kinematics, *IEEE Journal of Robotics and Automation, RA-2*(3), 1986, 171–179.

[4] H.W. Stone and A.C. Sanderson, A prototype arm signature identification system, *Proc. IEEE Int. Conf. Robotics and Automation*, March 1987, 175–182.

[5] B.W. Mooring and G.R. Tang, An improved method for identifying the kinematic parameters in a six axis robot, *Proc. 1983 ASME Computers in Engineering Conf.*, Las Vegas, Nevada, 1983, 79–84.

[6] B.W. Mooring, Z.S. Roth, and M.R. Driels, *Fundamentals of manipulator calibration* (New York: John Wiley and Sons, 1991).

[7] A.Y. Elatta, L.P. Gen, F.L. Zhi, Y. Daoyuan, and L. Fei, An overview of robot calibration, *Information Technology Journal*, *3*(1), 2004, 74–78.

[8] A.C. Majarena, J. Santolaria, D. Samper, and J.J. Aguilar, An overview of kinematic and calibration models using internal/external sensors or constraints to improve the behaviour of spatial parallel mechanisms, *Journal of Sensors*, *10*, 2010, 10256–10297.

[9] Y.-B.Wang, H.-P. Wu, and H. Handroos, Markov chain Monte Carlo (MCMC) methods for parameter estimation of a novel hybrid redundant robot, *International Journal of Fusion Engineering and Design*, *86*, 2011, 1863–1867.

[10] X. Zhong, J. Lewis, and F. Nagy, Inverse robot calibration using artificial neural networks, *International Journal of Engineering Applications of Artificial Intelligence*, *9*(1), 1996, 83–93.

[11] J.U. Dolinsky, I.D. Jenkinson, and G.J. Colquhoun, Application of genetic programming to the calibration of industrial robots, *International Journal of Computers in Industry*, *58*, 2007, 255–264.

[12] G. Alıcı, R. Jagielski, Y. Ahmet, and B. Shirinzadeh, Prediction of geometric errors of robot manipulators with particle swarm optimisation method, *Journal of Robotics and Autonomous Systems*, *54*, 2006, 956–966.

[13] Y. Liu, B. Liang, C. Li, L. Xue, S. Hu, and Y. Jiang, Calibration of a Steward parallel robot using genetic algorithm, *Proc. 2007 IEEE Int. Conf. on Mechatronics and Automation*, Harbin, China, 5–8 August 2007, 2495–2500.

[14] H. Wu and H. Handroos, Utilization of differential evolution in inverse kinematics solution of a parallel redundant manipulator, *Fourth Int. Conf. on Knowledge-Based Intelligent Engineering Systems & Allied Technologies*, Brighton, UK, 30 Aug–1 Sept, 2000, 812–815.

[15] J. Vesterstrom and R. Thomsen, A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems, *International Journal of Evolutionary Computation*, *2*, 2004, 1980–1987.

[16] J. Tvrdik, Competitive differential evolution and genetic algorithm in GA-DS toolbox, http://dsp.vscht.cz/konference_matlab/MATLAB06/prispevky/tvrdik/tvrdik.pdf (retrieved Apr. 19, 2012).

[17] S. Paterlini and T. Krink, Differential evolution and particle swarm optimisation in partitional clustering, *Journal of Computational Statistics & Data Analysis*, *50*, 2006, 1220–1247.

[18] R. Storn and K. Price, *Differential evolution – A practical approach to global optimization.* (Berlin: Springer, 2005).

[19] J. Liu and J. Lampinen, A fuzzy adaptive differential evolution algorithm, *International Journal of Soft Computing – A Fusion*

of Foundations, Methodologies and Applications, *9*(6), 2005, 448–462.
[20] H. Wu, H. Handroos, P. Pessi, J. Kilkki, and L. Jones, Development and control towards a parallel water hydraulic weld/cut for machining processes in ITER vacuum vessel, *International Journal of Fusion Engineering and Design*, *75–79*, 2005, 625–631.
[21] Y.-B.Wang, H.-P. Wu, and H. Handroos, On the error modeling of a novel mobile hybrid parallel robot, *Proceedings of the 2008 IEEE Conference on Robotics, Automation and Mechatronics*, Chengdu, China, Sept. 21–24, 2008, 1217–1222.
[22] Y.-B. Wang, P. Pessi, H.-P. Wu, and H. Handroos, Accuracy analysis of hybrid parallel robot for the assembling of ITER, *International Journal of Fusion Engineering and Design*, *84*, 2009, 1964–1968.
[23] R.P. Paul, Robot *manipulators: Mathematics, programming, and control* (Cambridge, MA: MIT Press, 1981).
[24] http://www.icsi.berkeley.edu/∼storn/code.html (retrieved Aug. 10, 2011).

Biographies

Yongbo Wang received his M.Sc. degree in mechanical engineering from Lappeenranta University of Technology (LUT), Finland, in 2009. He is currently working towards his Ph.D. in the Laboratory of Intelligent Machines at LUT. His main research interests are robot calibration and parameter identification of hybrid redundant serial–parallel robot.

Huapeng Wu earned his B.Sc. and M.Sc. degrees in mechanical engineering from Huazhong University of Science and Technology (HUST), China, in 1986 and 1993, respectively. He received his D.Sc. (Tech.) degree from Lappeenranta University of Technology (LUT), Finland, with the topic "Design and Control of a Parallel Robot" in 2001. His interests range from production machinery design to parallel robotics. He currently holds an associate professorship position in the Laboratory of Intelligent Machines at LUT.

Heikki Handroos earned his M.Sc. and D.Sc. (Tech.) degrees from Tampere University of Technology, Finland, in 1985 and 1991, respectively. He has carried out research on mechatronics for the past 26 years ranging from the modelling of hydraulic systems to the control and development of large-scale serial and parallel manipulators. He has published about 180 publications and led several academic, industrial and EU-funded projects on mechatronics. He has been a member of the ASME Dynamic Systems and Control Division since 1990. He is a full professor and the Head of the Laboratory of Intelligent Machines, LUT.

PUBLICATION 2

Wang ,Y.B. & Wu, H.P. & Handroos, H. (2011)

Markov Chain Monte Carlo (MCMC) methods for parameter estimation of a novel hybrid redundant robot

The paper has been published in the *Journal of Fusion Engineering and Design, 2011, vol. 86,* p. 1863-1867

Fusion Engineering and Design 86 (2011) 1863–1867

Contents lists available at ScienceDirect

Fusion Engineering and Design

journal homepage: www.elsevier.com/locate/fusengdes

Markov Chain Monte Carlo (MCMC) methods for parameter estimation of a novel hybrid redundant robot

Yongbo Wang*, Huapeng Wu, Heikki Handroos

Laboratory of Intelligent Machine, Lappeenranta University of Technology, FIN-53851 Lappeenranta, Finland

ARTICLE INFO

Article history:
Available online 17 February 2011

Keywords:
ITER
Parameter estimation
Markov Chain Monte Carlo
Hybrid robot

ABSTRACT

This paper presents a statistical method for the calibration of a redundantly actuated hybrid serial-parallel robot IWR (Intersector Welding Robot). The robot under study will be used to carry out welding, machining, and remote handing for the assembly of vacuum vessel of International Thermonuclear Experimental Reactor (ITER). The robot has ten degrees of freedom (DOF), among which six DOF are contributed by the parallel mechanism and the rest are from the serial mechanism. In this paper, a kinematic error model which involves 54 unknown geometrical error parameters is developed for the proposed robot. Based on this error model, the mean values of the unknown parameters are statistically analyzed and estimated by means of Markov Chain Monte Carlo (MCMC) approach. The computer simulation is conducted by introducing random geometric errors and measurement poses which represent the corresponding real physical behaviors. The simulation results of the marginal posterior distributions of the estimated model parameters indicate that our method is reliable and robust.

1. Introduction

Robot calibration is used to enhance accuracy of a given manipulator through software modification rather than changing the mechanical structure or imposing tighter tolerances in manufacturing process. In general, a standard calibration procedure consists of 4 steps: modeling, measurement, identification and compensation. The goal of the identification step is to determine the set of error parameters for a real robot to compensate the nominal geometric model and match the measured data [1]. The topic of parameter identification involves numerical methods, which has been studied in depth for a number of years by many researchers. Examination of the literature on robot calibration indicates that a variety of numerical methods have been employed to identify geometric and non-geometric parameters of pure-serial [2] or pure-parallel [3] manipulators. Very few of publications are focused on the calibration of hybrid robot [4].

In this paper, a general identification model for the redundant hybrid robot is developed to represent geometric errors from manufacturing and assembly processes. Furthermore, a novel identification method for the robot calibration is proposed based on the use of Markov Chain Monte Carlo (MCMC) algorithms to statistically estimate the error parameters of the studied robot. Different from traditional parameter identification methods, which produce only one best combination of optimal solutions for the unknown error parameters, MCMC algorithms [5], on the other hand, has the ability to find as many as possible combinations of optimal solutions whose empirical distribution can statistically fit the data equally well within a certain required accuracy range.

The paper is organized as follows: In Section 2 we describe the kinematics of the studied robot. The kinematic and identification model will be derived in this section. Section 3 gives the basic principles of MCMC and its application to the parameter estimations. Simulation results are given in Section 4, and conclusions are drawn in Section 5.

2. Error modeling

Fig. 1 shows a prototype of the hybrid serial-parallel robot under study, which is developed in Lappeenranta University of Technology and can be used for machining and assembling of vacuum vessel of ITER. The robot is composed of a 4 degrees of freedom (DOF) multi-link serial mechanism (named as Carriage) serially connected to a standard 6-DOF Stewart parallel mechanism (named as Hexa-WH), which aims to arrive at a compromise between a high stiffness of parallel manipulators and a large workspace of serial manipulators. In what follows, we first derive a nominal kinematic model for the proposed robot. Thereafter, based on the nominal model, a related identification model including unknown parameters is developed.

* Corresponding author. Tel.: +358 5 6212462; fax: +358 5 6212499.
E-mail address: yongbo.wang@hotmail.com (Y. Wang).

0920-3796/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.fusengdes.2011.01.062

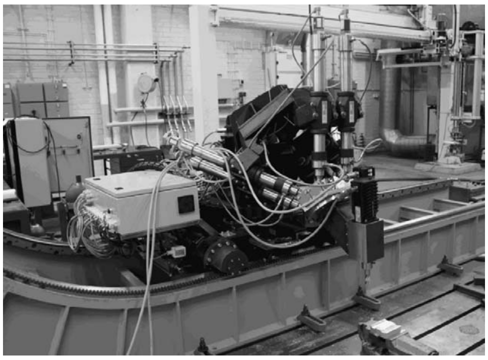


Fig. 1. Prototype of the hybrid serial-parallel robot.

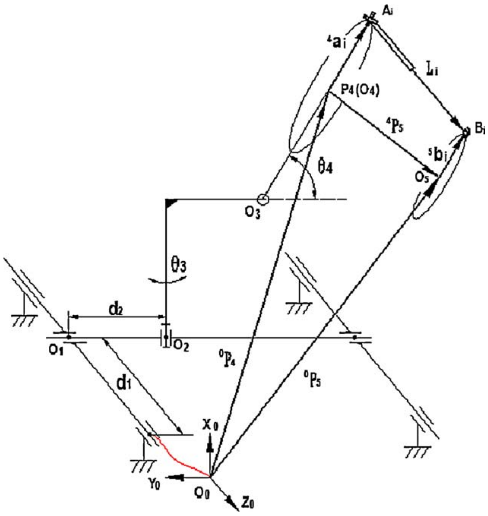


Fig. 2. Schematic diagram of the hybrid robot.

2.1. Kinematic model

The schematic diagram of the redundant hybrid manipulator is shown in Fig. 2. The connection platform frame {4} of the Hexa-WH is coincident with the end-effector of the Carriage. The global reference frame {0} is located at the left rail of the Carriage.

Based on this hybrid structure, a vector-loop equation is derived as:

$$\begin{aligned} {}^0\mathbf{P}_5 &= {}^0\mathbf{P}_4 + {}^0\mathbf{R}_4{}^4\mathbf{P}_5 = {}^0\mathbf{P}_4 + {}^0\mathbf{R}_4(l_i\mathbf{l}_i + {}^4\mathbf{a}_i - {}^4\mathbf{R}_5{}^5\mathbf{b}_i) \\ &= {}^0\mathbf{P}_4 + {}^0\mathbf{R}_4 l_i\mathbf{l}_i + {}^0\mathbf{R}_4{}^4\mathbf{a}_i - {}^0\mathbf{R}_5{}^5\mathbf{b}_i \end{aligned} \quad (1)$$

From Eq. (1), the nominal leg length, i.e., the inverse solution of the robot can be expressed as:

$$l_i\mathbf{l}_i = ({}^0\mathbf{R}_4)^{-1}({}^0\mathbf{P}_5 - {}^0\mathbf{P}_4 - {}^0\mathbf{R}_4{}^4\mathbf{a}_i + {}^0\mathbf{R}_5{}^5\mathbf{b}_i) \quad (2)$$

where ${}^0\mathbf{P}_5$ and ${}^0\mathbf{R}_5$ are the nominal position vector and rotation matrix of the end-effector frame {5} with respect to the fixed base frame {0}. ${}^0\mathbf{R}_4$ and ${}^0\mathbf{P}_4$ are the nominal rotation matrix and position matrix of frame {4} with respect to frame {0}. It can be obtained from the forward kinematics of the Carriage by using the commonly used DH modeling method proposed by Paul [6]. Based on this method, the corresponding nominal forward kinematics of Carriage ${}^0\mathbf{T}_4$ is written as:

$${}^0\mathbf{T}_4 = {}^0\mathbf{A}_1{}^1\mathbf{A}_2{}^2\mathbf{A}_3{}^3\mathbf{A}_4 = \begin{bmatrix} {}^0\mathbf{R}_4 & {}^0\mathbf{P}_4 \\ 0 & 1 \end{bmatrix} \quad (3)$$

${}^4\mathbf{P}_5$ in Eq. (1) is the position vector of the end-effector frame {5} with respect to the connection platform frame {4}. It can be calculated from the nominal inverse kinematics of Hexa-WH. Let $\mathbf{l}_i$ be the unit vector in the direction of $\mathbf{A}_i\mathbf{B}_i$, and l_i the magnitude of the leg vector $\mathbf{A}_i\mathbf{B}_i$. The following vector-loop equation represents the inverse kinematics of the ith limb of the parallel manipulator:

$$l_i\mathbf{l}_i = {}^4\mathbf{P}_5 + {}^4\mathbf{R}_5{}^5\mathbf{b}_i - {}^4\mathbf{a}_i, \quad i = 1, 2, \ldots, 6 \quad (4)$$

where ${}^4\mathbf{a}_i$ and ${}^5\mathbf{b}_i$ denote the position vectors of universal joints $\mathbf{A}_i$ and spherical joints $\mathbf{B}_i$ in frame {4} and frame {5} respectively, and ${}^4\mathbf{R}_5$ is the Z–Y–X Euler transformation matrix representing the orientation of Frame {5} related to Frame {4}.

2.2. Identification model

Considering small geometrical errors happen to robot kinematic DH parameters θ_i, d_i, a_i and α_i, we can get the error model of the Carriage as:

$${}^0\mathbf{T}_4^r = {}^0\mathbf{T}_4 + d^0\mathbf{T}_4 = \prod_{i=1}^{4}({}^{i-1}\mathbf{A}_i + d^{i-1}\mathbf{A}_i) = \begin{bmatrix} {}^0\mathbf{R}_4^r & {}^0\mathbf{P}_4^r \\ 0 & 1 \end{bmatrix} \quad (5)$$

Expanding Eq. (5) and ignoring second and higher order differential errors, it gives:

$$\begin{aligned} &d^0\mathbf{T}_4 = \delta\mathbf{T}^1 * {}^0\mathbf{T}_4, \quad \delta\mathbf{T}^1 = \sum_{i=1}^{4}([{}^0\mathbf{A}_i] * \delta^{i-1}\mathbf{A}_i * [{}^0\mathbf{A}_i]^{-1}) \\ &\delta^{i-1}\mathbf{A}_i = \begin{bmatrix} 0 & -c\alpha_i\delta\theta_i & s\alpha_i\delta\theta_i & \delta a_i \\ c\alpha_i\delta\theta_i & 0 & -\delta\alpha_i & a_i c\alpha_i\delta\theta_i + s\alpha_i\delta d_i \\ -s\alpha_i\delta\theta_i & \delta\alpha_i & 0 & -a_i s\alpha_i\delta\theta_i + c\alpha_i\delta d_i \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \quad (6)$$

From Eq. (5), the real rotation matrix ${}^0\boldsymbol{R}_4^r$ and real position vector ${}^0\mathbf{P}_4^r$ of frame {4} with respect to frame {0} can be formulated. The unknown constant error parameters $\delta\theta_i$, δd_i, δa_i and $\delta\alpha_i$ will be used as identification variables in the final objective function Eq. (13). Furthermore, according to the DH convention from Paul [6], we can find that the number of identification parameters of the serial part is equal to 12.

Similarly, taking into account the manufacturing and assembly errors, the vectors ${}^4\mathbf{a}_i$ and ${}^5\mathbf{b}_i$ will deviate from their nominal values and have constant error parameters $\delta^4\mathbf{a}_i$ and $\delta^5\mathbf{b}_i$, leg length l_i will also have an initial offset δl_i. The error model of Hexa-WH will be in the form of:

$$(l_i + \delta l_i)\mathbf{l}_i^r = {}^4\mathbf{P}_5^r + {}^4\mathbf{R}_5^r({}^5\mathbf{b}_i + \delta^5\mathbf{b}_i) - ({}^4\mathbf{a}_i + \delta^4\mathbf{a}_i), \quad i = 1, 2, \ldots, 6 \quad (7)$$

Since each joint $\mathbf{A}_i$ and $\mathbf{B}_i$ can provide 3 fixed coordinate error parameters and each leg has 1 fixed length error, the number of identification variables provided by the Hexa-WH is equal to 42.

Integrating the above error model of serial part and parallel part together, the final error model for the hybrid robot can be expressed as:

$$\begin{aligned} (l_i + \delta l_i)\mathbf{l}_i{}^r = ({}^0\mathbf{R}^r)^{-1} \lfloor {}^0\mathbf{P}_5{}^r - {}^0\mathbf{P}_4{}^r - {}^0\mathbf{R}_4{}^r({}^4\mathbf{a}_i + \delta^4\mathbf{a}_i) \\ + {}^0\mathbf{R}_5{}^r({}^5\mathbf{b}_i + \delta^5\mathbf{b}_i \rfloor, \quad i = 1, 2, \ldots, 6 \end{aligned} \quad (8)$$

In our calibration work, the real end-effector pose vector ${}^0\boldsymbol{P}_5^r$ and ${}^0\boldsymbol{R}_5^r$ can be obtained by an accurate measurement instrument and the real carriage pose vector ${}^0\boldsymbol{P}_4^r$ and ${}^0\boldsymbol{R}_4^r$ will be calculated from Eq. (5) by using the real sensor readings of the Carriage actuators.

3. MCMC method

Generally, a nonlinear model, with independent and Gaussian noise, can be presented in the form:

$$\mathbf{Y} = f(\mathbf{X}, \boldsymbol{\theta}) + \boldsymbol{\varepsilon} \tag{9}$$

The aim of this problem is to estimate the vector of unknown parameters θ based on a certain number of measurements $\mathbf{Y}$ and known input quantities $\mathbf{X}$ (constants, control variables, etc.). Bayesian approach provides a numerical method to statistically analyze the unknown parameters and its distribution. The Bayes formula is given by:

$$\pi(\boldsymbol{\theta}) = \frac{p(\mathbf{y}\left|\boldsymbol{\theta}\right.)p(\boldsymbol{\theta})}{\int p(\mathbf{y}\left|\boldsymbol{\theta}\right.)p(\boldsymbol{\theta})d\boldsymbol{\theta}} \tag{10}$$

where $p(\boldsymbol{\theta})$ is prior distribution. $p(\mathbf{y}|\boldsymbol{\theta})$ is likelihood function which gives the probability distribution of the observations $\mathbf{y}$ when given parameter values $\boldsymbol{\theta}$. The most likely values of the parameters are those that give high values for the posterior distribution $\pi(\boldsymbol{\theta})$. Assuming independent and identically distributed Gaussian error for n observations $\mathbf{y}_i$, we have

$$p(\mathbf{y}\left|\boldsymbol{\theta}\right.) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(\mathbf{y}_i - f(\mathbf{x}_i, \boldsymbol{\theta}))^2/2\sigma^2} = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-(1/2\sigma^2)SS_{\boldsymbol{\theta}}} \tag{11}$$

where $SS_{\boldsymbol{\theta}} = \sum_{i=1}^{n} (\mathbf{y}_i - f(\mathbf{x}_i, \boldsymbol{\theta}))^2$.

The intractable part of implementing Bayesian inference lies in the normalizing constant that requires integration over a high-dimensional space [7]. Fortunately, MCMC methods provide a way to solve this problem by which the need for computing these difficult integrals vanishes. The idea behind the MCMC algorithms is to generate a sequence of random variables $\{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots\}$, whose empirical distribution can asymptotically approach to the posterior distribution $\pi(\boldsymbol{\theta})$. The simplest MCMC variant is the Metropolis algorithm [5] which basically has the following steps:

Step 1: Initialization

- Set $\boldsymbol{\theta}_1 = \min_\theta \sum_{i=1}^{n} (\mathbf{y}_i - f(\mathbf{x}_i, \boldsymbol{\theta}))^2$ by using some optimization methods. In this work, Differential Evolution (DE) algorithm [8], a simple but powerful evolutionary optimization algorithm which has the ability to minimize real-valued, high nonlinear, and multi-modal objective functions, is employed to search a global optimum as the initial vector value.
- Define the length of simulation chain N_{simu}.
- Select a proposal distribution q and set $SS_{\text{old}} = SS_{\boldsymbol{\theta}_1}$.

Step 2: Simulation loop

- Generate $\boldsymbol{\theta}_{\text{new}}$ from the proposal distribution $q(\cdot|\boldsymbol{\theta}_{\text{old}})$, and compute SS_{new}.
- Calculate the acceptance probability

$$\alpha = \min\left(1, \frac{\pi(\boldsymbol{\theta}_{\text{new}})}{\pi(\boldsymbol{\theta}_{\text{old}})}\right) = \min\left(1, \frac{p(\mathbf{y}\left|\boldsymbol{\theta}_{\text{new}}\right.)}{p(\mathbf{y}\left|\boldsymbol{\theta}_{\text{old}}\right.)}\right) = \min\left(1, \exp\left\{-\frac{1}{2\sigma^2}(SS_{\text{new}} - SS_{\text{old}})\right\}\right) \tag{12}$$

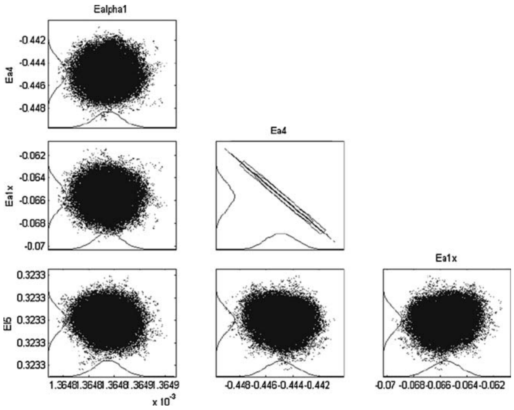


Fig. 3. Two-dimensional marginal posterior distributions for parameters $\delta\alpha_1$, δa_4, δa_{1x}, δl_5. The distributions drawn along the axis are the corresponding one-dimensional marginal density.

- The new value is accepted if $SS_{\text{new}} < SS_{\text{old}}$ or $u < \exp\{-(1/2\sigma^2)(SS_{\text{new}} - SS_{\text{old}})\}$, where u is a random number generated from U [0,1].
- Repeat the simulation loop until N_{simu} samples have been created.

4. Simulation results and analysis

In order to verify the validity and effectiveness of the MCMC-based method to estimate the kinematic error parameters of the hybrid robot, the numerical simulation is performed in this section. In the simulation, we generate a set of fixed values which can physically represent the real geometrical errors caused by manufacturing and assembly processes. Furthermore, 100 measurement poses (${}^0\mathbf{P}_5^r$ and ${}^0\mathbf{R}_5^r$) and the corresponding joint displacement of Carriage actuators are randomly generated within the robot workspace to calculate and simulate the real measured data $l_{i,j}^m$, i.e., the observation matrix $\mathbf{y}$. On the other hand, take the 54 error parameters as random variables $\boldsymbol{\theta}$ in Eq. (8) to calculate $l_{i,j}^r$. The error residuals between the measured leg length from inner sensor and the calculated leg length can be used to express objective function as

$$\boldsymbol{\theta} = \min_\theta \sum_{j=1}^{n} \sum_{i=1}^{6} (y_{i,j} - f(x_{i,j}, \theta))^2 = \min \sum_{j=1}^{n} \sum_{i=1}^{6} (l_{i,j}^m - l_{i,j}^r)^2 \tag{13}$$

In Eq. (13) n is the number of measurement points, $l_{i,j}^r$ is real leg lengths including error parameters from Eq. (8). $l_{i,j}^m$ is a certain measured value of the ith leg in the jth measurement point. The task of simulation is to obtain a posterior distribution chain for error parameters using MCMC sampling methods. The MCMC toolbox for Matlab developed by Laine et al. [9] is employed to our simulation. The obtained chain is a matrix of samples, which is commonly used to calculate the posterior means, the standard deviations and correlations, etc. After running a chain of length 200,000, we get the estimated geometrical mean values of Carriage and Hexa-WH as listed in Table 1.

From the simulation results in Table 1 we can see that the estimated mean values have successfully converged to the given fixed geometrical errors. Furthermore, the identification parameters are not too correlated. Fig. 3 gives examples of two-dimensional marginal posterior correlations of a randomly selected 4 parameters from the final model. The uncorrelated parameters have been

Table 1
Comparison of given geometrical errors and posterior means of the estimated parameters.

Symbol (nominal, error)	Nominal values	Assumed geometrical errors	Estimated mean values
α_1, $\delta\alpha_3$	90°	0.0782°	0.078197°
α_2, $\delta\alpha_2$	90°	0.0571°	0.0571°
α_3, $\delta\alpha_3$	90°	−0.048°	−0.048°
α_4, $\delta\alpha_4$	−90°	0.0417°	0.041714°
a_3, δa_3	252	−0.2164	−0.2164
a_4, δa_4	354	−0.4451	−0.44516
d_3, δd_3	422	0.1681	0.1681
d_4, δd_4	0	−0.3857	−0.38566
θ_1, $\delta\theta_1$	0	0.0213°	0.0213°
θ_2, $\delta\theta_2$	90°	0.0794°	0.0794°
θ_3, $\delta\theta_3$	0°	0.0464°	0.0464°
θ_4, $\delta\theta_4$	0°	0.0345°	0.034484°
a_{1x}, δa_{1x}	231.6663	−0.0654	−0.065343
a_{1y}, δa_{1y}	−231.9022	0.0687	0.068738
a_{1z}, δa_{1z}	0	0.0928	0.093021
a_{2x}, δa_{2x}	316.663	0.0448	0.044857
a_{2y}, δa_{2y}	−84.6778	−0.0942	−0.094162
a_{2z}, δa_{2z}	0	−0.0731	−0.072893
a_{3x}, δa_{3x}	85	0.0229	0.022957
a_{3y}, δa_{3y}	316.58	0.0133	0.013338
a_{3z}, δa_{3z}	0	−0.0136	−0.013558
a_{4x}, δa_{4x}	−85	−0.0752	−0.075143
a_{4y}, δa_{4y}	316.58	−0.0976	−0.097562
a_{4z}, δa_{4z}	0	0.0167	0.016695
a_{5x}, δa_{5x}	−316.663	0.0576	0.057657
a_{5y}, δa_{5y}	−84.6778	−0.0486	−0.048561
a_{5z}, δa_{5z}	0	0.0329	0.032932
a_{6x}, δa_{6x}	−231.6663	−0.0117	−0.011643
a_{6y}, δa_{6y}	−231.9022	0.0676	0.067639
a_{6z}, δa_{6z}	0	0.0273	0.027392
b_{1x}, δb_{1x}	32.5	0.0581	0.0581
b_{1y}, δb_{1y}	−125.93	−0.0648	−0.064799
b_{1z}, δb_{1z}	0	0.0717	0.0717
b_{2x}, δb_{2x}	125.309	0.0847	0.0847
b_{2y}, δb_{2y}	34.819	−0.0478	−0.047799
b_{2z}, δb_{2z}	0	0.0324	0.0324
b_{3x}, δb_{3x}	92.809	−0.0139	−0.0139
b_{3y}, δb_{3y}	91.111	−0.0266	−0.0266
b_{3z}, δb_{3z}	0	−0.0281	−0.0281
b_{4x}, δb_{4x}	−92.809	−0.0594	−0.059401
b_{4y}, δb_{4y}	91.111	0.0375	0.0375
b_{4z}, δb_{4z}	0	0.0088	0.0088
b_{5x}, δb_{5x}	−125.309	0.0228	0.0228
b_{5y}, δb_{5y}	34.819	−0.0566	−0.0566
b_{5z}, δb_{5z}	0	−0.0368	−0.0368
b_{6x}, δb_{6x}	−32.5	−0.0638	−0.0638
b_{6y}, δb_{6y}	−125.93	−0.0087	−0.0086997
b_{6z}, δb_{6z}	0	−0.0736	−0.0736
δl_1	0	−0.3794	−0.3794
δl_2	0	−0.0895	−0.0895
δl_3	0	0.1650	0.165
δl_4	0	−0.3048	−0.3048
δl_5	0	0.3233	0.3233
δl_6	0	0.0774	0.0774

exactly identified and the correlated parameters have only a little effect on the final estimated values.

5. Conclusions

This paper presents a MCMC-based calibration method to identify the geometrical parameter errors which are caused by manufacturing and assembly processes. A parameter identification model which has the ability to account for the geometric error sources is derived for our studied hybrid robot. Using the MCMC algorithm and the derived identification model, 54 independent kinematic error parameters of the robot are successfully identified from the calculation of mean values in posterior distribution chain. It can be seen from the simulation results that MCMC-based calibration algorithm is reliable and robust, which can be easily employed to identify error parameters for the high nonlinear kinematic models.

References

[1] B.W. Mooring, Z.S. Roth, M.R. Driels, Fundamentals of Manipulator Calibration, John Wiley and Sons Inc., New York, 1991.
[2] A. Omodel, G. Legnani, R. Adamini, Three methodologies for the calibration of industrial manipulators: experimental results on a SCARA robot, Journal of Robotic Systems 17 (6) (2000) 291–307.
[3] M. Verner, F. Xi, C. Mechefske, Optimal calibration of parallel kinematic machines, Transactions of the ASME Journal of Mechanical Design 127 (2005) 62–69.
[4] H. Wang, K.-C. Fan, Identification of strut and assembly errors of a 3-PRS serial-parallel machine tool, International Journal of Machine Tool & Manufacture 44 (2004) 1171–1178.
[5] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equations of state calculations by fast computing machines, Journal of Chemical Physics 21 (6) (1953) 1087–1092.

[6] R. Paul, Robot Manipulators: Mathematics, Programming and Control, MIT Press, Cambridge, MA, 1981.
[7] K. Gallagher, K. Charvin, S. Nielsen, M. Sambridge, J. Stephenson, Markov Chain Monte Carlo (MCMC) sampling methods to determine optimal models, model resolution and model choice for Earth Science problems, Marine and Petroleum Geology 26 (2009) 525–535.
[8] R. Storn, K. Price, Differential Evolution—A Practical Approach to Global Optimization, Springer, Berlin, 2005.
[9] M. Laine, http://www.helsinki.fi/~mjlaine/mcmc/ (retrieved on May 15, 2010).

PUBLICATION 3

Wu, H.P. & Handroos, H. & Pelab P. & Wang, Y.B. (2011)

IWR-Solution for the ITER Vacuum Vessel Assembly

The paper has been published in the *Journal of Fusion Engineering and Design, 2011, vol. 86,* p. 1834-1837

Fusion Engineering and Design 86 (2011) 1834–1837

Contents lists available at ScienceDirect

Fusion Engineering and Design

journal homepage: www.elsevier.com/locate/fusengdes

IWR-solution for the ITER vacuum vessel assembly

H. Wu [a,*], H. Handroos [a], P. Pela [b], Y. wang [a]

[a] *Laboratory of Intelligent Machines, Lappeenranta University of Technology, Finland*
[b] *Tekes, Finland*

ARTICLE INFO

Article history:
Available online 14 January 2011

Keywords:
ITER assembly
Parallel robot

ABSTRACT

The assembly of ITER vacuum vessel (VV) is still a very big challenge as the process can only be done from inside the VV. The welding of the VV assembly is carried out using the dedicated robotic systems. The main functions of the robots are: (i) measuring the actual space between every two sectors, (ii) positioning of the 150 kg splice plates between the sector shells, (iii) welding the splice plates to the sector shells, (iv) NDT of the welds, (v) repairing, including machining of the welds, (vi) He-leak tests of the welds, and (vii) the non-planned functions that may turn out. This paper presents a reasonable method to assemble the ITER VV. In this article, one parallel mobile robot, running on the track rail fixed on the wall inside the VV, is designed and tested. The assembling process, carried out by the mobile robot together with the welding robot, is presented.

1. Introduction

Assembling is one of the biggest challenges in the ITER; some critical issues are still remained to be solved. The walls of the ITER sectors are made of 60 mm thick stainless steel and are joined together by the high efficiency structural and the leak tight welds. The assembling process mainly includes: (i) preparing splice plates; (ii) transporting splice plates; (iii) welding; (iv) port assembling; (v) NDT testing; and (vi) machining and re-welding. After the initial assembling of the vacuum vessel (VV), the sectors need to be replaced for repair. The whole assembling process has to be carried out inside the VV. Because the commercially available robots are too weak and too large to carry out the required machining operations and the lifting of the possible e-beam gun column system, the conventional serial kinematic robots are lack of required stiffness and accuracy in such machining conditions. The development of the full remote welding and cutting tools, contributed by the Home Team in USA, was completed in June 1998 [1], of which the robot was built in the serial link arm on a rail-mounted vehicle moving on the guide rail, however the developed system is not able to carry out the machining process inside the ITER due to its low stiffness. Since 2000, the EFDA in EU has launched several tasks to develop an intersector welding robot (IWR) for carrying out welding and machining inside the VV [2]. The Laboratory of Intelligent Machine in Lappeenranta University of Technology participated in the related projects and has developed two generations of special hybrid machines as the solutions to the tasks. The machining, welding and handing tests have been curried out in this laboratory by cooperating with CEA in France, VTT in Finland and Ansaldo in Italy. In 2006, the EFDA evaluated the different possible methods based on the commercial serial robot, the special machines and the IWR robot. The evaluation report concluded that the hybrid parallel robot IWR is the best solution to the required tasks.

This paper analyses the key issues in assembling of the ITER VV. To fulfill the assembling task, a mobile hybrid parallel mechanism machine is introduced and the optimized assembling process, carried out by IWR cooperated with another welding robot, is proposed.

2. Requirements of VV assembly

The ITER VV consists of nine sectors and 53 port structures, which will be jointed together by the field welds. During the assembling process of the VV sectors, the customized splice plates are used to accommodate the dimensional differences between sectors so as to facilitate their relative alignment and allow access to the components surrounding the vessel. It is expected that the back-side protection is required to achieve desirable welds. All the operations should be carried out from inside of the VV [3]. Besides the joint welding between splice plates, a machine cutting process is needed for re-welding and repairing in some drawback points. A multifunction tool system is needed to fulfill the following tasks: welding, machining, splice plate handing, and easy going in and out of the VV.

The specifications in the tool system are defined as: accuracy ±0.1 mm; dynamic machining force 3 kN; handing payload 6 kN; mobility six degrees of freedom; lower mass <1 ton; speed up to 1.2 m/min.

* Corresponding author. Tel.: +358 400191656; fax: +358 56212499.
E-mail address: huapeng@lut.fi (H. Wu).

0920-3796/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.fusengdes.2010.12.050

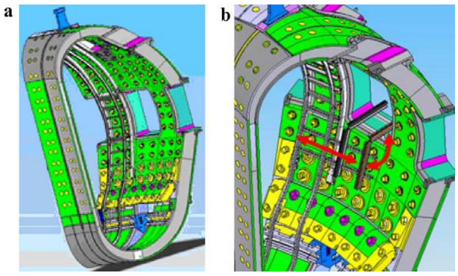


Fig. 1. Track rail inside sector.

The robot should fulfill the following tasks in several positions and structures: welding; NDT; machining; measuring; positioning; thermal shield joining; repair; cleaning; transporting; viewing; and additional actions that are related to misalignment of sectors, jams and unsatisfactory operations (e.g. welds, NDT, measuring, positioning).

To accomplish the above tasks, a mobile parallel kinematics machine (PKM) together with a simple welding robot is designed and tested.

3. Structure of assembling system

3.1. Track rail

First of all, the assembling tasks are carried out by the mobile robot, which runs on a track rail fixed on the wall of the VV. The track rail has to be assembled inside a sector before transferring to the assembly site. The rail can be fixed on the surface by utilizing flexible housing or cooling blocks in Fig. 1(a). This process raises two issues: (i) disassembling the track rail after assembling the VV and (ii) passage for robot to come in and out of the VV. The solution is found as shown in Fig. 1(b): the robot can go inside through the port with a track rail, one segment of the rail turns a certain degrees and then slide to the main track rail, the robot is then able to run in the main rail. After the assembly process has finished, the robot locates at the slidable rail and slide to the track rail at the port, the track rail at the port turns to the vertical position and the robot moves to the upper-side of the rail, then the track rail turns back to the horizontal position and the robot runs out of the VV. When the robot can go in and out of the VV, the disassembling of the track rail can be easily conducted by the robot.

3.2. Parallel kinematics machine tool

The PKM tool, which has found wide applications in industry [4], is the main tool to carry out handing, machining and other accurate tasks in the assembling process. It has a ten degrees of freedom (Fig. 2).

The PKM consists of two relatively independent sub-structures: (i) the Hexa-WH — a Stewart platform-based parallel mechanism, driven by six water hydraulic cylinders, which contributes the full six degrees of freedom for the end-effector; and (ii) the carriage, which offers the Hexa-WH four additional degrees of freedom, namely the tip motion, the rotation, the linear motion, and the tracking motion. The function of the carriage is to enlarge workspace and offer the robot a higher mobility. The robot is referred as a hybrid redundant manipulator, since it has not only the six basic degrees of freedom but also the four degrees of freedom extra provided by the carriage.

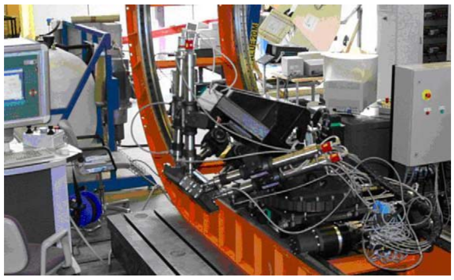


Fig. 2. Prototype of IWR.

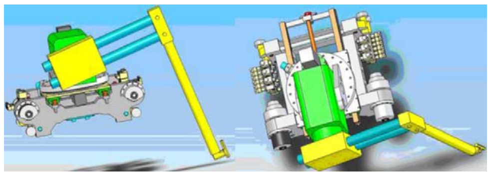


Fig. 3. Welding robot.

3.3. Welding robot

The welding robot (Fig. 3) has a carriage similar to that of the parallel robot, and the carriage has serial links with a four degrees of freedom. The robot has a simple structure and a large workspace, and mostly carries out the welding process in co-operation with the IWR robot.

3.4. Splice plate holder

Transporting the splice plates into the gap between every two adjacent sectors is a demanding task in assembling the VV. Each plate weighs more than 100 kg and should be placed accurately for welding. During the assembling, collisions may happen. The transportation operation requires a holder to have a sufficient payload capacity and flexibility. One suitable solution is the heavy duty vacuum lift mounted on the robot's end-effector (Fig. 4) [5]. In the lift, a battery-powered super efficient pump system extracts air from an integral steel reservoir, and a vacuum level monitoring system switches off the pump when the preset vacuum level is achieved. Once the system is primed, a solenoid valve mounted on the reservoir activates and extracts air from a suction pad. The pad is fitted with a replaceable hard-wear seal that can seal on a rough surface.

Fig. 4. Heavy duty lift.

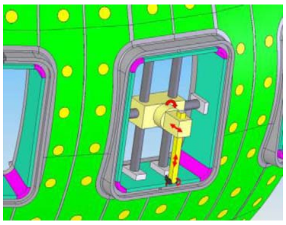


Fig. 5. Port welding robot.

3.5. *Port welding and robot*

After the sectors are welded together, it is time to assemble ports. The aforesaid welding robot cannot perform this task. Therefore, a special robot is needed; Fig. 5 shows a four degrees of freedom robot (two rotations and two linear motions). The robot is standing inside the port and supported by the beams against the port wall and tightened by screws. The robot body is made of aluminum, and it is light and simple. By screws, the lengths of the supporting beams are adjustable so that the robot can fit ports of different sizes.

4. Process cycles

The assembly of the VV sectors mainly contains the following steps: (i) preparing splice plates; (ii) splice plate transporting; (iii) tack- and multi-pass welding; (iv) NDT testing; and (v) machining and re-welding.

4.1. *Preparing splice plate*

First of all, the tolerance of the gap can be 20 mm, and the accuracy of the splice plate should be 0.1 mm for welding. Before two sectors are joined, the distance between the two sectors should be measured accurately so that a suitable splice plate can be prepared to compensate the mismatch. For measuring the distance between the two adjacent sectors, the F4E proposed a mechanical measuring system shown in Fig. 6(a).

By the proposed method, the distance of the edges between every two adjacent sectors can be measured. However the orientation of the edges cannot be measured, for example, the cases in which distances are the same while the shapes of the splice plates are different. Un-touching sensors shown in Fig. 6(b), such as Scout seam tracker (6-D laser tracker), can be applied. The Scout seam tracker can offer 6-D information of the edge and the surface for a splice plate. Two Scout sensors can be used to detect two edges in parallel. The sensors are mounted on the IWR robot's end-effector. According to the robot's position and the information from the two sensors, an accurate splice plate (±0.1 mm tolerance) can be made.

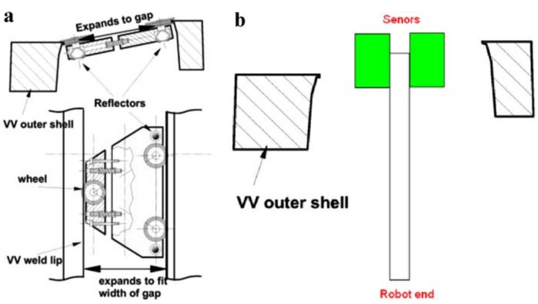


Fig. 6. (a) Mechanical measuring system and (b) untouched measuring system.

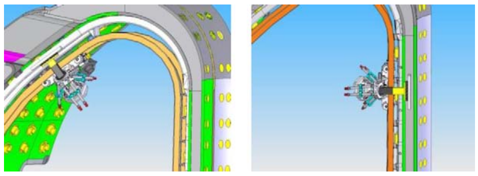


Fig. 7. Robot takes splice plate from port.

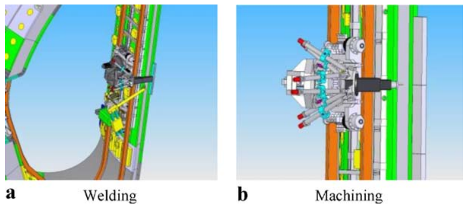


Fig. 8. Welding and milling process: (a) welding and (b) machining.

4.2. *Transporting splice plates*

A splice plate is transferred to the port by a crane lift, and then is taken by the IWR robot with the vacuum lift from the port into the position for welding (Fig. 7).

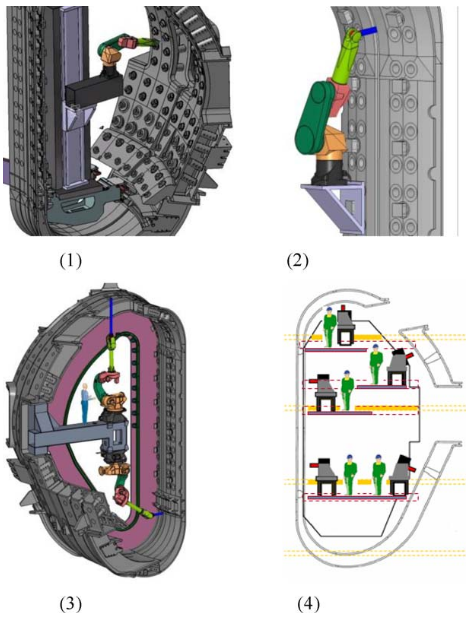


Fig. 9. Optional methods for assembly of VV.

Table 1
Main capabilities of different robots.

Robots	Payload (kg)	Dynamic work force (kg)	Repeatability (mm)	Robot mass (kg)	Mounting
Motorman HF600	600		0.5	2400	Floor
Kuka KR 240	240		0.12	1267	Floor
Kuka KR 1000	1000		0.2	4690	Floor
Kuka KR 500	500		0.15/0.3		Floor, ceiling
ABB IRB 6650S	145		0.14/0.28	2175	Shelf
OTC AX-V500	500		0.5	3000	Floor
Fauuc M-2000iA	900		0.3		Floor
Fauuc M-900iA	350		0.3		Floor, ceiling, angle, wall
IWR	600	300	<0.1	890	Movable on rails, all angles

4.3. Welding and machining

The welding processes include the tack-welding, the root-welding, and the multi-pass welding:

(i) When the IWR robot puts a splice plate in the right position, the welding robot carries out the tack-welding (Fig. 8).
(ii) The IWR robot releases the splice plate, then takes another splice plate from the port and repeats the tack-welding again.
(iii) After the splice plates are fixed, the IWR robot takes the milling tool for cutting.
(iv) After the tack-welding, the welding robot carries out the root-welding, and joins the two seams on both sides of each splice plate.
(v) The welding robot joints the ends between the splice plates after the IWR has milled the gap between the two splice plates.
(vi) After the root welding has finished, IWR conducts the NDT testing to find out whether the welds meet the quality requirements, and the defective welds need re-machining and re-welding.
(vii) Multi-pass welding is carried out by the welding robot, and each pass welding are examined by the NDT testing to find out if the welds meet the quality requirements. If needed, machining will be carried out for the defective welds.

5. Comparison

For the assembly of the VV, five possible optional methods have been investigated (Fig. 9): (1) large robots supported by a heavy mono-rail fixed on a platform travelling on CTM's; (2) small robots clamped on the VV and inserted in the holes of four housings and jammed; (3) robot on support frame anchored to the equatorial ports; (4) machine tools on multi-beam frame; and (5) the IWR parallel robot machine on track rail.

The evaluation of those methods has been given by EFDA [6]. The evaluation indexes includes: accuracy, force capability, functions of handing and machining, cost, assembly difficulty, productivity and service. Table 1 shows the most popular industrial heavy duty robots and their capabilities, those capabilities mainly include: payload, accuracy, weight and mobility. According to the comparisons in Table 1, the parallel robot IWR with a welding robot on the tack rail is one of the most suitable solutions for the assembly of the VV.

6. Conclusion

The assembly of the ITER VV is still a very big challenge, and the process can only be done from inside the VV. In this work, the IWR robot with a track rail assembled on the individual sectors for assembling the ITER VV has been studied. Some key problems have been solved by using this method. These problems include: (1) disassembling of track rail after assembling finished by using the slidable track on port; (2) preparing of splice plate by using 6-D un-touching sensor to measure the gap between sectors; (3) transferring the splice plates by the IWR with the heavy duty vacuum lift; (4) welding and machining by using the IWR and the welding robot; and (5) port assembly by using the port welding robot. Finally the comparison of different potential VV assembly methods has been given.

References

[1] K. Koizumi, L. Jones, V. Krylov, D.E. Nelson, M. Onozuka, ITER R&D: vacuum vessel and in-vessel components: vacuum vessel, Fusion Eng. Des. 55 (2001) 193–203.
[2] H. Wu, H. Handroos, J. Kovanen, A. Rouvinen, P. Hannukainen, T. Saira, L. Jones, Design of parallel intersector weld/cut robot for machining processes in ITER vacuum vessel, Fusion Eng. Des. 69 (2003) 327–331.
[3] I. Barlow, Call for nomination: vacuum vessel assembly, March 25, 2010.
[4] Stewart, A platform with six degree of freedom, Proc. Inst. Mech. Eng., Lond. 180 (1965) 371–386.
[5] http://www.vacuumlift.uk.com/page_1241799871930.html.(retrieved 20.06.2010).
[6] Vacuum Vessel design and manufacture improvements, document no. EJJ-PSM-T-0001-00, 2005.

PUBLICATION 4

Wang ,Y.B. & Pessi, P. & Wu, H.P. & Handroos, H. (2009)

Accuracy Analysis of Hybrid Parallel Robot for the Assembling of ITER

The paper has been published in the *Journal of Fusion Engineering and Design, 2009, vol. 84, No. 2,* p. 1964-1968

Fusion Engineering and Design 84 (2009) 1964–1968

Contents lists available at ScienceDirect

Fusion Engineering and Design

journal homepage: www.elsevier.com/locate/fusengdes

Accuracy analysis of hybrid parallel robot for the assembling of ITER

Yongbo wang [a,b], Pekka Pessi [a], Huapeng Wu [a,*], Heikki Handroos [a]

[a] Institute of Mechatronics and Virtual Engineering, Lappeenranta University of Technology, Skinnarilankatu 34, 53850 Lappeenranta, Finland
[b] The State Key Laboratory of Mechanical Transmission, Chongqing University, China

ARTICLE INFO

Article history:
Available online 14 January 2009

Keywords:
ITER vacuum vessel
Parallel robot
Error and stiffness modeling

ABSTRACT

This paper presents a novel mobile parallel robot, which is able to carry welding and machining processes from inside the international thermonuclear experimental reactor (ITER) vacuum vessel (VV). The kinematics design of the robot has been optimized for ITER access. To improve the accuracy of the parallel robot, the errors caused by the stiffness and manufacture process have to be compensated or limited to a minimum value. In this paper kinematics errors and stiffness modeling are given. The simulation results are presented.

1. Introduction

The vacuum vessel of ITER is composed of nine stainless steel sectors welded together and each sector is about 10 m high and 6 m wide. The sectors of ITER vacuum vessel (VV) require more stringent tolerance ($\pm$5 mm) than normally expected for the size of the structure involved. For the ITER assembling, conventional serial robots able to offer bigger work space, are very difficult to carry out the machining and welding process from inside the vacuum vessel due to their insufficient stiffness and bigger size. It is believed that parallel robot has high stiffness, accuracy and high speed than conventional serial robots. But it does not mean they have infinite stiffness and accuracy and it offers relatively small workspace. To overcome this kind of limitations and take advantage both of their merits (bigger workspace and higher stiffness), a compromised hybrid redundant robot which can be used to perform the welding, machining and remote handling is developed in Lappeenranta University of Technology. To improve the accuracy of the parallel robot, the errors caused by the stiffness and manufacture process have to be compensated or limited to a minimum value. Thus, the modeling of the stiffness and geometric errors of the robot are absolutely necessary to be built before the compensation. As the robot have 10 degrees freedom with 4 degrees redundant, the modeling is very complex and high nonlinear, this paper present methods of modeling of both stiffness and geometry error of a redundant hybrid robot, and the simulation results have been given in the paper. The stiffness modeling can be used in trajectory planning to achieve minimum deflection, and geometric error modeling can be used for the robot calibration to compensate the errors cursed by assembling and machine manufacturing of the robot, therefore the performance of the robot for the assembly of ITER are much better.

2. Kinematics analysis and error modeling

The proposed hybrid robot is shown in Fig. 1, which is connected by two parts in series, i.e., the serial part (4-DOF carriage) and the parallel part (6-DOF Hexapod mechanism) [1]. To simplify the analysis, the two parts will be first studied separately, and then combined together to obtain the final solutions.

2.1. Error modeling of the carriage

For the 4-DOF carriage mechanism, we will use the well-known Denavit–Hartenberg (D–H) convention to construct the coordinate system [2], and then derive the relative error model based on the method provided by Veitschegger and Wu [4]. The schematic diagram of the carriage mechanism is established in Fig. 2, which provides two translational movements and two rotational movements.

According to the coordinate systems established in Fig. 2, we can obtain the corresponding D–H link parameters as listed in Table 1 and the nominal D–H homogeneous transformation matrix as given in the following equation:

$$^{0}\mathbf{A}_4 = {}^{0}\mathbf{A}_1{}^{1}\mathbf{A}_2{}^{2}\mathbf{A}_3{}^{3}\mathbf{A}_4$$

* Corresponding author. Tel.: +358 5 6212435; fax: +358 5 6212499.
E-mail address: huapeng@lut.fi (H. Wu).

0920-3796/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.fusengdes.2008.11.072

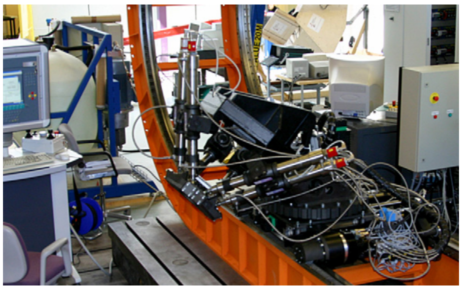


Fig. 1. The experimental prototype developed in LUT.

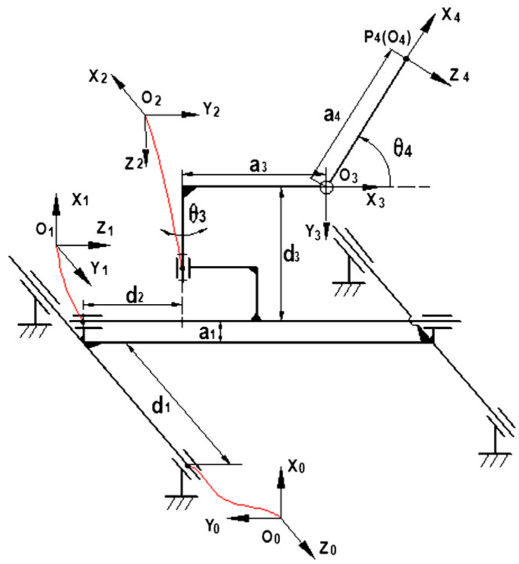


Fig. 2. Coordinate system of carriage.

$$= \begin{bmatrix} s\theta_4 & 0 & c\theta_4 & a_1 + d_3 + a_4 s\theta_4 \\ -s\theta_3 c\theta_4 & -c\theta_3 & s\theta_3 s\theta_4 & -d_2 - a_3 s\theta_3 - a_4 s\theta_3 c\theta_4 \\ c\theta_3 c\theta_4 & -s\theta_3 & -c\theta_3 s\theta_4 & d_1 + a_3 c\theta_3 + a_4 c\theta_3 c\theta_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

Based on Eq. (1), the nominal forward and inverse kinematics of the carriage can be figured out easily. Furthermore, if there are small parameter errors in robot kinematic D–H parameters θ_i, d_i, a_i, and α_i, there will be a differential change $d^{i-1}\mathbf{A}_i$ between the two consecutive joint coordinates. Therefore, the actual relationship between the two successive joint coordinates will be written as

$$^{i-1}\mathbf{A}_i^c = {}^{i-1}\mathbf{A}_i + d^{i-1}\mathbf{A}_i \tag{2}$$

where $^{i-1}\boldsymbol{A}_i$ is the homogeneous matrix which has the nominal link parameters that can express the relationship between the joint coordinates $i-1$ and i, and $d^{i-1}\mathbf{A}_i$ is the differential change due to errors in the link parameters. It can be approximated as a linear function of four kinematics errors by Taylor's series:

$$d^{i-1}\boldsymbol{A}_i = \frac{\partial^{i-1}\boldsymbol{A}_i}{\partial\theta_i}\,\Delta\theta_i + \frac{\partial^{i-1}\boldsymbol{A}_i}{\partial d_i}\,\Delta d_i + \frac{\partial^{i-1}\boldsymbol{A}_i}{\partial a_i}\,\Delta a_i + \frac{\partial^{i-1}\boldsymbol{A}_i}{\partial \alpha_i}\,\Delta \alpha_i \tag{3}$$

where $\Delta\theta_i$, Δd_i, Δa_i, and $\Delta\alpha_i$ are small errors in the D–H kinematic parameters and the partial derivatives are evaluated with the nominal geometrical link parameters. From Eq. (1), take the partial derivative with respect to θ_i, d_i, a_i, and α_i respectively, we can figure out $\partial^{i-1}A_i/\partial\theta$, $\partial^{i-1}A_i/\partial d_i$, $\partial^{i-1}A_i/\partial a_i$ and $\partial^{i-1}A_i/\partial\alpha_i$ easily.

If let $d^{i-1}\boldsymbol{A}_i = {}^{i-1}\boldsymbol{A}_i \times \delta^{i-1}\boldsymbol{A}_i$, and

$$\delta^{i-1}\boldsymbol{A}_i = \boldsymbol{D}_{\theta_i}\,\Delta\theta_i + \boldsymbol{D}_{d_i}\,\Delta d_i + \boldsymbol{D}_{a_i}\,\Delta a_i + \boldsymbol{D}_{\alpha_i}\,\Delta\alpha_i \tag{4}$$

Then expanding Eq. (4) into matrix form we can obtain

$$\delta^{i-1}\boldsymbol{A}_i = \begin{bmatrix} 0 & -c\alpha_i\,\Delta\theta_i & s\alpha_i\,\Delta\theta_i & \Delta a_i \\ c\alpha_i\,\Delta\theta_i & 0 & -\Delta\alpha_i & a_i c\alpha_i\,\Delta\theta_i + s\alpha_i\,\Delta d_i \\ -s\alpha_i\,\Delta\theta_i & \Delta\alpha_i & 0 & -a_i s\alpha_i\,\Delta\theta_i + c\alpha_i\,\Delta d_i \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{5}$$

The above expression gives the differential translation and rotation vectors for the joints which are not parallel or near parallel as functions of the four D–H kinematic errors.

In the case of the presented 4-DOF carriage, the nominal position and orientation of the task point p_4 with respect to the base frame due to the 4×4 kinematic errors can be expressed as

$$^0\boldsymbol{A}_4^c = {}^0\boldsymbol{A}_4 + d^0\boldsymbol{A}_4 = \prod_{i=1}^{4}({}^{i-1}\boldsymbol{A}_i + d^{i-1}\boldsymbol{A}_i) \tag{6}$$

Expanding Eq. (6), and ignoring second and higher-order differential errors, then the relation between the differential change in carriage and the change in link parameters can be derived as

$$d^0\boldsymbol{A}_4 = \delta\boldsymbol{A}^1 \times {}^0\boldsymbol{A}_4, \quad \delta\boldsymbol{A}^1 = \sum_{i=1}^{4}([{}^0\boldsymbol{A}_i] \times \delta^{i-1}\boldsymbol{A}_i \times [{}^0\boldsymbol{A}_i]^{-1}) \tag{7}$$

where $\delta\mathbf{A}^1$ is the first order error matrix transformation in the fixed base frame. According to Paul [5], such a differential operator has the following form:

$$\delta T = \begin{bmatrix} 0 & -\delta\theta_z & \delta\theta_y & \delta d_x \\ \delta\theta_z & 0 & -\delta\theta_x & \delta d_y \\ -\delta\theta_y & \delta\theta_x & 0 & \delta d_z \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{8}$$

If let $\delta\boldsymbol{X}_0 = \begin{bmatrix} \delta d_x & \delta d_y & \delta d_z & \delta\theta_x & \delta\theta_y & \delta\theta_z \end{bmatrix}^T \in \Re^{6\times 1}$ denote the positional and orientation errors of the carriage, then from Eqs. (7) and (8), we can get:

$$\delta X_0 = \sum_{i=1}^{4}\Delta\boldsymbol{x}_i = \sum_{i=1}^{4}(G_i\ \Delta\boldsymbol{y}_i) \tag{9}$$

where $\Delta\boldsymbol{x}_i = \begin{bmatrix} \delta dx_i & \delta dy_i & \delta dz_i & \delta\theta x_i & \delta\theta y_i & \delta\theta z_i \end{bmatrix}^T$, and , G_i is the identification Jacobian matrix. $\Delta\boldsymbol{y}_i = \begin{bmatrix} \Delta\theta_i & \Delta d_i & \Delta a_i & \Delta\alpha_i \end{bmatrix}^T$

Table 1
D–H parameters of carriage.

Joint i	α_i	a_i	d_i	θ_i
1	$\pi/2$	a_1	d_1 (var)	0
2	$\pi/2$	0	d_2 (var)	$\pi/2$
3	$\pi/2$	a_3	d_3	θ_3 (var)
4	$-\pi/2$	a_4	0	θ_4 (var)

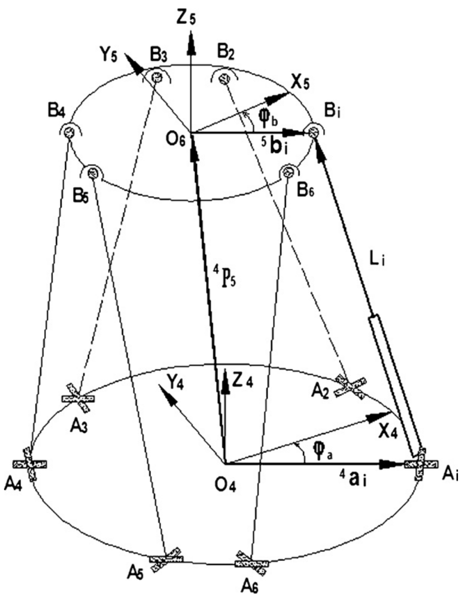


Fig. 3. Nominal model of the Hexapod parallel mechanism.

2.2. Kinematic analysis and error modeling of Hexa-WH

Fig. 3 shows a schematic diagram of hexapod parallel mechanism, for the purpose of analysis, two Cartesian coordinate systems, frames $O_4(X_4, Y_4, Z_4)$ and $O_5(X_5, Y_5, Z_5)$ are attached to the base plate and the end-effector, respectively. Six variable limbs are connected with the base plate by Universal joints and the task platform by Spherical joints.

For the nominal kinematic parameters, the following vector-loop equation represents the kinematics of the ith limb of the manipulator

$$\overrightarrow{A_iB_i} = {}^4\boldsymbol{P}_5 + {}^4\boldsymbol{R}_5{}^5\boldsymbol{b}_i - {}^4\boldsymbol{a}_i \qquad (i = 1, 2, \ldots, 6) \tag{10}$$

where ${}^4\boldsymbol{P}_5$ denotes the position vector of the task frame {5} with respect to the base frame {4}, and ${}^4\boldsymbol{R}_5$ is the Z–Y–X Euler transformation matrix expressing the orientation of the frame {5} relative to the frame {4}, and the ${}^4\boldsymbol{a}_i$, ${}^5\boldsymbol{b}_i$ represent the position vectors of U-joints A_i and S-joints B_i in the coordinate frames {4} and {5}, respectively.

Let $\mathbf{1}_i$ be the unit vector in the direction of $\overrightarrow{A_iB_i}$, and l_i represents the magnitude of the leg vector $\overrightarrow{A_iB_i}$. Differentiating both sides of Eq. (10) yields

$$\delta l_i \boldsymbol{l}_i + l_i \delta \boldsymbol{l}_i = \delta^4\boldsymbol{P}_5 + \delta^4\boldsymbol{R}_5{}^5\boldsymbol{b}_i + {}^4\boldsymbol{R}_5\delta^5\boldsymbol{b}_i - \delta^4\boldsymbol{a}_i \qquad (i = 1, 2, \ldots, 6) \tag{11}$$

Let ${}^4\boldsymbol{R}_5{}^5\boldsymbol{b}_i = \boldsymbol{s}_i$, and multiply both sides of Eq. (11) with the unit direction vector $\boldsymbol{l}_i^T$, since $\boldsymbol{l}_i^T\boldsymbol{l}_i = 1$, $\boldsymbol{l}_i^T\delta\boldsymbol{l}_i = 0$ we can obtain:

$$\begin{aligned} \delta l_i &= \boldsymbol{l}_i^T\delta^4\boldsymbol{P}_5 + \boldsymbol{l}_i^T\delta^4\boldsymbol{\Omega}_5 \times \boldsymbol{s}_i + \boldsymbol{l}_i^{T4}\boldsymbol{R}_5\delta^5\boldsymbol{b}_i - \boldsymbol{l}_i^T\delta^4\boldsymbol{a}_i \\ &= \boldsymbol{l}_i^T\delta^4\boldsymbol{P}_5 + (\boldsymbol{s}_i \times \boldsymbol{l}_i)^T\delta^4\boldsymbol{\Omega}_5 + \boldsymbol{l}_i^{T4}\boldsymbol{R}_5\delta^5\boldsymbol{b}_i - \boldsymbol{l}_i^T\delta^4\boldsymbol{a}_i \\ &= [\boldsymbol{l}_i^T (\boldsymbol{s}_i \times \boldsymbol{l}_i)^T] \begin{bmatrix} \delta^4\boldsymbol{P}_5 \\ \delta^4\boldsymbol{\Omega}_5 \end{bmatrix} + \begin{bmatrix} -\boldsymbol{l}_i^T & \boldsymbol{l}_i^{T4}\boldsymbol{R}_5 \end{bmatrix} \begin{bmatrix} \delta^4\boldsymbol{a}_i \\ \delta^5\boldsymbol{b}_i \end{bmatrix} \end{aligned} \tag{12}$$

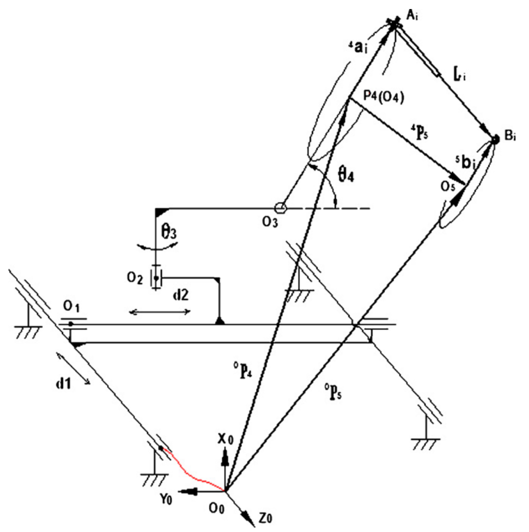


Fig. 4. Schematic diagram of IWR.

Eq. (12) can be rewritten in a matrix form as

$$\delta\boldsymbol{L} = \boldsymbol{J}_1\delta\boldsymbol{X}_1 + \boldsymbol{J}_2\delta\boldsymbol{P}_1 \tag{13}$$

Since $\boldsymbol{J}_1 \in \Re^{6\times 6}$ is a square matrix, and no singular points exist inside the workspace [3], $\mathbf{J}_1$ is invertible. Therefore Eq. (13) can be written as:

$$\delta\boldsymbol{X}_1 = \boldsymbol{J}_1^{-1}\delta\boldsymbol{L} - \boldsymbol{J}_1^{-1}\boldsymbol{J}_2\delta\boldsymbol{P}_1 \tag{14}$$

where $\delta\boldsymbol{X}_1 \in \Re^{6\times 1}$ denotes the position and orientation error vector of the end-effector. The first term on the right side represents the errors induced by actuators and the second one is the position errors from the passive joints Ai and B_i.

2.3. Kinematic analysis and error modeling of the hybrid manipulator

The schematic diagram of the redundant hybrid manipulator is shown in Fig. 4, which is a combination of carriage and Hexapod manipulator mentioned above. The base plate frame {4} of Hexapod is coincided with the end task frame of the carriage. The global base frame {0} is located at the left rail.

According to the geometry, a vector-loop equation can be derived as

$$\begin{aligned} {}^0\boldsymbol{P}_5 &= {}^0\boldsymbol{P}_4 + {}^0\boldsymbol{R}_4{}^4\boldsymbol{P}_5 = {}^0\boldsymbol{P}_4 + {}^0\boldsymbol{R}_4(l_i\boldsymbol{l}_i + {}^4\boldsymbol{a}_i - {}^4\boldsymbol{R}_5{}^5\boldsymbol{b}_i) \\ &= {}^0\boldsymbol{P}_4 + {}^0\boldsymbol{R}_4 l_i\boldsymbol{l}_i + {}^0\boldsymbol{R}_4{}^4\boldsymbol{a}_i - {}^0\boldsymbol{R}_5{}^5\boldsymbol{b}_i \end{aligned} \tag{15}$$

where ${}^0\boldsymbol{P}_5$ is the position vector of the task frame {5} (or end-effector) with respect to the fixed base frame {0}, and ${}^0\boldsymbol{R}_4$ is the rotation matrix of the frame {4} with respect to frame {0}.

Differentiating both sides of Eq. (15) and multiplying unit direction vector $\boldsymbol{l}_i^T$ yields

$$\begin{aligned} [\boldsymbol{l}_i^T(\boldsymbol{r}_{bi} \times \boldsymbol{l}_i)]^T \begin{bmatrix} \delta^0\boldsymbol{P}_5 \\ \delta^0\boldsymbol{\Omega}_5 \end{bmatrix} &= [\boldsymbol{l}_i^T(\boldsymbol{r}_{ai} \times \boldsymbol{l}_i)^T + ({}^0\boldsymbol{R}_4 l_i\boldsymbol{l}_i \times \boldsymbol{l}_i)^T] \begin{bmatrix} \delta^0\boldsymbol{P}_4 \\ \delta^0\boldsymbol{\Omega}_4 \end{bmatrix} \\ &\quad + \boldsymbol{l}_i^{T0}\boldsymbol{R}_4\boldsymbol{l}_i\delta l_i + [\boldsymbol{l}_i^{T0}\boldsymbol{R}_4 - \boldsymbol{l}_i^{T0}\boldsymbol{R}_5] \begin{bmatrix} \delta^5\boldsymbol{a}_i \\ \delta^4\boldsymbol{b}_i \end{bmatrix} \end{aligned} \tag{16}$$

where $\boldsymbol{r}_{bi} = {}^0\boldsymbol{R}_5{}^5\boldsymbol{b}_i$, $\boldsymbol{r}_{ai} = {}^0\boldsymbol{R}_4{}^4\boldsymbol{a}_i$

Eq. (16) can be rewritten in a matrix form as

$$\boldsymbol{J}_3\delta\boldsymbol{X} = \boldsymbol{J}_4\delta\boldsymbol{X}_0 + \boldsymbol{J}_5\delta\boldsymbol{L} + \boldsymbol{J}_6\delta\boldsymbol{P}_1 \tag{17}$$

Since $\boldsymbol{J}_3 \in \Re^{6\times6}$ is a square matrix, and no singular points exist inside the workspace, $\mathbf{J}_3$ is invertible. Therefore, Eq. (17) can also be rewritten as:

$$\delta\boldsymbol{X} = \boldsymbol{J}_3^{-1}\boldsymbol{J}_4\delta\boldsymbol{X}_0 + \boldsymbol{J}_3^{-1}\boldsymbol{J}_5\delta\boldsymbol{L} + \boldsymbol{J}_3^{-1}\boldsymbol{J}_6\delta\boldsymbol{P}_1 \tag{18}$$

where $\delta\boldsymbol{X} = \left[\delta^0\boldsymbol{P}_5 \quad \delta^0\boldsymbol{\Omega}_5\right]^T \in \Re^{6\times1}$ denote the final output pose errors, and the first term on the right is the errors caused by the carriage, the second and third one represent the errors induced by the Hexapod mechanism.

3. Stiffness analysis of robot

The stiffness modeling is built separately to carriage mechanism and parallel mechanism. Final stiffness of robot will combine both stiffness of carriage and parallel mechanism.

3.1. Stiffness of the carriage mechanism

The sources of stiffness for the carriage include the frame stiffness, joints' stiffness, link stiffness, and active stiffness due to the position feedback control. It is assumed that the primary source of the stiffness is the active and passive joint stiffness in the axial direction of actuation torque or force. The active stiffness is mostly dependent on the controller, the position error of actuators can be limited to a very small value by using robust control law such as PID control, and in this case we can assume that this variation is negligibly small. In this paper, however, we combine stiffness of the speed reducer, drive shafts, and the servo system into an equivalent stiffness. The stiffness of carriage frame is very difficult to be modelled because of the complex structure, the deformation mostly depend on the position of linear bearings on the track rail (the distance of d_2 as shown in Fig. 4), the stiffness of frame can be simplify to two beams and its deformation will be determined by the position d_2. The stiffness of screw driver for liner motion will be taken account of the joint stiffness. The stiffness of carriage mechanism is defined deflection of point O_4 with respect to coordinate O_0, including two translation joints d_1, d_2 and two revolution joints θ_3, θ_4 (shown in Fig. 2). For small deflections of the joints we have

$$\tau_i = \lambda_i\ \Delta\boldsymbol{q}_i \quad (i = 1, 2, 3, 4) \tag{19}$$

where τ_i is the torque or force transmitted through the ith joint, Δq_i is the corresponding deflection at the joint,

λ_i is equivalent stiffness of ith joint.

Eq. (19) can be written as

$$\tau = \lambda\,\Delta\mathbf{q} \tag{20}$$

where $\boldsymbol{\tau} = [\tau_1\ \ \tau_2\ \ \tau_3\ \ \tau_4]^{\mathrm{T}}$, $\Delta\mathbf{q} = [\Delta q_1,\ \Delta q_2,\ \Delta q_3,\ \Delta q_4]^{\mathrm{T}}$, and $\boldsymbol{\lambda} = \mathrm{diag}[\lambda_1\ \lambda_2\ \lambda_3\ \lambda_4]$ is an 4×4 diagonal matrix.

From kinematics model of carriage, the relationship between the joint displacement $\Delta\mathbf{q}$ and the displacement $\Delta\mathbf{y}$ of point O_4 can be defined as

$$\Delta\mathbf{y} = \mathbf{G}\ \Delta\mathbf{q} \tag{21}$$

where $\mathbf{G}$ is the Jacobian matrix of carriage

The force or torque $\boldsymbol{\tau}$ in joints is also related to the force $\mathbf{F}$ at point O_4 by the Jacobian matrix $\mathbf{G}$

$$\tau = \mathbf{G}^{\mathrm{T}}\mathbf{F} \tag{22}$$

From Eqs. (20)–(22), we can obtain the stiffness of the carriage.

$$\Delta\mathbf{y} = [\mathbf{G}\lambda^{-1}\mathbf{G}^{\mathrm{T}}]\mathbf{F} \tag{23}$$

Eq. (23) presents the defection error at point O_4 with respect to a force $\mathbf{F}$.

3.2. Stiffness of parallel mechanism Hexa-WH

From the solution of the inverse kinematics we can compute a stiffness matrix of Hexa-WH. The stiffness matrix is a function of the length of the cylinders. Jacobian matrix is defined as:

$$\boldsymbol{H} = [h_{i1}, h_{i2}, h_{i3}, h_{i4}, h_{i5}, h_{i6}] \tag{24}$$

where

$h_{i1} = \frac{\partial l_i}{\partial y}$; $h_{i1} = \frac{\partial l_i}{\partial z}$; $h_{i1} = \frac{\partial l_i}{\partial x\gamma}$; $h_{i1} = \frac{\partial l_i}{\partial \beta}$; $h_{i1} = \frac{\partial l_i}{\partial x\alpha}$

The stiffness matrix of the parallel manipulator has the form:

$$\boldsymbol{K} = \boldsymbol{H}^T\boldsymbol{K}_{cyl}\boldsymbol{H} \tag{25}$$

where

$\boldsymbol{K}_{cyl}$ is a diagonal matrix where the terms are spring constants of each cylinder. The spring constant varies depend on the cylinder stroke:

$$k = \frac{{A_1}^2}{((A_1x + V_h)/B_w) + (A_1x/B_c) + (V_h/B_h)} + \frac{{A_2}^2}{((A_2(l-x) + V_h)/B_w) + (A_2(l-x)/B_c) + (V_h/B_h)} \tag{26}$$

where A is area and V is volume; x is cylinder stroke and l is cylinder length; B_w, B_c and B_h is a bulk modulus of the water, cylinder and hose. The subscripts 1 and 2 refer to the chamber of the double-acting cylinder. The deflection of the end effector in reference coordinate O_4 is:

$$\Delta\boldsymbol{s} = \boldsymbol{K}^{-1}\boldsymbol{W} \tag{27}$$

where $\mathbf{W}$ is a vector $\mathbf{W}=[F_x\ F_y\ F_z\ T_x\ T_y\ T_z]^{\mathrm{T}}$, that is the forces and torques affected at the point O_5.

The stiffness of whole robot can be combined from stiffness of carriage and stiffness of parallel mechanism Hexa-WH.

4. Simulation results

In this paper the geometry errors are simulated in Matlab. In order to evaluate the final output errors caused by the error sources, a simulation example was performed using the following nominal parameters.

$$\left|{}^4\boldsymbol{a}_i\right| = 328\,\text{mm},\quad \left|{}^5\boldsymbol{b}_i\right| = 130\,\text{mm},\quad a_1 = 91\,\text{mm},\quad a_2 = 0,$$
$$a_3 = 252\,\text{mm},\quad a_4 = 354\,\text{mm};\quad d_3 = 331\,\text{mm},\quad d_4 = 0$$

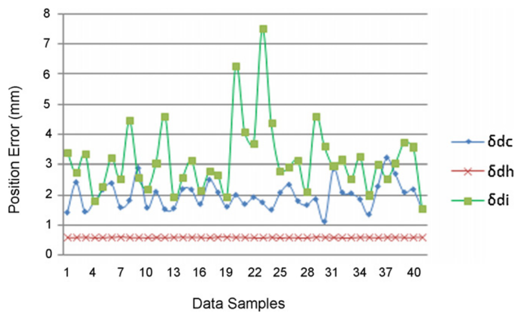


Fig. 5. Comparison of the absolute position error of carriage, Hexapod and IWR.

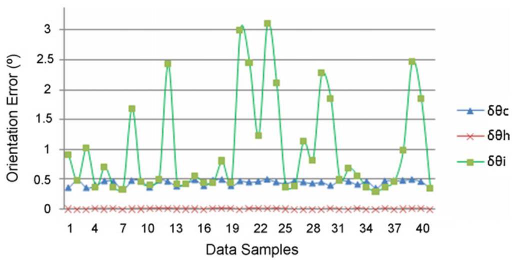


Fig. 6. Comparison of the absolute orientation error of carriage, Hexapod and IWR.

Moreover, to estimate the accuracy of the derived error model, we assume a certain kinematic errors occurred in the carriage and Hexa-WH

$$\left|\delta \boldsymbol{L}\right| = 0.5\,\text{mm}, \qquad \left|\delta \boldsymbol{P}_1\right| = 0.1\,\text{mm}, \qquad \left|\Delta\alpha_i\right| = \left|\Delta\theta_i\right| = 0.1^\circ; \qquad \left|\Delta a_i\right| = \left|\Delta d_i\right| = 0.5\,\text{mm}$$

The range of the actuator input values are given in below, which will be generated by the random function in Matlab. Figs. 5 and 6 illustrate the comparison of the absolute position and orientation error of carriage, Hexapod and the whole robot (IWR).

$$0 < d_1 < 800\,\text{mm}, \quad 0 < d_2 < 300\,\text{mm}, \quad 0^\circ < \theta_3 < 180^\circ, \quad 0^\circ < \theta_4 < 90^\circ, \quad 0^\circ < \alpha < 15^\circ, \quad 0^\circ < \beta < 15^\circ, \quad 0^\circ < \gamma < 10^\circ.$$

Comparing the absolute position and orientation errors of the carriage, Hexapod and IWR, we can see that the carriage error is the most important error sources to the final output errors, which causes about 80% of the whole errors. The final position errors are not greater than 10 mm, which can be reduced to satisfy the accuracy requirement by means of some calibration methods in next step.

5. Conclusions

In this paper, a redundant hybrid robot used for both machining and assembling of Vacuum Vessel of ITER is introduced. An error model derived for the proposed robot has the ability to account for the static sources of errors. Both geometric error and deflection error models have been derived. Due to the redundant freedom of the robot, first we divide it into serial part and parallel part, and then formulate the error model respectively, finally combine them together to get the final error model. The geometric error model has been simulated in Matlab and the results show that about 80% amount of errors in the end-effector is caused by serial link mechanism, i.e. carriage. In practice, to obtain desired accuracy of robot, these errors have to be reduced by further parameter identification methods. In stiffness model, the stiffness of parallel mechanism is mostly dependent on actuators. And for serial mechanism carriage, however, the stiffness is dependent on not only actuators, but also the links and its serial structure. Thus parallel mechanism can offer more high stiffness than serial mechanism. The deflection model can be used in optimization trajectory planning to achieve minimum deflection during the robot motion.

Acknowledgements

This work, supported by the European communities under the contract of association between EURATOM and Finnish Tekes, was carried out within the framework of the European Fusion development Agreement. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

[1] H. Wu, H. Handroos, P. Pessi, J. Kilkki, L. Jones, Development and control towards a parallel water hydraulic weld/cut for machining processes in ITER vacuum vessel, Int. J. Fusion Eng. Des. 75-79 (2005) 625–631.
[2] L.-W. Tsai, Robot Analysis—The Mechanics of Serial and Parallel Manipulators, Wiley & Sons, New York, 2000.
[3] T. Ropponen, T. Arai, Accuracy Analysis of a Modified Hexapod Manipulator, IEEE Conf. Robot. Autom. (1995) 521–524.
[4] W.K. Veitschegger, C.-H. Wu, Robot analysis based on kinematics, IEEE Conf. Robot. Autom. RA-2 (September (3)) (1986) 171–179.
[5] R.P. Paul, Robot Manipulators: Mathematics, Programming, and Control, The MIT Press, Cambridge, MA, 1982.

__PUBLICATION 5__

Wang ,Y.B. & Wu, H.P. & Handroos, H. (2012)

Accuracy Improvement of a Hybrid Robot for ITER Application Using POE Modeling Method

The paper has been submitted for review in the
Journal of Fusion Engineering and Design

Accuracy improvement of a hybrid robot for ITER application using POE modeling method

Yongbo Wang*, Huapeng Wu, Heikki Handroos

Laboratory of Intelligent Machines, Lappeenranta University of Technology, FIN-53851, Lappeenranta, Finland

This paper focuses on the kinematic calibration of a 10 degree-of-freedom (DOF) redundant serial-parallel hybrid intersector welding/cutting robot (IWR) to improve its accuracy. The robot was designed to perform the assembling and repairing tasks of the vacuum vessel (VV) of the international thermonuclear experimental reactor (ITER). By employing the product of exponentials (POE) formula, we extended the POE-based calibration method from serial robot to redundant serial-parallel hybrid robot. The proposed method combines the forward and inverse kinematics together to formulate a hybrid calibration method for serial-parallel hybrid robot. Because of the high nonlinear characteristics of the error model and too many error parameters need to be identified, the traditional iterative linear least-square algorithms cannot be used to identify the error parameters. This paper employs a global optimization algorithm, Differential Evolution (DE), to identify error parameters by solving the inverse kinematics of the hybrid robot. Furthermore, the DE algorithm was also adopted to solve the forward kinematics of the hybrid robot to verify the accuracy improvement of the end-effector using the identified error parameters. Numerical simulations were carried out by generating random assembling and manufacturing errors in the allowed tolerance range and generating a number of configuration poses in the robot workspace. Simulation of realistic experimental conditions shows that the accuracy of the end-effector can be improved to the same precision level of the given external measurement device.

Keywords: ITER, Accuracy; Differential Evolution; Hybrid robot; Product of Exponentials (POE).

1. Introduction

The assembly of vacuum vessel (VV) of the international thermonuclear experimental reactor (ITER) need to fulfill some tasks such as measuring the gap between two adjacent sectors, transporting premade splice plate to match the measured gap, welding, machining and NDT testing the sectors. All of these assembly tasks are required to be performed by a robot from the inside of ITER VV. Due to high accuracy (±0.1 mm) and big workspace requirements of the assembly robot, commercially available serial robot or parallel robot cannot be directly used. To solve this problem, a 10 degree-of-freedom (DOF) redundant serial-parallel hybrid intersector welding/cutting robot (IWR) was developed in Lappeenranta University of Technology, Finland [1, 2].

Generally, to meet a specified accuracy requirement of a robot, there are two solutions available. One is to impose stringent tolerances at the manufacturing and assembling phases, but the costs would be increased dramatically with the increase of accuracy. Alternatively, the most cost-effective way to improve robot accuracy is the kinematic calibration after the robot being assembled. By this way you only need to identify the errors mathematically and compensate them in the control software. Robot calibration can be classified into dynamic calibration and static or kinematic calibration. In most cases the kinematic calibration can satisfy the accuracy requirement effectively. This paper stresses the kinematic calibration issues including error modeling and parameter identification method for serial-parallel hybrid robot.

In our previous work [1], a hybrid modeling method was developed to calibrate the serial-parallel robot. But some redundant parameters were introduced inevitably since the hybrid model is a combination of DH model [3] and vector chain analytical model. The presence of redundant parameters would deteriorate the identification results so they have to be eliminated for higher accuracy requirement. Product of Exponential formula (POE), however, is an algorithm can be used to represent the kinematics of an open–chain mechanism as the product of exponential of twists [4], [5]. The global and geometric representation of a manipulator kinematics greatly simplifies the analysis of the mechanism and makes the POE representation superior to the DH method. In this paper, we extend this product-of-exponentials error modeling method from serial robot to serial-parallel hybrid robot.

Parameter identification involves mathematical optimization method, which can be classified into two categories. The iterative linearization method is used to find out the identification Jacobian matrix and extract error parameters by recursively solving the linear system [6]. The advantage of this method is a less computation time but it suffers from ill-conditioning in the case of the error model with redundant parameters. On the other hand, the nonlinear optimization method is adopted to minimize the errors between the measured and predicted values based on the Euclidean norm. This method is

* *Corresponding author. Tel.: +358 5 6212462; fax: +358 5 6212499*
Email address: yongbo.wang@hotmail.com (Y.Wang).

computation-intensive and redundant parameters may deteriorate the identification results but the complex computation of identification Jacobian is avoided. The comparison of some global optimization methods for benchmark or real-world applications can be found in the literatures such as [7] and [8]. In general, Differential Evolution (DE) proposed by Storn [9] is a simple but effective evolutionary algorithm to solve nonlinear, multi-modal and global optimization problems. So the DE algorithm will be employed in this paper to minimize the position error of the end-effector.

2. Kinematic and Error modeling

2.1 Kinematic model

The schematic of the proposed serial-parallel hybrid robot is shown in Fig.1. The robot, serially connected by a kinematically redundant multi-link serial mechanism (named as carriage) and a 6 degree-of-freedom hexapod parallel mechanism (named as Hexa-WH), aims to compromise between a high stiffness of parallel manipulators and a large workspace of serial manipulators.

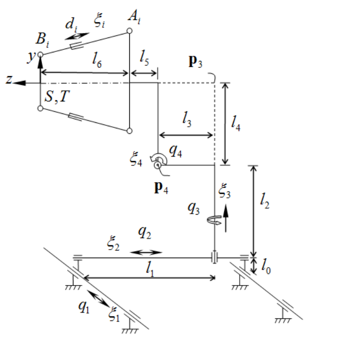


Fig. 1. Schematic of the proposed serial-parallel hybrid robot.

Due to the redundant structure, the inverse solution of the hybrid robot can have an infinite number of joint configurations for the same given end-effector configuration. But the inverse solution of the parallel mechanism can be easily obtained if the forward solution of the serial mechanism has been decided. Based on the hybrid structure and the POE formula [5], a coincident base frame S and tool frame T can be attached to the end-effector when the robot in its reference configuration. The definition of POE can be referred to Appendix A. The forward kinematics of the carriage are given by

$$g_{s5}(\mathbf{q}) = e^{\hat{\xi}_1 q_1} e^{\hat{\xi}_2 q_2} e^{\hat{\xi}_3 q_3} e^{\hat{\xi}_4 q_4} g_{s5}(0)\,. \qquad (1)$$

The inverse solution of the Hexa-WH platform is quite simple and obvious in terms of the geometry of the manipulator. Let $\mathbf{a}_{si}$,$\mathbf{b}_{si}$ be the position vector of point A_i and B_i with respect to the base frame S; and $\mathbf{a}_{5i}$,$\mathbf{b}_{ti}$ be the position vector of point A_i with respect to the tip frame of serial mechanism and tool frame T respectively. Then the extension of the prismatic joints, i.e. the nominal leg lengths of the Hexa-WH can be obtained:

$$d_i = \left\|\mathbf{b}_{si} - \mathbf{a}_{si}\right\| = \left\|g_{st}(\mathbf{q})\cdot\mathbf{b}_{ti} - g_{s5}(\mathbf{q})\cdot\mathbf{a}_{5i}\right\|, \quad i=1,2,\ldots,6. \qquad (2)$$

2.1 Nonlinear calibration model

In practice, since the manufacturing and assembling errors are unavoidable, the actual leg length would have a joint offset error, the real location of the point A_i and B_i would never agree with the designed ones, and the twist of the serial mechanism would also have some deviations, the error model of the hybrid robot can be written as

$$d_i^p = d_i + \delta d_i = \left\|\mathbf{b}_{si}^p - \mathbf{a}_{si}^p\right\|, \qquad (3)$$

where δd_i is leg joint offset, $\mathbf{b}_{si}^p = g_{st}^m(\mathbf{q})(\mathbf{b}_{ti} + \delta\mathbf{b}_{ti})$, and $g_{st}^m(\mathbf{q})$ are the measured end-effector pose frame T with respect to the base frame S, the predicted position vector of A_i can be expressed as:

$$\begin{aligned}\mathbf{a}_{si}^p &= \mathbf{a}_{si} + \delta\mathbf{a}_{si} = \mathbf{a}_{si} + \delta g_{s5}(\mathbf{q})\cdot\mathbf{a}_{5i} + g_{s5}(\mathbf{q})\cdot\delta\mathbf{a}_{5i} \\ &= \mathbf{a}_{si} + \left(\delta g_{s5}(\mathbf{q})\cdot g_{s5}(\mathbf{q})^{-1}\right)\cdot\mathbf{a}_{si} + g_{s5}(\mathbf{q})\cdot\delta\mathbf{a}_{5i}\end{aligned} \qquad (4)$$

where the error matrix $\delta g_{s5}(\mathbf{q})\cdot g_{s5}(\mathbf{q})^{-1}$ can be calculated according to the equation of (A.10).

According to the identifiability anylysis of He [5], the number of identifiable parameters of a revolute joint is 6 and a prismatic joint is 3. So the carriage will have 18 error parameters since the carriage has 2 prismatic joints and 2 revolute joints. Furthermore, each location of the spherical joint A_i and B_i provides 3 fixed coordinate error parameters, and each leg provides 1 leg joint offset error, thus the number of identification parameters from the Hexa-WH is 42.

Based on the calibration model (3), a nonlinear objective function can be formulated as the form of (5). The idea behind this nonlinear optimization method is to minimize the deviations between the measured and predicted values based on the Euclidean norm. The task of the parameter identification step is to search for a set of optimum solution of the error parameters

$$\mathbf{x} = \left[\delta\boldsymbol{\xi}_1, \delta\boldsymbol{\xi}_2, \delta\boldsymbol{\xi}_3, \delta\boldsymbol{\xi}_4, \delta d_i, \delta\mathbf{a}_i, \delta\mathbf{b}_i\right]_{60\times 1} \qquad (4)$$

to minimize: $$f(\mathbf{x}) = \sum_{j=1}^{N}\sum_{i=1}^{6}(d_{i,j}^m - d_{i,j}^p)^2 \qquad (5)$$

where $d_{i,j}^m$ and $d_{i,j}^p$ represent the i^{th} measured and predicted leg length in the j^{th} measurement configuration, N is the number of measurement configurations.

3. Simulation results and analysis

To validate the effectiveness of the proposed calibration method, some numerical simulations are carried out in this section. The object function(5) is highly nonlinear, to solve the optimization problem and identify the parameter in (4), Deferential Evolution algorithm is employed, the DE control parameters can be selected according to the scheme of DE/rand-to-best/1 [9] and the open source Matlab® code of DE from [10] is applied in the simulation. The simulations were implemented on a computer with an Intel® Core 2 Duo processor E8500, 3.16GHz and 3.25 GB of RAM. The simulation procedures are as follows:

1) Generate 100 sets of end-effector measurement poses and the corresponding carriage actuated-joint displacements within the robot workspace.

2) Assume 60 constant error parameters to represent the corresponding real physical manufacturing and assembly errors within the designed tolerance range.

3) Calculate the measured leg lengths $d_{i,j}^{m}$ according to (3) and (4).

4) Take the 60 error parameters as variables in the optimization function (5) to calculate the predicted leg lengths $d_{i,j}^{p}$ and implement the DE algorithm to search for an optimal combination of error parameters to minimize the value of the fitness function under some program terminal conditions.

To simulate the real application conditions and verify the robustness of the calibration algorithm, we assume that the end-effector poses are measured with a high resolution laser tracker measurement instrument. The position and orientation measurement accuracy are in the range of [-0.01, 0.01] mm and [-0.00001, 0.00001] rad. respectively. The measurement noise is regarded as a Gaussian distribution, with the ranges obeying the 3σ rule. Then the standard deviations of the position noise orientation noise are 0.003mm and 0.000003 rad. respectively. Fig.2 shows the result of fitness values for 4 different runs and different number of measurement poses after 6000 generations. We can see that with the increase of the measurement poses, the fitness value and the CPU time are also increased. By using the identified error parameters in the case of 50 measurement poses and DE algorithm, the position error of 25 end-effector poses can be calculated and plotted as seen in Fig.3 and Fig.4. From the simulation results in Table 1, it can be seen that the accuracy of the end-effector has been improved to the same precision level of the given external measurement device. It is noted that the end-effector pose error before calibration is dependent on the preset kinematic errors, and the accuracy of the end-effector after calibration is dependent on the accuracy of the given measurement device system.

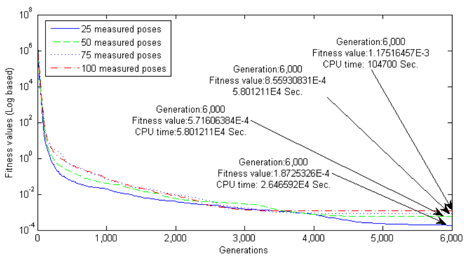


Fig.2 Fitness values with 4 different runs and different measurement poses.

Table 1: The calibration result of the end-effector before and after calibration for 25 measurement poses

Errors type	Before calibration	After calibration
RMS position	0.3604 mm	0.001mm
RMS orientation	0.0316°	0.000248°
Max. position	3.797 mm	0.0098 mm
Max. orientation	0.4778°	0.0024°

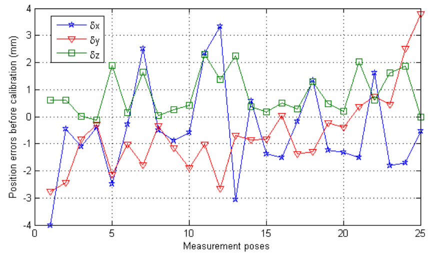


Fig.3 Plot of position errors before calibration with 25 measurement poses.

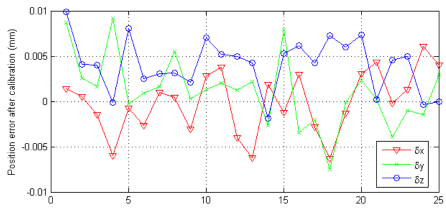


Fig.4 Plot of position errors after calibration with 25 measurement poses.

4. Conclusions and future work

In this paper, we extended the POE-based calibration method from serial robot to serial-parallel hybrid robot. The error parameters of the model, which takes into account mainly the geometrical errors originated from manufacturing and assembly processes, are identified and fitted to the given measurement data by employing Differential Evolution (DE) algorithm. The simulation results indicate that our proposed modeling and identification method for hybrid robot is robust and effective, the complete pose measurement of the end-effector is enough for the calibration, the measurement of the connection point between the serial and parallel part is not necessary. The future work will be focused on the experimental verifications of our method for the current robotic system and extend the proposed method to other serial-parallel robot to verify its practicability.

Appendix A

The below summarizes the mathematic background which related to the POE-based calibration. For more details please refer to [5], [11].

a) The Lie Group *SO(3)*, also referred as the rotation group, has the form of

$$SO(3) = \left\{ \mathbf{R} \in \Re^{3\times 3} : \mathbf{R}\mathbf{R}^T = \mathbf{I}, \det \mathbf{R} = 1 \right\}. \qquad \text{(A.1)}$$

b) The Lie Group *SE(3)*, also known in the robotics literature as the homogeneous transformation matrix, has the form of

$$SE(3) = \left\{ g = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} : \mathbf{R} \in SO(3), \mathbf{p} \in \Re^{3\times 1} \right\}. \qquad \text{(A.2)}$$

c) The Lie algebra so(3), is a vector space of the skew-symmetric matrices, such that

$$so(3) = \{ \hat{\boldsymbol{\omega}} \in \Re^{3\times 3} : \hat{\boldsymbol{\omega}}^T = -\hat{\boldsymbol{\omega}} \}, \qquad \text{(A.3)}$$

where the vector $\boldsymbol{\omega}=(\omega_x, \omega_y, \omega_z)^T \in R^{3\times 1}$, which correspondents to the axis of a rigid body rotation.

d) The Lie algebra *se(3)*, is defined as

$$se(3) = \{ \hat{\boldsymbol{\xi}} \in \begin{bmatrix} \hat{\boldsymbol{\omega}} & \mathbf{v} \\ \mathbf{0} & 0 \end{bmatrix} : \hat{\boldsymbol{\omega}} \in so(3), \mathbf{v} \in \Re^{3\times 1} \}, \quad \text{(A.4)}$$

where $\hat{\boldsymbol{\xi}}$ is termed as the twist, and $\boldsymbol{\xi}=(\boldsymbol{\omega}, \mathrm{v})^T$ is the twist coordinate of $\hat{\boldsymbol{\xi}}$. $\boldsymbol{\omega}$ is the unit directional vector of the screw axis, **v** is the position of the axis with respect to the origin. For revolute joint, if $\boldsymbol{p} \in R^{3\times 1}$ is an arbitary point on the axis, then $\mathbf{v}=-\boldsymbol{\omega}\times\mathbf{p}$. For prismatic joint, $\boldsymbol{\omega}=0$, **v** represents the unit directional vector of the axis.

e) Adjoint transformation, is a 6×6 matrix which transforms twists from one coordinate frame to another, written as Ad(g). Thus, given $g \in SE(3)$, Ad(g) can be expressed as

$$Ad(g) = \begin{bmatrix} \mathbf{R} & \mathbf{0}_{3\times 3} \\ \hat{\mathbf{b}}\mathbf{R} & \mathbf{R} \end{bmatrix}, \qquad \text{(A.5)}$$

where $\hat{\mathbf{b}}$ is the skew-symmetric matrix of vector **b**.

f) Exponential of *se(3)*, presents an important connection between the Lie Group *SE(3)* and its Lie algebra *se(3)*. Given $\hat{\boldsymbol{\xi}} \in se(3)$, $\boldsymbol{\xi}=(\boldsymbol{\omega}, \mathrm{v})^T$ and

$\|\boldsymbol{\omega}\| = \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2}$, then

$$e^{\hat{\boldsymbol{\xi}}q} = \begin{bmatrix} e^{\hat{\boldsymbol{\omega}}q} & (\mathbf{I}_3 - e^{\hat{\boldsymbol{\omega}}q})(\boldsymbol{\omega}\times\mathbf{v}) + \boldsymbol{\omega}\boldsymbol{\omega}^T\mathbf{v}q \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{b} \\ \mathbf{0} & 1 \end{bmatrix}, \quad \text{(A.6)}$$

where if $\|\boldsymbol{\omega}\|=1$, then

$$\mathbf{R} = e^{\hat{\boldsymbol{\omega}}q} = \mathbf{I}_3 + \sin(q)\hat{\boldsymbol{\omega}} + (1-\cos(q))\hat{\boldsymbol{\omega}}^2, \quad \text{(A.7)}$$

If $\|\boldsymbol{\omega}\|=0$, which means the joint is prismatic, then

$$\mathbf{R}=\mathbf{I}_3, \quad \mathbf{b}=q\mathbf{v}. \qquad \text{(A.8)}$$

g) Forward kinematics using POE formular

The forward kinematics of an *n-degree-of -freedom* serial robot is given by

$$g_{st}(\mathbf{q}) = e^{\hat{\xi}_1 q_1} e^{\hat{\xi}_1 q_1} \cdots e^{\hat{\xi}_n q_n} g_{st}(0), \qquad \text{(A.9)}$$

where $g_{st}(0)$ represents the rigid body transformation between tool frame T and base frame S when the manipulator is in its reference configuration (q=0).

h) POE based error modeling

According to the error model of He [5], if let the base frame coincident with the tool frame in the reference configuration, and assuming no errors in $g_{st}(0)$ and q, then a POE based error model can be expressed in an explicit form as

$$\begin{aligned} \left[\delta g g^{-1}\right]^{\vee} &= (\delta e^{\hat{\xi}_1 q_1} \cdot e^{-\hat{\xi}_1 q_1})^{\vee} + Ad(e^{\hat{\xi}_1 q_1})(\delta e^{\hat{\xi}_2 q_2} \cdot e^{-\hat{\xi}_2 q_2})^{\vee} \\ &+ \cdots + Ad(\prod_{i=1}^{n-1} e^{\hat{\xi}_i q_i})(\delta e^{\hat{\xi}_n q_n} \cdot e^{-\hat{\xi}_n q_n})^{\vee} \end{aligned} \qquad \text{(A.10)}$$

where

$$\begin{aligned} &(\delta e^{\hat{\xi}_i q_i} \cdot e^{-\hat{\xi}_i q_i})^{\vee} \\ &= \left(q_i \mathbf{I} + \frac{4-\theta_i \sin(\theta_i) - 4\cos(\theta_i)}{2\|\boldsymbol{\omega}\|^2}\boldsymbol{\Omega}_i + \frac{4\theta_i - 5\sin(\theta_i) + \theta_i\cos(\theta_i)}{2\|\boldsymbol{\omega}\|^3}\boldsymbol{\Omega}_i^2 \right. \\ &\left. + \frac{2-\theta_i \sin(\theta_i) - 2\cos(\theta_i)}{2\|\boldsymbol{\omega}\|^4}\boldsymbol{\Omega}_i^3 + \frac{2\theta_i - 3\sin(\theta_i) + \theta_i\cos(\theta_i)}{2\|\boldsymbol{\omega}\|^5}\boldsymbol{\Omega}_i^4 \right)\delta\boldsymbol{\xi}_i \end{aligned}$$

and

$$\boldsymbol{\Omega}_i = \begin{bmatrix} \hat{\boldsymbol{\omega}}_i & \mathbf{0}_{3\times 3} \\ \hat{v}_i & \hat{\boldsymbol{\omega}}_i \end{bmatrix}, \quad \theta_i = \|\boldsymbol{\omega}_i\| q_i, \quad \|\boldsymbol{\omega}_i\| = \sqrt{\omega_{xi}^2 + \omega_{yi}^2 + \omega_{zi}^2}$$

References

[1] Y. Wang, H. Wu, H. Handroos, Markov Chain Monte Carlo (MCMC) methods for parameter estimation of a novel hybrid redundant robot, International Journal of Fusion Engineering and Design, 86(2011), 1863−1867.

[2] H. Wu, H. Handroos, P. Pessi, J. Kilkki and L. Jones, Development and control towards a parallel water hydraulic weld/cut for machining processes in ITER Vacuum Vessel, International Journal of Fusion Engineering and Design, 75−79, 2005, 625−631

[3] R. Paul, Robot Manipulators: Mathematics, Programming, and Control, Cambridge, MA: MIT Press, 1981.

[4] K. Okamura and F. C. park, Kinematic Calibration Using the Product of Exponentials Formula, Robotica, Vol. 14, pp. 415-421, 1996.

[5] R. B. He, Y. J. Zhao, S. N. Yang, and S. Z. Yang, Kinematic-parameter Identification for Serial-robot Calibration Based on POE Formula, IEEE Trans. on Robotics, Vol. 26, No. 3 pp. 411-423, June 2010.

[6] G.Yang, I-Ming, Chen, W. K. Lim and S. H. Yeo, Self-calibration of three-legged modular reconfigurable parallel robots based on leg-end distance errors, Robotica, Vol, 19, pp. 187-198, 2001.

[7] J. Tvrdik, Competitive differential evolution and genetic algorithm in GA-DS toolbox, Available from: http://dsp.vscht.cz/konference_matlab/MATLAB06/prispevky/tvrdik/tvrdik.pdf, (retrieved on Apr. 19, 2012).

[8] S. Paterlini, T. Krink, Differential evolution and particle swarm optimisation in partitional clustering, Journal of Computational Statistics & Data Analysis, 50(2006), 1220-1247.

[9] R. Storn, K. Price, Differential Evolution – a practical approach to global optimization, Berlin: Springer, 2005.

[10] Webpage: http://www.icsi.berkeley.edu/~storn/code.html (retrieved on Jun. 10, 2012).

[11] R. M. Murray, Z. X. Li, and S. S. Sastry, A Mathematical Introduction to Robotic Manipulation. Boca Ration, Florida: CRC Press, 1994.

PUBLICATION 6

Wang ,Y.B. & Wu, H.P. & Handroos, H. (2011)

Identifiable Parameter Analysis for the Kinematic Calibration of a Hybrid Robot

The paper has been published in the *ASME 2011 International Design Engineering Technical Conferences (IDETC) and Computers and Information in Engineering Conference (CIE)*, Aug. 28-31, 2011, Washington DC, p. 1964-1968

Proceedings of the ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2011
August 28-31, 2011, Washington, DC, USA

IDETC2011 - 47573

IDENTIFIABLE PARAMETER ANALYSIS FOR THE KINEMATIC CALIBRATION OF A HYBRID ROBOT

Yongbo Wang*
Laboratory of Intelligent Machines
Lappeenranta University of Technology, Finland
Email: yongbo.wang@lut.fi

Huapeng Wu
Laboratory of Intelligent Machines
Lappeenranta University of Technology, Finland
Email: huapeng.wu@lut.fi

Heikki Handroos
Laboratory of Intelligent Machines
Lappeenranta University of Technology, Finland
Email: heikki.handroos@lut.fi

ABSTRACT

In this paper, a statistical method for the determination of the identifiable parameters of a hybrid serial-parallel robot IWR (Intersector Welding Robot) is presented. This method is based on the Markov Chain Monte Carlo (MCMC) algorithm to analyze the posterior distribution and correlation of the error parameters. Differential Evolution algorithm is employed to search a global optimizer as initial values for the random sampling of MCMC. The robot under study has ten degrees of freedom (DOF) and will be used to carry out welding, machining, and remote handing for the assembly of vacuum vessel of the international thermonuclear experimental reactor (ITER). In this paper, a kinematic error model which involves assembling and manufacturing error parameters is developed for the proposed robot. Based on this error model, the mean values of the unknown parameters are statistically analyzed and estimated using the proposed method. Computer simulations reveal that all the reduced independent kinematic parameters can be identified with the complete pose measurements. Results also demonstrate that the identification method is robust and effective with the given measurement noise.

1. INTRODUCTION

It has been acknowledged that kinematic calibration is the economical and effective way to enhance accuracy of a given robot after assembling since it only involves software modification rather than changing the mechanical structure or imposing tighter tolerances in manufacturing process. In general, a standard calibration procedure consists of 4 steps: modeling, measurement, identification and compensation. The aim of the identification step is to determine the set of error parameters for a given actual robot to compensate the nominal geometric model and match the measured data [1]. The topic of parameter identification involves numerical optimization method in which model parameters are identified from several measured robot end-effector poses. Everett et al. [2] proposed that a good kinematic model for calibration should satisfy three requirements: completeness, proportionality, and equivalence. Base on this requirement, a number of different modeling methods have been developed for kinematic modeling of robot manipulators. The most popular methods are the method proposed by Denavit and Hartengerg (DH) [3], and the modified DH method established by Hayati [4], which are widely used for serial manipulators [5]. Other alternative kinematic modeling methods to perform robot calibration exist, for example, the S-model developed by Stone, Sanderson, and Neuman [6], which uses six parameters for each link and these parameters are converted to DH parameters. The zero-reference model proposed by Mooring [7, 8] does not rely on the DH formalism and consists of establishing a reference coordinate system fixed in the work space and an end-effector coordinate system attached to the end-effector of the robot. Unlike the standard serial mechanisms applied in industrial robots, parallel mechanisms have several closed kinematic chains, so the vector chain

* Correspondence author, Phone: (358) 05 621 2462 , Email: yongbo.wang@lut.fi.

analytical method is commonly adopted in kinematic modeling of parallel robots [9, 10].

To be complete, the kinematic model should have the required number of independent and identifiable kinmatic parameters. In the case of serial robots, Khalil and Gautier [11] proposed an identification method in which the identifiable parameters are calculated from QR decomposition of the analytical observation matrix. Besnard and Khalil [12] extended this method for determining the identifiable parameters of parallel robots even in the case where the identification Jacobian matrix cannot be obtained analytically. For the kinematic calibration of hybrid robot, very few of publications can be found. Fan et al [13] presented a calibration method for a hybrid five degrees of freedom (DOF) manipulator, in the work, the serial part of the robot are taken as a ruler to measure the end-effector's offset caused by a parallel mechanism at different configurations and the calibration error model is dependent on the measurement method.

This paper proposes a novel parameter identification method for the kinematic calibration of redundant hybrid serial-parallel robot. This approach is based on the use of Markov Chain Monte Carlo (MCMC) algorithms to statistically estimate error parameters of the studied robot. The MCMC, originally introduced by Metropolis [14], which has become a common title for algorithms that simulate values from a probability distribution known only up to a normalizing constant. It has the ability to find as many as possible combinations of optimal solutions whose empirical distribution can statistically fit the data equally well within a certain required accuracy range [15]. Furthermore, the proposed MCMC-based parameter identification method can be employed to find out the correlations of the identified parameters, and then the identifiable parameters can be easily determined and the identification model can be simplified and improved to match the actual robot. Therefore, the evaluation of the condition number of identification Jacobian is not necessary.

The organization of this paper is as follows: In Section 2 we describe the kinematic modeling of the studied robot. The kinematic and preliminary identification model will be derived in this section. Section 3 gives the basic principles of MCMC algorithm and its application to the parameter estimations. Section 4 reports simulation results on different conditions and the improved identification model are determined by analyzing the correlations of the simulated parameters of the preliminary identification model. Section 5 summarizes our findings from this study.

2. DESCRIPTION AND MODELING OF THE ROBOT

A prototype of the redundant hybrid serial-parallel robot under study is shown in Fig.1, which is developed in Lappeenranta University of Technology and can be used for machining and assembling of vacuum vessel of ITER.

The robot is composed of a redundant 4-DOF multi-link serial mechanism (named as Carriage) serially connected to a standard 6-DOF Stewart parallel mechanism (named as Hexa-WH), which aims to arrive at a compromise between a high stiffness of parallel manipulators and a large workspace of serial manipulators. In what follows, we first derive a nominal kinematic model for the proposed robot. Thereafter, based on the nominal model, a related identification model including unknown parameters is developed.

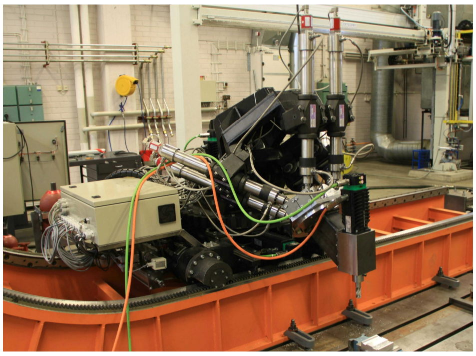


Figure1. Prototype of the studied hybrid robot

2.1 Kinematic Model

The kinematic structure of the hybrid serial-parallel robot is shown in Fig. 2. The end-effector of the robot is located in the platform coordinate frame {5} of Hexa-WH. The coordinate frame of the tip point of Carriage is coincident with the platform coordinate frame {4} of Hexa-WH. The global reference frame {0} is located at the left rail of the Carriage.

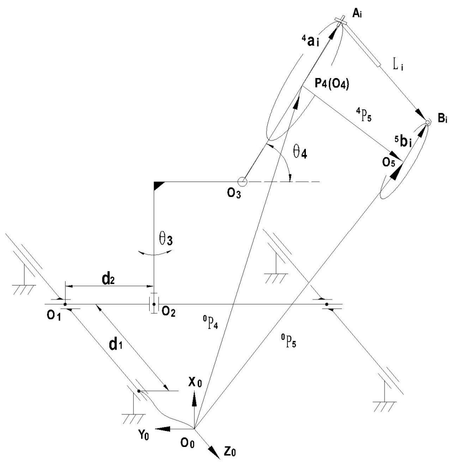


Figure 2. Structure of the studied hybrid robot

Based on this hybrid structure, a vector-loop equation is derived:

$$
\begin{aligned}
{}^{0}\mathbf{P}_5 &= {}^{0}\mathbf{P}_4 + {}^{0}\mathbf{R}_4\, {}^{4}\mathbf{P}_5 = {}^{0}\mathbf{P}_4 + {}^{0}\mathbf{R}_4\left(l_i\mathbf{l}_i + {}^{4}\mathbf{a}_i - {}^{4}\mathbf{R}_5\, {}^{5}\mathbf{b}_i\right) \\
&= {}^{0}\mathbf{P}_4 + {}^{0}\mathbf{R}_4 l_i\mathbf{l}_i + {}^{0}\mathbf{R}_4\, {}^{4}\mathbf{a}_i - {}^{0}\mathbf{R}_5\, {}^{5}\mathbf{b}_i
\end{aligned}
\tag{1}
$$

From Eqn. (1), the nominal leg length, i.e., the inverse solution of the robot can be obtained as:

$$
l_i\mathbf{l}_i = \left({}^{0}\mathbf{R}_4\right)^{-1}\left({}^{0}\mathbf{P}_5 - {}^{0}\mathbf{P}_4 - {}^{0}\mathbf{R}_4\, {}^{4}\mathbf{a}_i + {}^{0}\mathbf{R}_5\, {}^{5}\mathbf{b}_i\right) \tag{2}
$$

where ${}^{0}\mathbf{P}_5$ and ${}^{0}\mathbf{R}_5$ are the nominal position vector and rotation matrix of the end-effector frame {5} with respect to the fixed base frame {0}. ${}^{0}\mathbf{R}_4$ and ${}^{0}\mathbf{P}_4$ are the nominal rotation matrix and position vector of frame {4} with respect to frame {0}, which can be obtained from the forward kinematics of the Carriage by using the commonly used DH modeling method proposed by Paul [16]. Based on this method, the corresponding nominal forward kinematics of Carriage ${}^{0}\mathbf{T}_4$ is written as:

$$
{}^{0}\mathbf{T}_4 = {}^{0}\mathbf{A}_1\, {}^{1}\mathbf{A}_2\, {}^{2}\mathbf{A}_3\, {}^{3}\mathbf{A}_4 = \begin{bmatrix} {}^{0}\mathbf{R}_4 & {}^{0}\mathbf{P}_4 \\ 0 & 1 \end{bmatrix} \tag{3}
$$

in addition, in Eqn. (1) ${}^{4}\mathbf{P}_5$ is the position vector of the end-effector frame {5} with respect to the connection platform frame {4}. It can be calculated from the nominal inverse kinematics of Hexa-WH. Let $\mathbf{l}_i$ be the unit vector in the direction of $\mathbf{A}_i\mathbf{B}_i$, and l_i the magnitude of the leg vector $\mathbf{A}_i\mathbf{B}_i$. The following vector-loop equation represents the inverse kinematics of the i^{th} limb of the parallel manipulator:

$$
l_i\mathbf{l}_i = {}^{4}\mathbf{P}_5 + {}^{4}\mathbf{R}_5\, {}^{5}\mathbf{b}_i - {}^{4}\mathbf{a}_i,\ i = 1,2,\ldots 6 \tag{4}
$$

where ${}^{4}\mathbf{a}_i$ and ${}^{5}\mathbf{b}_i$ denote the nominal position vectors of universal joints $\mathbf{A}_i$ and spherical joints $\mathbf{B}_i$ in frame {4} and frame {5} respectively, and ${}^{4}\mathbf{R}_5$ is the Z-Y-X Euler transformation matrix representing the orientation of Frame {5} related to Frame {4}.

2.2 Preliminary Identification Model

In order for a kinematic model to be used for calibration, the model must satisfy three criteria, i.e. completeness, proportionality, and equivalence, as discussed in [1, 2]. To be complete, the model must contain sufficient number of independent parameters to describe the kinematics of the studied robot. The minimum number of geometrical parameters for serial robots is given by Mooring et al [1]

$$
\mathrm{C} = 4\mathrm{R} + 2\mathrm{P} + \mathrm{T} \tag{5}
$$

where R and P are the number of revolute and prismatic joints respectively, and T is the number of end-effector pose parameters, which are measured by the external measuring system.

Considering small geometrical errors happen to robot kinematic DH parameters θ_i, d_i, a_i and α_i, we can get the error model of the Carriage as:

$$
{}^{0}\mathbf{T}_4^r = {}^{0}\mathbf{T}_4 + d\,{}^{0}\mathbf{T}_4 = \prod_{i=1}^{4}\left({}^{i-1}\mathbf{A}_i + d\,{}^{i-1}\mathbf{A}_i\right) = \begin{bmatrix} {}^{0}\mathbf{R}_4^r & {}^{0}\mathbf{P}_4^r \\ 0 & 1 \end{bmatrix} \tag{6}
$$

Expanding Eqn. (6) and ignoring second and higher-order differential errors, it gives:

$$
d\,{}^{0}\mathbf{T}_4 = \delta\mathbf{T}^1 * {}^{0}\mathbf{T}_4\,, \quad \delta\mathbf{T}^1 = \sum_{i=1}^{4}\left(\left[{}^{0}\mathbf{A}_i\right] * \delta\,{}^{i-1}\mathbf{A}_i * \left[{}^{0}\mathbf{A}_i\right]^{-1}\right)
$$

$$
\delta\,{}^{i-1}\mathbf{A}_i = \begin{bmatrix} 0 & -c\alpha_i\delta\theta_i & s\alpha_i\delta\theta_i & \delta a_i \\ c\alpha_i\delta\theta_i & 0 & -\delta\alpha_i & a_i c\alpha_i\delta\theta_i + s\alpha_i\delta d_i \\ -s\alpha_i\delta\theta_i & \delta\alpha_i & 0 & -a_i s\alpha_i\delta\theta_i + c\alpha_i\delta d_i \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{7}
$$

Then, the actual rotation matrix ${}^{0}\mathbf{R}_4^r$ and actual position vector ${}^{0}\mathbf{P}_4^r$ of frame {4} with respect to frame {0} can be formulated from Eqn. (6). The unknown constant error parameters $\delta\theta_i$, δd_i, δa_i and $\delta\alpha_i$ are used as identification variables in the final objective function Eqn. (15).

According to eqn. (5), and noted that the external pose measurement of connection point frame {4} is not necessary, so a standard DH model of the carriage should have 12 parameters, 8 of them are from revolute joints and 4 of them from prismatic joints.

For multi-loop parallel robots, the number of independent parameters can be calculated by using the formula proposed by Vischer [17]

$$
\mathrm{C} = 3\mathrm{R} + \mathrm{P} + SS + E + 6L + 6(F-1) \tag{8}
$$

where R is the number of revolute joints, P is the number of prismatic joints, SS is the number of pairs of spherical joints, E is the number of measurement encoders, L is the number of loops and F is the number of arbitrarily located frames. According to this definition, the independent geometrical parameters of Hexa-WH is 42, which including three coordinates describing the location of the spherical joint A_i on the connection platform, three coordinates for the location of the spherical joint B_i on the end-effector platform and another one for the link encoder offset l_i for each joint link train, so the number of identification parameters provided by Hexa-WH is equal to 42.

Considering the small manufacturing and assembling tolerances in the physical structure, the identification model of Hexa-WH can be written as:

$$
\begin{aligned}
&(l_i + \delta l_i)\mathbf{l}_i^r = {}^{4}\mathbf{P}_5^r + {}^{4}\mathbf{R}_5^r({}^{5}\mathbf{b}_i + \delta\,{}^{5}\mathbf{b}_i) - ({}^{4}\mathbf{a}_i + \delta\,{}^{4}\mathbf{a}_i) \\
&i = 1,2,\ldots 6
\end{aligned}
\tag{9}
$$

Integrating the above identification model of serial part and parallel part together, the preliminary identification model for the hybrid robot can be expressed as:

$$
\begin{aligned}
&(l_i + \delta l_i)\mathbf{l}_i^r = ({}^{0}\mathbf{R}_4^r)^{-1}\left[{}^{0}\mathbf{P}_5^r - {}^{0}\mathbf{P}_4^r - {}^{0}\mathbf{R}_4^r({}^{4}\mathbf{a}_i + \delta\,{}^{4}\mathbf{a}_i) + {}^{0}\mathbf{R}_5^r({}^{5}\mathbf{b}_i + \delta\,{}^{5}\mathbf{b}_i)\right] \\
&i = 1,2,\ldots 6
\end{aligned}
\tag{10}
$$

In the calibration work, the actual end-effector pose vector ${}^{0}\mathbf{P}_5^r$ and ${}^{0}\mathbf{R}_5^r$ can be obtained by an accurate measurement instrument and the actual Carriage pose vector ${}^{0}\mathbf{P}_4^r$ and ${}^{0}\mathbf{R}_4^r$ will be calculated from Eqn. (6) using transducer readings of the Carriage actuators.

Obviously, the total number of identification parameters of the preliminary identification model is 54 if the correlations of these parameters are not taken into account. But in order for the preliminary identification model to meet the completeness requirement, the redundant and correlated parameters of the hybrid robot must be eliminated. In the following section, a MCMC-based method is introduced to determine the identifiable parameters of the preliminary identification model of the hybrid robot.

3. MCMC-BASED PARAMETER IDENTIFICATION

Generally, a nonlinear model, with independent and Gaussian noise, can be presented in the form:

$$\mathbf{Y} = f(\mathbf{X}, \boldsymbol{\theta}) + \boldsymbol{\varepsilon} \tag{11}$$

The aim of this problem is to estimate the vector of unknown parameters $\boldsymbol{\theta}$ based on a certain number of measurements $\mathbf{Y}$ and known input quantities $\mathbf{X}$ (constants, control variables, etc.). Bayesian approach provides a numerical method to statistically analyze the unknown parameters and their distributions. The Bayes formula is given as:

$$\pi(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})\, p(\boldsymbol{\theta})}{\int p(\mathbf{y}|\boldsymbol{\theta})\, p(\boldsymbol{\theta})\, d\boldsymbol{\theta}} \tag{12}$$

where p($\boldsymbol{\theta}$) is prior distribution. p($\mathbf{y}|\boldsymbol{\theta}$) is likelihood function which gives the probability distribution of the observations $\mathbf{y}$ when given parameter values $\boldsymbol{\theta}$. The most likely values of the parameters are those that give high values for the posterior distribution $\pi(\boldsymbol{\theta})$. Assuming independent and identically distributed Gaussian error for n observations $\mathbf{y}_i$, we have

$$p(\mathbf{y}|\boldsymbol{\theta}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(\mathbf{y}_i - f(\mathbf{x}_i, \boldsymbol{\theta}))^2/2\sigma^2} = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-\frac{1}{2\sigma^2} SS_{\boldsymbol{\theta}}} \tag{13}$$

where $SS_{\boldsymbol{\theta}} = \sum_{i=1}^{n} (\mathbf{y}_i - f(\mathbf{x}_i, \boldsymbol{\theta}))^2$.

The intractable part of implementing Bayesian inference lies in the normalizing constant that requires integration over a high-dimensional space [18]. Fortunately, MCMC methods provide a way to solve this problem by which the need for computing these difficult integrals vanishes. The idea behind the MCMC algorithms is to generate a sequence of random variables {$\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$, ...}, whose empirical distribution can asymptotically approach to the posterior distribution $\pi(\boldsymbol{\theta})$. The simplest MCMC variant is the Metropolis algorithm [14] which basically has the following steps:

Step 1: Initialization

- Set $\boldsymbol{\theta}_1 = \min_{\theta} \sum_{i=1}^{n} (\mathbf{y}_i - f(\mathbf{x}_i, \boldsymbol{\theta}))^2$; select a suitable numerical optimization method. In this work, Differential Evolution (DE) algorithm [19], a simple but powerful evolutionary optimization algorithm which has the ability to minimize real-valued, high nonlinear, and multi-modal objective functions, is employed to search a global optimum with certain accuracy as the initial vector values for MCMC sampling.
- Define the length of simulation chain N_{simu}.
- Select a proposal distribution q and set $SS_{old} = SS_{\boldsymbol{\theta}_1}$.

Step 2: Simulation loop

- Generate θ_{new} from the proposal distribution $q(\cdot\,|\,\boldsymbol{\theta}_{old})$, and compute SS_{new}.
- Calculate the acceptance probability
$$\alpha = \min\left(1, \frac{\pi(\boldsymbol{\theta}_{new})}{\pi(\boldsymbol{\theta}_{old})}\right) = \min\left(1, \frac{p(\mathbf{y}|\boldsymbol{\theta}_{new})}{p(\mathbf{y}|\boldsymbol{\theta}_{old})}\right)$$
$$= \min(1, \exp\left\{-\frac{1}{2\sigma^2}(SS_{new} - SS_{old})\right\}) \tag{14}$$
- The new value is accepted if $SS_{new} \prec SS_{old}$ or $u \prec \exp\left\{-\frac{1}{2\sigma^2}(SS_{new} - SS_{old})\right\}$, where u is a random number generated from U [0, 1].
- Repeat the simulation loop until N_{simu} samples have been created.

4. SIMULATION RESULTS AND ANALYSIS

The purpose of this work is to verify the validity and effectiveness of the MCMC-based method for the kinematic error parameter identification and identifiable parameter determination of the hybrid robot as shown in Fig.1. Prior to a real calibration experiment in near future, a numerical simulation is performed for a virtual prototype with realistic kinematic parameters. In the simulation, a set of preset errors which can physically represent the actual geometrical errors caused by manufacturing and assembling processes is generated by the random function in Matlab. The kinematic parameters and assumed preset errors of the hybrid robot are listed in Table1. Furthermore, two data set with 100 measurement configurations (${}^0\mathbf{P}_5^r$ and ${}^0\mathbf{R}_5^r$) and the corresponding joint displacement of Carriage actuators are randomly generated within the robot workspace to calculate and simulate the actual measured data $l_{i,j}^m$, i.e. the observation matrix $\mathbf{y}$. One data set is without noise on measurements and another data set are added Gaussian noise on each pose measurements. On the other hand, we take the error parameters from the identification model as random variables $\boldsymbol{\theta}$ in Eqn. (10) to calculate $l_{i,j}^r$. The error residuals between the measured leg length from inner sensor and the calculated leg length can be used to express objective function as

$$\boldsymbol{\theta} = \min_{\theta} \sum_{j=1}^{n}\sum_{i=1}^{6} (y_{i,j} - f(x_{i,j}, \theta))^2 = \min \sum_{j=1}^{n}\sum_{i=1}^{6} (l_{i,j}^m - l_{i,j}^r)^2 \tag{15}$$

In Eqn. (15), n is the number of measurement points, $l_{i,j}^r$ is the calculated leg length including error parameters from Eqn. (10). $l_{i,j}^m$ is a certain measured value of the ith leg in the jth measurement point. The task of simulation is to obtain a posterior distribution chain for error parameters using MCMC sampling methods. The MCMC toolbox for Matlab developed by Laine [20] is employed to our simulation. The obtained chain is a matrix of samples, which is commonly used to calculate the posterior means, the standard deviations and correlations, etc. The length of simulation chain N_{simu} is set to be 200000 in every simulation runs. In what follows, we first simulate the preliminary identification model without measurement noise to find the independent identifiable parameters, and then we simulate the improved identification model with reduced parameters by adding measurement noise and without noise to further verify our analysis.

4.1 Calibration of Preliminary Identification Model (54 Parameters) Without Measurement Noise

If the data set without noise is used, after running a chain of length 200000, we obtain a 200000×54 MCMC chain matrix for the sampled identification parameters. From it the posterior means, standard deviations, correlations and some illustrative plots such as histograms and density estimates can be calculated [15]. Table 1 gives the posterior mean values and standard deviations computed from the posterior sample matrix. The table shows that the posterior mean values of the independent parameters have been identified to be exactly the same as the preset errors with a very high precision (10^{-5} mm and 10^{-8} rad) for standard deviations, but the correlated or dependent parameters have not been identified correctly and have lower standard deviations. To find out the correlations of these dependent parameters, we select some parameters of interest to plot the two-dimensional posterior distributions as shown in fig. 3-8. From fig. 3 it can be seen that there are three parameter pairs (δa_4, δa_{1x}), (δa_4, δa_{2x}) and (δa_{1x}, δa_{2x}) are very strongly correlated and the ratio between the parameters is well identified, but the parameters themselves are not. For further analysis we can see that the parameter δa_4 is also strongly correlated with the rest x direction parameters δa_{3x}, δa_{4x}, δa_{5x}, δa_{6x} as shown in Fig. 4. The same phenomenon can be found in parameters between δd_4 and the y direction parameters δa_{1y}, δa_{2y}, δa_{3y}, δa_{4y}, δa_{5y} and δa_{6y} as shown in fig. 5-6. Some of parameters in z direction are correlated with $\delta\theta_4$ or $\delta\alpha_4$ as shown in fig. 7-8. Based on the above correlation analysis, model refinement and re-parameterization can be made. To make the model be complete and have a required number of minimal independent parameters, all of the error parameters in A_i joint can be removed, and then the geometrical errors of these joints are transferred to the remained corresponding independent parameters δa_4, δd_4, $\delta\alpha_4$ and $\delta\theta_4$ according to superposition principle. Consequently, there are 36 independent and identifiable parameters left in the improved identification model.

Table1. Nominal parameter values, preset geometrical errors and posterior mean values of the preliminary identification model with 54 parameters (without measurement noise)

Parameter (nominal, error)	Nominal values (mm, °)	Preset errors (mm, °)	Posterior mean (mm, °)	Posterior Std
α_1, $\delta\alpha_1$	-90°	0.0782°	0.07819°	2.352×10^{-8}
α_2, $\delta\alpha_2$	90°	0.0571°	0.0571°	3.1528×10^{-9}
α_3, $\delta\alpha_3$	90°	-0.048°	-0.048°	5.7552×10^{-9}
α_4, $\delta\alpha_4$	90°	0.0417°	0.04173°	1.6735×10^{-5}
a_3, δa_3	252	-0.2164	-0.2164	2.2333×10^{-5}
a_4, δa_4	354	-0.4451	-0.4451	0.0048288
d_3, δd_3	422	0.1681	0.1681	2.7881×10^{-5}
d_4, δd_4	0	-0.3857	-0.38564	0.0073678
θ_1, $\delta\theta_1$	0	0.0213°	0.0213°	2.6166×10^{-8}
θ_2, $\delta\theta_2$	90°	0.0794°	0.0794°	2.2173×10^{-8}
θ_3, $\delta\theta_3$	0°	0.0464°	0.0464°	4.8552×10^{-8}
θ_4, $\delta\theta_4$	0°	0.0345°	0.03449°	1.1837×10^{-5}
a_{1x}, δa_{1x}	-231.902	-0.0654	-0.06538	0.0048255
a_{1y}, δa_{1y}	-231.666	0.0687	0.068645	0.0073711
a_{1z}, δa_{1z}	0	0.0928	0.092879	0.0041492
a_{2x}, δa_{2x}	-84.6778	0.0448	0.044815	0.0048282
a_{2y}, δa_{2y}	-316.663	-0.0942	-0.09425	0.0073716
a_{2z}, δa_{2z}	0	-0.0731	-0.07301	0.0062136
a_{3x}, δa_{3x}	316.58	0.0229	0.02291	0.004833
a_{3y}, δa_{3y}	-85	0.0133	0.013246	0.0073733
a_{3z}, δa_{3z}	0	-0.0136	-0.01367	0.0080805
a_{4x}, δa_{4x}	316.58	-0.0752	-0.07518	0.0048307
a_{4y}, δa_{4y}	85	-0.0976	-0.09765	0.007371
a_{4z}, δa_{4z}	0	0.0167	0.016552	0.0080475
a_{5x}, δa_{5x}	-84.6778	0.0576	0.057614	0.0048249
a_{5y}, δa_{5y}	316.663	-0.0486	-0.04865	0.0073654
a_{5z}, δa_{5z}	0	0.0329	0.03272	0.0061585
a_{6x}, δa_{6x}	-231.902	-0.0117	-0.01168	0.0048258
a_{6y}, δa_{6y}	231.6663	0.0676	0.067545	0.0073668
a_{6z}, δa_{6z}	0	0.0273	0.027181	0.004128
b_{1x}, δb_{1x}	32.5	0.0581	0.058101	2.8297×10^{-5}
b_{1y}, δb_{1y}	-125.93	-0.0648	-0.0648	1.5322×10^{-5}
b_{1z}, δb_{1z}	0	0.0717	0.0717	1.6098×10^{-5}
b_{2x}, δb_{2x}	125.309	0.0847	0.084701	2.8356×10^{-5}
b_{2y}, δb_{2y}	34.819	-0.0478	-0.04779	1.4938×10^{-5}
b_{2z}, δb_{2z}	0	0.0324	0.0324	1.8154×10^{-5}
b_{3x}, δb_{3x}	92.809	-0.0139	-0.01389	2.7064×10^{-5}
b_{3y}, δb_{3y}	91.111	-0.0266	-0.02660	1.8706×10^{-5}
b_{3z}, δb_{3z}	0	-0.0281	-0.02810	2.1123×10^{-5}
b_{4x}, δb_{4x}	-92.809	-0.0594	-0.0594	2.897×10^{-5}
b_{4y}, δb_{4y}	91.111	0.0375	0.0375	1.9324×10^{-5}
b_{4z}, δb_{4z}	0	0.0088	0.008799	1.9959×10^{-5}
b_{5x}, δb_{5x}	-125.309	0.0228	0.022802	3.211×10^{-5}
b_{5y}, δb_{5y}	34.819	-0.0566	-0.0566	1.4835×10^{-5}
b_{5z}, δb_{5z}	0	-0.0368	-0.0368	1.4931×10^{-5}
b_{6x}, δb_{6x}	-32.5	-0.0638	-0.06379	3.043×10^{-5}
b_{6y}, δb_{6y}	-125.93	-0.0087	-0.00869	1.305×10^{-5}
b_{6z}, δb_{6z}	0	-0.0736	-0.0736	1.1859×10^{-5}
l_1, δl_1	0	-0.3794	-0.3794	2.3128×10^{-5}
l_2, δl_2	0	-0.0895	-0.0895	2.8054×10^{-5}
l_3, δl_3	0	0.1650	0.165	4.7325×10^{-5}
l_4, δl_4	0	-0.3048	-0.3048	5.3156×10^{-5}
l_5, δl_5	0	0.3233	0.3233	2.8076×10^{-5}
l_6, δl_6	0	0.0774	0.0774	1.5764×10^{-5}

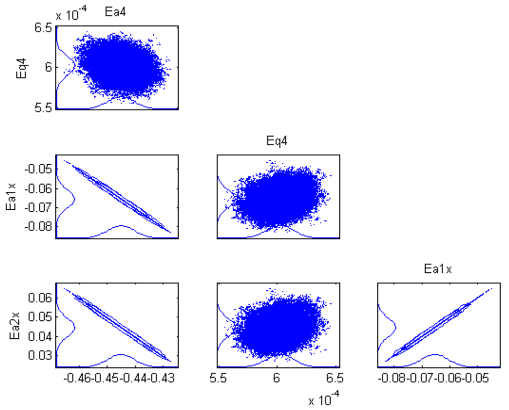


Figure 3. Pairwise scatter plots of two dimensional marginal posterior distributions for parameters δa_4 (Ea_4), $\delta\theta_4$ (Eq_4), δa_{1x} (Ea_{1x}), δa_{2x} (Ea_{2x}). The distributions plotted along the axis are the corresponding one dimensional marginal density.

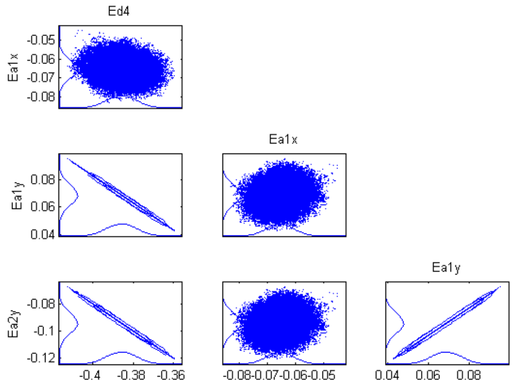


Figure 5. Pairwise scatter plots of two dimensional marginal posterior distributions for parameters δd_d (Ed_4), δa_{1y} (Ea_{1y}), δa_{2y} (Ea_{2y}), δa_{1x} (Ea_{1x}). The distributions plotted along the axis are the corresponding one dimensional marginal density.

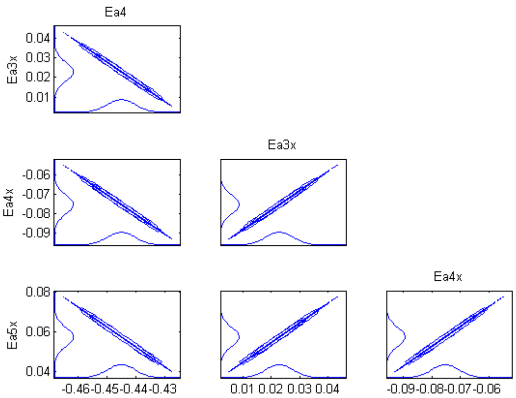


Figure 4. Pairwise scatter plots of two dimensional marginal posterior distributions for parameters δa_4 (Ea_4), δa_{3x} (Ea_{3x}), δa_{4x} (Ea_{4x}), δa_{5x} (Ea_{5x}). The distributions plotted along the axis are the corresponding one dimensional marginal density.

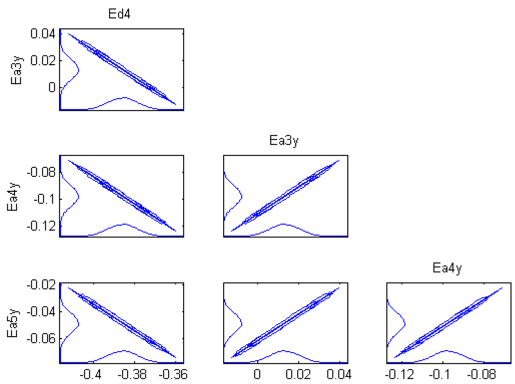


Figure 6. Pairwise scatter plots of two dimensional marginal posterior distributions for parameters δd_4 (Ed_4), δa_{3y} (Ea_{3y}), δa_{4y} (Ea_{4y}), δa_{5y} (Ea_{5y}). The distributions plotted along the axis are the corresponding one dimensional marginal density.

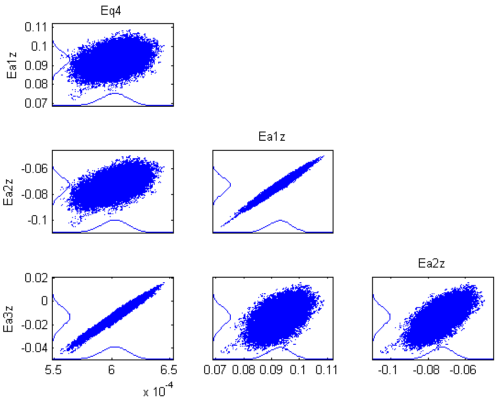


Figure 7. Pairwise scatter plots of two dimensional marginal posterior distributions for parameters $\delta\theta_4$ (Eq_4), δa_{1z} (Ea_{1z}), δa_{2z} (Ea_{2z}), δa_{3z} (Ea_{3z}). The distributions plotted along the axis are the corresponding one dimensional marginal density.

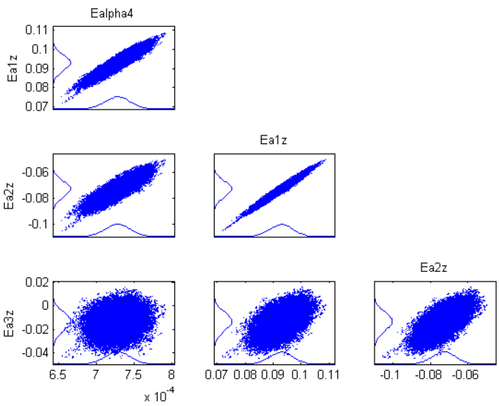


Figure 8. Pairwise scatter plots of two dimensional marginal posterior distributions for parameters $\delta\alpha_4$ ($Ealpha_4$), δa_{1z} (Ea_{1z}), δa_{2z} (Ea_{2z}), δa_{3z} (Ea_{3z}). The distributions plotted along the axis are the corresponding one dimensional marginal density.

4.2 Calibration of Improved Identification Model (36 Parameters) Without Measurement Noise

Table 2 gives the simulation results of the posterior mean values and standard deviations of the 36 parameters. It is to be noted that the correlations of the parameters have been successfully eliminated, and every parameter has been identified to be almost the same as the preset errors, and the standard deviations arrive at very high precisions (10^{-6} mm and 10^{-9} rad).

Table2. Nominal parameter values, preset geometrical errors and posterior mean values of the improved identification model with 36 parameters (without measurement noise)

Parameter (nominal, error)	Nominal values (mm, °)	Preset errors (mm, °)	Posterior mean (mm, °)	Posterior Std
α_1, $\delta\alpha_1$	-90°	0.0782°	0.0782°	2.2777×10^{-9}
α_2, $\delta\alpha_2$	90°	0.0571°	0.0571°	2.7999×10^{-9}
α_3, $\delta\alpha_3$	90°	-0.048°	-0.048°	2.6991×10^{-9}
α_4, $\delta\alpha_4$	90°	0.0417°	0.0417°	2.3243×10^{-9}
a_3, δa_3	252	-0.2164	-0.2164	8.8467×10^{-7}
a_4, δa_4	354	-0.4451	-0.4451	1.039×10^{-6}
d_3, δd_3	422	0.1681	0.1681	2.613×10^{-6}
d_4, δd_4	0	-0.3857	-0.3857	1.4035×10^{-6}
θ_1, $\delta\theta_1$	0	0.0213°	0.0213°	2.6084×10^{-9}
θ_2, $\delta\theta_2$	90°	0.0794°	0.0794°	1.2234×10^{-9}
θ_3, $\delta\theta_3$	0°	0.0464°	0.0464°	3.2017×10^{-9}
θ_4, $\delta\theta_4$	0°	0.0345°	0.0345°	1.6873×10^{-9}
b_{1x}, δb_{1x}	32.5	0.0581	0.0581	2.5743×10^{-6}
b_{1y}, δb_{1y}	-125.93	-0.0648	-0.0648	1.362×10^{-6}
b_{1z}, δb_{1z}	0	0.0717	0.0717	1.3075×10^{-6}
b_{2x}, δb_{2x}	125.309	0.0847	0.0847	2.5714×10^{-6}
b_{2y}, δb_{2y}	34.819	-0.0478	-0.0478	1.3262×10^{-6}
b_{2z}, δb_{2z}	0	0.0324	0.0324	1.1825×10^{-6}
b_{3x}, δb_{3x}	92.809	-0.0139	-0.0139	2.6619×10^{-6}
b_{3y}, δb_{3y}	91.111	-0.0266	-0.0266	1.4087×10^{-6}
b_{3z}, δb_{3z}	0	-0.0281	-0.0281	9.956×10^{-7}
b_{4x}, δb_{4x}	-92.809	-0.0594	-0.0594	2.8469×10^{-6}
b_{4y}, δb_{4y}	91.111	0.0375	0.0375	1.3211×10^{-6}
b_{4z}, δb_{4z}	0	0.0088	0.0088	9.9319×10^{-7}
b_{5x}, δb_{5x}	-125.309	0.0228	0.0228	2.982×10^{-6}
b_{5y}, δb_{5y}	34.819	-0.0566	-0.0566	1.0043×10^{-6}
b_{5z}, δb_{5z}	0	-0.0368	-0.0368	1.0054×10^{-6}
b_{6x}, δb_{6x}	-32.5	-0.0638	-0.0638	2.7734×10^{-6}
b_{6y}, δb_{6y}	-125.93	-0.0087	-0.0087	1.03×10^{-6}
b_{6z}, δb_{6z}	0	-0.0736	-0.0736	9.6405×10^{-7}
l_1, δl_1	0	-0.3794	-0.3794	1.584×10^{-6}
l_2, δl_2	0	-0.0895	-0.0895	1.372×10^{-6}
l_3, δl_3	0	0.1650	0.1650	1.3435×10^{-6}
l_4, δl_4	0	-0.3048	-0.3048	1.6412×10^{-6}
l_5, δl_5	0	0.3233	0.3233	1.5983×10^{-6}
l_6, δl_6	0	0.0774	0.0774	1.3004×10^{-6}

4.3 Calibration of Improved Identification Model (36 Parameters) With Measurement Noise

To simulate the realistic situation, we assume that the position and orientation of the end-effector will be measured with a laser tracker. The position measurement is with accuracy ±0.01mm and orientation measurement with accuracy ±0.00001rad. The measurement noise is regarded as a Gaussian distribution, with the ranges obeying the 3σ rule. The standard deviations of the position noise and orientation measurement noise are 0.003mm and 0.000003rad, respectively. Table 3 presents the simulation results of the posterior mean values and standard deviations of the reduced 36 parameters with pose measurement. The results show that all of the independent and identifiable parameters have successfully converged to the preset errors only with a slight difference, and the standard deviations arrive at very high precisions (10^{-5} mm and 10^{-8} rad). Furthermore, from Table 3 and Fig.9 we can see that every parameter is independent and identifiable. Measurement noises do have influence on the identification results, but MCMC-based identification method has ability to decrease the influence as small as possible.

Table3. Nominal parameter values, preset geometrical errors and posterior mean values of the improved identification model with 36 parameters (with measurement noise)

Parameter (nominal, error)	Nominal values (mm, °)	Preset errors (mm, °)	Posterior mean (mm, °)	Posterior Std
α_1, $\delta\alpha_1$	-90°	0.0782°	0.077859°	2.3614×10^{-8}
α_2, $\delta\alpha_2$	90°	0.0571°	0.057234°	2.699×10^{-8}
α_3, $\delta\alpha_3$	90°	-0.048°	-0.04781°	2.697×10^{-8}
α_4, $\delta\alpha_4$	90°	0.0417°	0.041479°	2.3256×10^{-8}
a_3, δa_3	252	-0.2164	-0.21465	9.0571×10^{-6}
a_4, δa_4	354	-0.4451	-0.4450	1.0434×10^{-5}
d_3, δd_3	422	0.1681	0.17487	2.6985×10^{-5}
d_4, δd_4	0	-0.3857	-0.3856	1.4193×10^{-5}
θ_1, $\delta\theta_1$	0	0.0213°	0.021623°	2.5993×10^{-8}
θ_2, $\delta\theta_2$	90°	0.0794°	0.079532°	1.2297×10^{-8}
θ_3, $\delta\theta_3$	0°	0.0464°	0.046937°	3.2475×10^{-8}
θ_4, $\delta\theta_4$	0°	0.0345°	0.034722°	1.6977×10^{-8}
b_{1x}, δb_{1x}	32.5	0.0581	0.061509	2.6428×10^{-5}
b_{1y}, δb_{1y}	-125.93	-0.0648	-0.065029	1.3881×10^{-5}
b_{1z}, δb_{1z}	0	0.0717	0.069514	1.2575×10^{-5}
b_{2x}, δb_{2x}	125.309	0.0847	0.088151	2.6055×10^{-5}
b_{2y}, δb_{2y}	34.819	-0.0478	-0.047026	1.2997×10^{-5}
b_{2z}, δb_{2z}	0	0.0324	0.030068	1.1668×10^{-5}
b_{3x}, δb_{3x}	92.809	-0.0139	-0.010355	2.6603×10^{-5}
b_{3y}, δb_{3y}	91.111	-0.0266	-0.025498	1.4205×10^{-5}
b_{3z}, δb_{3z}	0	-0.0281	-0.030009	9.9522×10^{-6}
b_{4x}, δb_{4x}	-92.809	-0.0594	-0.056781	2.8328×10^{-5}
b_{4y}, δb_{4y}	91.111	0.0375	0.038118	1.3413×10^{-5}
b_{4z}, δb_{4z}	0	0.0088	0.007394	1.0058×10^{-5}
b_{5x}, δb_{5x}	-125.309	0.0228	0.024621	2.9881×10^{-5}
b_{5y}, δb_{5y}	34.819	-0.0566	-0.055924	1.0115×10^{-5}
b_{5z}, δb_{5z}	0	-0.0368	-0.037492	1.0006×10^{-5}
b_{6x}, δb_{6x}	-32.5	-0.0638	-0.060853	2.804×10^{-5}
b_{6y}, δb_{6y}	-125.93	-0.0087	-0.008302	1.0275×10^{-5}
b_{6z}, δb_{6z}	0	-0.0736	-0.07438	9.5351×10^{-6}
l_1, δl_1	0	-0.3794	-0.38249	1.5735×10^{-5}
l_2, δl_2	0	-0.0895	-0.092193	1.3539×10^{-5}
l_3, δl_3	0	0.1650	0.16235	1.3171×10^{-5}
l_4, δl_4	0	-0.3048	-0.3082	1.6392×10^{-5}
l_5, δl_5	0	0.3233	0.31997	1.6464×10^{-5}
l_6, δl_6	0	0.0774	0.074825	1.3081×10^{-5}

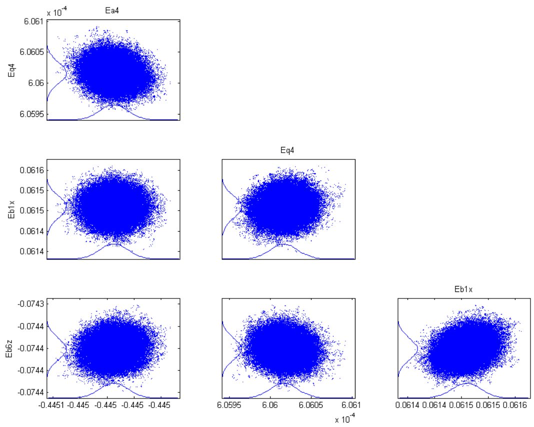


Fig.9 Pairwise scatter plots of two dimensional marginal posterior distributions for parameters δa_4 (Ea_4), $\delta\theta_4$ (Eq_4), δb_{1x} (Eb_{1x}), δb_{6z} (Eb_{6z}). The distributions plotted along the axis are the corresponding one dimensional marginal density.

5. CONCLUSIONS

In this paper, a MCMC-based kinematic calibration method for identifying the geometrical parameter errors resulting from manufacturing and assembling errors of redundant hybrid robot is reported. A preliminary parameter identification model with redundant parameters is derived for the studied redundant hybrid robot. Base on this model, MCMC algorithm is employed to statistically analyze the correlations and posterior mean values of the identified parameters. This method can be used to determine the independent identifiable parameters of a multi-redundant hybrid robot and improve the corresponding identification model without calculating condition number of the identification Jacobian matrix. The simulation results show that the MCMC-based calibration method is reliable and robust, which can be easily employed to identify error parameters of high nonlinear kinematic models.

REFERENCES

[1] B. W. Mooring, Z. S. Roth, M. R. Driels, Fundamentals of Manipulator Calibration, John Wiley and Sons Inc., New York, 1991.

[2] L.J. Everett, Morris Driels, and B.W. Mooring, Kinematic modeling for robot calibration. *In Proceedings of 1987 IEEE International Conference on Robotics and Automation*, pp. 792-797, Philadelphia, April 1986.

[3] J. Denavit, R. S. Hartenberg, Kinematic modeling for robot calibration, *Trans. ASME Journal of Applied Mechanics*, Vol. 22, June 1955, 215-221.

[4] R. P. Paul, Robot Manipulators: Mathematics, Programming, and Control,Cambridge, MA: MIT Press, 1981.

[5] S. Hayati, M. Mirmirani, Improving the absolute positioning accuracy of robot manipulators, *Journal of Robotic Systems*, Vol. 2, No. 2, 397-413.

[6] H. W. Stone, Kinematic modeling, identification and control of robotic manipulator, *Ph.D. Thesis*, Robotic Institute, Carnegie Mellon University.

[7] B. W. Mooring, G. R. Tang, An improved method for identifying the kinematic parameters in a six axis robot, in: *Proceedings of the 1983 ASME Computers in Engineering Conference,* Las Vegas, Nevada, 79-84.

[8] B. W. Mooring, Z. S. Roth, and M. R. Driels, Fundamentals of Manipulator Calibration, John Wiley and Sons Inc., New York, 1991.

[9] Tian Huang, Derek G. Chetwynd, David J. Whitehouse, Jinsong Wang, A general and novel approach for parameter identification of 6-DOF parallel kinematic machines, *Int. J. of Mechanism an Machine Theory,* 40 (2005) 219-239.

[10] David Daney, Ioannis Z. Emiris, Identification of parallel robot kinematic parameters under uncertainties by using algebraic methods, *Proceedings of the 11th World Congress in Mechanism and Machine Science*, April 1-4, Tianjin, China, 2004.

[11] W. Khalil, M. Gautier, Calculation of the identifiable parameters for robot calibration, *in 9th IFAC/IFOR Symp. on Identif. and Sys. Parameter Estimation*, pp. 888-892, Budapest, Hungaria, 1991.

[12] S.Besnard, W. Khalil, Identifiable Parameters for Parallel Robots Kinematic Calibration, *Proceedings of the 2001 IEEE International Conference on Robotics & Automation*, pp. 2859-2865, Seoul, Korea, May 21-26, 2001.

[13] Fan LiangZhi, A.Y. Elatta, Li XiaoPing, Kinematic calibration for a hybrid 5DOF manipulator based on 3-ROS in-actuated parallel manipulator*, Int. J. of Adv manuf Technol,* 25 (2005) 730-734.

[14] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equations of State Calculations by Fast Computing Machines, *Journal of Chemical Physics* , 21 (6) (1953) 1087–1092.

[15] Olli Malve, Marko Laine, Heikki Haario, Teija Kirkkala, Jouko Sarvala, Bayesian Modelling of Algal Mass Occurrences – Using Adaptive MCMC methods with a lake water quality model, *Int. J. of Environmental Modelling & Software,* 22 (2007), 966-977.

[16] R. Paul, Robot Manipulators: Mathematics, Programming, and Control, *Cambridge, MA: MIT Press*, 1981.

[17] P. Vischer, Improving the Accuracy of Parallel Robots, *PhD thesis* No.1570 (Ecole Polytechnique fédérale de Lausanne EPFL, 1996).

[18] K. Gallagher, K. Charvin, S. Nielsen, M. Sambridge, J. Stephenson, Markov Chain Monte Carlo (MCMC) sampling methods to determine optimal models, model resolution and model choice for Earth Science problems, *Marine and Petroleum Geology*, 26 (2009) 525–535.

[19] R. Storn, K. Price, Differential Evolution – a practical approach to global optimization, *Berlin: Springer*, 2005.

[20] M. Laine: http://www.helsinki.fi/~mjlaine/mcmc/ (retrieved on May 15, 2010).

PUBLICATION 7

Wang ,Y.B. & Wu, H.P. & Handroos, H. (2012)

Differential-Evolution-Based Parameter Identification Method for a Redundant Hybrid Robot Using POE Model

The paper has been published in the *43rd International Symposium on Robotics (ISR 2012)*, Aug. 29-31, 2012, Taipei, Taiwan.

The 43rd Intl. Symp. on Robotics (ISR2012),Taipei, Taiwan, Aug. 29-31, 2012

Differential-Evolution-Based Parameter Identification Method for a Redundant Hybrid Robot Using POE Model

Yongbo Wang
Laboratory of Intelligent Machines, Lappeenranta University of Technology,Finland
e-mail: yongbo.wang@lut.fi

Huapeng Wu
Laboratory of Intelligent Machines, Lappeenranta University of Technology,,Finland
e-mail: huapeng.wu@lut.fi

Heikki Handroos
Laboratory of Intelligent Machines, Lappeenranta University of Technology,,Finland
e-mail: heikki.handroos@lut.fi

***Abstract* - This paper presents a kinematic calibration method for a redundantly actuated hybrid robot to improve its absolute positioning accuracy. The studied robot is composed of a kinematically redundant serial mechanism with 4 degrees of freedom to enlarge its workspace and a standard Stewart parallel manipulator with full 6 degrees of freedom to improve its accuracy of the end-effector. It will be used to carry out welding, machining, and remote handing for the assembly of vacuum vessel of the international thermonuclear experimental reactor (ITER). Based on the product of exponentials (POE) formula, an error model involving 60 kinematic parameters is derived, which accounts for the kinematic errors originated from the manufacturing and assembly processes. Due to its hybrid serial-parallel kinematic structures and a large number of identification parameters, the traditional iterative least-square algorithm cannot be used to identify the error parameters. In this paper, by combining the forward calibration method with the inverse calibration method to formulate a hybrid calibration method, the parameter identification process is transformed into a global nonlinear optimization problem, and then Differential Evolution algorithm is employed to search a set of optimum solution from the error model to minimize the objective function. Numerical simulation reveals that all the preset error parameters can be successfully recovered under the ideal experimental condition without measurement noise. Simulation also demonstrate that the identification method is robust and effective with the given measurement noise.**

Keywords: Identification, Differential Evolution, Hybrid robot, Product of Exponentials (POE).

I. Introduction

After being assembled, the actual robot kinematic parameter values will deviate from those designed or nominal ones due to the imprecision of the assembling and fabricating processes. Kinematic parameter calibration is a very important procedure to compensate the deviations and improve the robot accuracy through software modification rather than redesigning the mechanical structures or imposing tighter tolerances in machining process. In general, robot calibration can be regarded as an integrated process consisting of modeling, measurement, identification and compensation. This paper focuses on the kinematic parameter modeling and identification issues.

Examining the existed modeling methods, we can find that most of them are focused on the pure serial or pure parallel robot without redundant structures. For serial robot, the most popular modeling methods are the DH model which proposed by Denavit and Hartengberg [1], and the Modified DH model which established by Hayati [2]. Furthermore, S-model developed by Stone, sanderson, and Neuman [3], completer and parametrically continuous (CPC) model proposed by Zhuang *et al.*[4], Mooring's Zero reference model [5], and Park and Okamura's POE model [6], Chen's local POE model [7] are also utilized in some literatures[8]. For Parallel robot, the vector chain analytical method is commonly employed, DH, POE modeling method can also be found in some publications [9], [10]. To my knowledge, there is no generic error modelling method for hybrid robot.

For the identification method, it can be classified into two categories. If the error model is simple and linearizable, then we can use the iterative linearization method to find the identification Jacobian matrix, and then recursively solves the linear system to get the optimal solution. The advantage of this method is less computation time but will suffer from ill-conditioning if there are redundant parameters exist. On the other hand, if the error model is complex and high nonlinear, then we can use nonlinear optimization method which minimize the average error between the measured and predicted values based on the Euclidean norm. This method is computation intensive and redundant parameters may degrade the identification results but the identification Jacobian is not necessary. Some global optimization algorithms such as artificial neural networks [11], genetic programming [12], and genetic algorithms (GA) [13], have been successfully employed to calibrate serial or parallel robots.

In this paper, we extend the POE-based calibration method from serial robot to serial-parallel hybrid robot with prismatic and revolute actuator joint. The method is a combination of both forward model for serial mechanism and inverse model for parallel mechanism. The error parameters of the model, which take into account mainly the geometrical errors originated from manufacturing and assembly processes, are identified and fitted to the given measurement data by employing Differential Evolution (DE) algorithm. DE is a simple but effective evolutionary algorithm for solving nonlinear and global optimization problems [14]. It has proven a superior performance both in widely used benchmark functions [15] and real-world applications [16] for identifying system parameters. The simulation results indicate that our proposed modelling and identification method for hybrid robot is robust and effective, the complete pose measurement of the end-effector is enough for the calibration, the measurement of the connection point between the serial and parallel part is not necessary.

This article is organized as follows. A brief mathematic background introduction of the POE modelling method is presented in Section Ⅱ. The kinematic and identification models of the serial-parallel hybrid robot are derived in Section Ⅲ. The implementation of DE algorithm is presented in Section Ⅳ. Simulation results are given in Section Ⅴ, and conclusions are drawn in Section Ⅵ.

II. Mathematic Background of POE Based Calibration

To facilitate the error modeling of the studied robot, some related mathematic concepts are summarized in this section. For more details refer to [8], [17].

A. POE representation for Robot kinematics

a) The Lie Group *SO(3)*, or the Special Orthogonal Group, also referred as the rotation group, has the form of

$$SO(3)=\{\mathbf{R}\in\Re^{3\times3}:\mathbf{R}\mathbf{R}^T=\mathbf{I},\det\mathbf{R}=1\}. \quad (1)$$

Every rigid body rotation about a fixed axis can be expressed as an $R\in SO(3)$.

b) The Lie Group *SE(3)*, or the Special Euclidean Group, also known in the robotics literature as the homogeneous transformation matrix, has the form of

$$SE(3)=\left\{g=\begin{bmatrix}\mathbf{R} & \mathbf{p}\\ \mathbf{0} & 1\end{bmatrix}:\mathbf{R}\in SO(3),\mathbf{p}\in\Re^{3\times1}\right\}. \quad (2)$$

SE(3) represents the group of general rigid body motions including rotation and translation.

c) The Lie algebra of *SO(3)*, denoted by *so(3)*, is a vector space of the skew-symmetric matrices, such that

$$so(3)=\{\hat{\boldsymbol{\omega}}\in\Re^{3\times3}:\hat{\boldsymbol{\omega}}^T=-\hat{\boldsymbol{\omega}}\},$$
$$\hat{\boldsymbol{\omega}}=\begin{bmatrix}0 & -\omega_z & \omega_y\\ \omega_z & 0 & -\omega_x\\ -\omega_y & \omega_x & 0\end{bmatrix} \quad (3)$$

where the vector $\boldsymbol{\omega}=(\omega_x,\omega_y,\omega_z)^T\in R^{3\times1}$,which correspondents to the axis of a rigid body rotation. The rotation can be represented in an exponential form as $\mathbf{R}=e^{\hat{\boldsymbol{\omega}}q}$, where q represents the angle of the rotation.

d) The Lie algebra of *SE(3)*, denoted by *se(3)*, is defined as

$$se(3)=\{\hat{\xi}\in\begin{bmatrix}\hat{\boldsymbol{\omega}} & \mathbf{v}\\ \mathbf{0} & 0\end{bmatrix}:\hat{\boldsymbol{\omega}}\in so(3),\mathbf{v}\in\Re^{3\times1}\}, \quad (4)$$

where $\hat{\xi}$ admits a six-dimentional vector presentation: $\xi=(\boldsymbol{\omega},\mathrm{v})^T$, termed as twist. The twist ξ represents the line coordinate of the screw axis of a general rigid body motion. $\boldsymbol{\omega}$ is the unit directional vector of the axis, $\mathbf{v}$ is the position of the axis with respect to the origin. In the exponential form, $g=e^{\hat{\xi}q}\in SE(3)$,where $q\in R$ is joint variable which represents the angle or displacement of a joint motion. For revolute joint, if $\boldsymbol{p}\in R^{3\times1}$ is an arbitary point on the axis, then $\mathbf{v}=-\boldsymbol{\omega}\times\mathbf{p}$. For prismatic joint, $\boldsymbol{\omega}=0$, $\mathbf{v}$ represents the unit directional vector of the axis.

e) Adjoint transformation, is a 6×6 matrix which transforms twists from one coordinate frame to another, written as Ad(g). Thus, given $g\in SE(3)$, Ad(g) can be expressed as

$$Ad(g)=\begin{bmatrix}\mathbf{R} & \mathbf{0}_{3\times3}\\ \hat{\mathbf{b}}\mathbf{R} & \mathbf{R}\end{bmatrix}, \quad (5)$$

where $\hat{\mathbf{b}}$ is the skew-symmetric matrix of vector b.

f) Exponential of *se(3)*, presents an important connection between a Lie Group *SE(3)* and its Lie algebra *se(3)*. Given $\hat{\xi}\in se(3)$, $\xi=(\boldsymbol{\omega},\mathrm{v})^T$ and $\|\boldsymbol{\omega}\|=\sqrt{\omega_x^2+\omega_y^2+\omega_z^2}$, then

$$e^{\hat{\xi}q}=\begin{bmatrix}e^{\hat{\boldsymbol{\omega}}q} & (\mathbf{I}_3-e^{\hat{\boldsymbol{\omega}}q})(\boldsymbol{\omega}\times\mathbf{v})+\boldsymbol{\omega}\boldsymbol{\omega}^T\mathbf{v}q\\ \mathbf{0} & 1\end{bmatrix}=\begin{bmatrix}\mathbf{R} & \mathbf{b}\\ \mathbf{0} & 1\end{bmatrix}, \quad (6)$$

where if $\|\boldsymbol{\omega}\|=1$, then

$$\mathbf{R}=e^{\hat{\boldsymbol{\omega}}q}=\mathbf{I}_3+\sin(q)\hat{\boldsymbol{\omega}}+(1-\cos(q))\hat{\boldsymbol{\omega}}^2$$
$$=\begin{bmatrix}\omega_x^2v_q+c_q & \omega_x\omega_yv_q-\omega_zs_q & \omega_x\omega_zv_q+\omega_ys_q\\ \omega_x\omega_yv_q+\omega_zs_q & \omega_y^2v_q+c_q & \omega_y\omega_zv_q-\omega_xs_q\\ \omega_x\omega_zv_q-\omega_ys_q & \omega_y\omega_zv_q+\omega_xs_q & \omega_z^2v_q+c_q\end{bmatrix}, \quad (7)$$

here c_q, s_q are abbreviations for *cos*(*q*) and *sin(q)* respectively, and $v_q=1-c_q$.

If $\|\boldsymbol{\omega}\|\neq1$,

$$\mathbf{R}=\mathbf{I}_3+\frac{\sin(\|\boldsymbol{\omega}\|q)}{\|\boldsymbol{\omega}\|}\hat{\boldsymbol{\omega}}+\frac{1-\cos(\|\boldsymbol{\omega}\|q)}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}^2, \quad (8)$$

and

$$\mathbf{b}=\left(q\mathbf{I}_3+\frac{1-\cos(\|\boldsymbol{\omega}\|q)}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}+\frac{\|\boldsymbol{\omega}\|q-\sin(\|\boldsymbol{\omega}\|q)}{\|\boldsymbol{\omega}\|^3}\hat{\boldsymbol{\omega}}^2\right)\mathbf{v} \quad (9)$$

If $\|\boldsymbol{\omega}\|=0$, which means the joint is prismatic, then

$$\mathbf{R}=\mathbf{I}_3,\quad \mathbf{b}=q\mathbf{v}. \quad (10)$$

g) Forward kinematics using POE formular

Combining the individual joint motions, the forward kinematics for an *n-degree-of -freedom* serial robot is given by

$$g_{st}(\mathbf{q})=e^{\hat{\xi}_1q_1}e^{\hat{\xi}_1q_1}\cdots e^{\hat{\xi}_nq_n}g_{st}(0), \quad (11)$$

where $g_{st}(0)$ represents the rigid body transformation between tool frame *T* and base frame *S* when the manipulator is in its reference configuration ($\mathbf{q}=0$). We can define any configuration of the manipulator as the reference configuration. One natural choice is to let the base frame be coincident with the tool frame in reference configuration, then $g_{st}(0)=\mathbf{I}$. The twist coordinates ξ_i for the individual joints of a manipulator depend on the choice of reference configuration (as well as base frame) and so the reference configuration is usually chosen such that the kinematic analysis is as simple as possible.

B. POE based error modeling

According to the error model of He [8], if let the base frame coincident with the tool frame in the reference configuration, and assuming no errors in $g_{st}(0)$ and $\mathbf{q}$, then a POE based error model can be expressed in an explicit form as

$$\begin{aligned}\left[\delta g g^{-1}\right]^{\vee} &= (\delta e^{\hat{\xi}_1 q_1} \cdot e^{-\hat{\xi}_1 q_1})^{\vee} \\ &+ Ad(e^{\hat{\xi}_1 q_1})(\delta e^{\hat{\xi}_2 q_2} \cdot e^{-\hat{\xi}_2 q_2})^{\vee} \\ &+ \cdots + Ad(\prod_{i=1}^{n-1} e^{\hat{\xi}_i q_i})(\delta e^{\hat{\xi}_n q_n} \cdot e^{-\hat{\xi}_n q_n})^{\vee}\end{aligned} \quad , (12)$$

where

$$\begin{aligned}&(\delta e^{\hat{\xi}_i q_i} \cdot e^{-\hat{\xi}_i q_i})^{\vee} \\ &= \left(q_i \mathbf{I} + \frac{4-\theta_i \sin(\theta_i) - 4\cos(\theta_i)}{2\|\boldsymbol{\omega}\|^2} \boldsymbol{\Omega}_i \right. \\ &+ \frac{4\theta_i - 5\sin(\theta_i) + \theta_i \cos(\theta_i)}{2\|\boldsymbol{\omega}\|^3} \boldsymbol{\Omega}_i^2 \\ &+ \frac{2-\theta_i \sin(\theta_i) - 2\cos(\theta_i)}{2\|\boldsymbol{\omega}\|^4} \boldsymbol{\Omega}_i^3 \\ &\left. + \frac{2\theta_i - 3\sin(\theta_i) + \theta_i \cos(\theta_i)}{2\|\boldsymbol{\omega}\|^5} \boldsymbol{\Omega}_i^4 \right) \delta\boldsymbol{\xi}_i\end{aligned} \quad , (13)$$

and

$$\boldsymbol{\Omega}_i = \begin{bmatrix} \hat{\boldsymbol{\omega}}_i & \mathbf{0}_{3\times 3} \\ \hat{v}_i & \hat{\boldsymbol{\omega}}_i \end{bmatrix}, \quad \theta_i = \|\boldsymbol{\omega}_i\| q_i .$$

$$\|\boldsymbol{\omega}_i\| = \sqrt{\omega_{xi}^2 + \omega_{yi}^2 + \omega_{zi}^2}$$

III. Error Modeling of the Hybrid Robot

The schematic of the proposed serial-parallel hybrid robot is shown in Fig.1**.** The robot, serially connected by a kinematically redundant multi-link serial mechanism (named as carriage) and a standard 6 degree-of-freedom Stewart parallel mechanism (named as Hexa-WH), aims to compromise between a high stiffness of parallel manipulators and a large workspace of serial manipulators. In the reference configuration, the base frame S and the tool frame T are coincided with each other on the end-effector. The designed or nominal kinematic parameters are listed in Table Ⅰ and Table Ⅱ.

TABLE I
KINEMATIC PARAMETERS IN THE REFERENCE CONFIGURATION

Symbols	Values (mm)	Symbols	Values (mm)
l_0	45	P_{3x}	0
l_1	320	P_{3y}	0
l_2	330	P_{3z}	-628
l_3	252	P_{4x}	0
l_4	313	P_{4y}	-313
l_5	116.84	P_{4z}	-376
l_6	259.93		

Due to the redundant structure, the inverse solution of the hybrid robot can have an infinite number of joint configurations for the same given end-effector configuration. But if the forward solution of the serial mechanism has been decided, then the inverse solution of the parallel mechanism can be easily obtained for a given end-effector configuration.

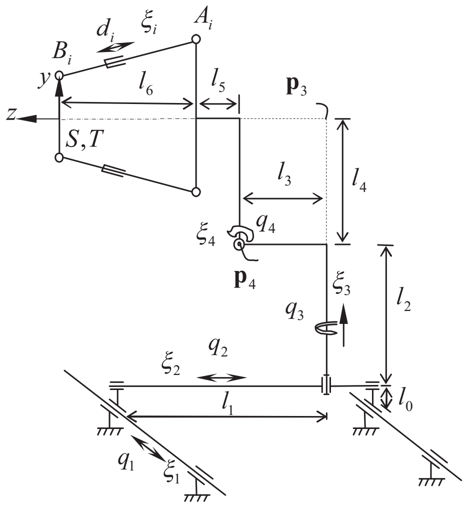


Fig. 1. Schematic of the proposed serial-parallel hybrid robot in the reference configuration.

C. A. kinematic modeling using POE

Base on the hybrid structure and the POE formula, the forward kinematics of the carriage, or the pose of the coincident point between serial and parallel mechanism, can be given by

$$g_{s5}(\mathbf{q}) = e^{\hat{\xi}_1 q_1} e^{\hat{\xi}_2 q_2} e^{\hat{\xi}_3 q_3} e^{\hat{\xi}_4 q_4} g_{s5}(0) . \quad (14)$$

The inverse solution of the Stewart platform is quite simple and obvious from the geometry of the manipulator. Let $\mathbf{a}_{si}$,$\mathbf{b}_{si}$ be the position vector of point A_i and B_i with respect to the base frame *S*; and $\mathbf{a}_{5i}$,$\mathbf{b}_{ti}$ be the position vector of point A_i with respect to the tip frame of serial mechanism and tool frame T. Then the extension of the prismatic joints, i.e. the leg lengths of the Hexa-WH can be obtained:

$$d_i = \|\mathbf{b}_{si} - \mathbf{a}_{si}\| = \|g_{st}(\mathbf{q}) \cdot \mathbf{b}_{ti} - g_{s5}(\mathbf{q}) \cdot \mathbf{a}_{5i}\|, \quad i=1,2,\cdots,6. \quad (15)$$

D. B. Nonlinear Calibration model Using POE

In practice, since the manufacturing and assembling errors are unavoidable, the actual leg length would have a joint offset error, the real location of the point A_i and B_i would never agree with the designed ones, and the twist of the serial mechanism would also have some deviations, the error model of the hybrid robot can be written as

$$d_i^p = d_i + \delta d_i = \left\| \mathbf{b}_{si}^p - \mathbf{a}_{si}^p \right\|, \tag{16}$$

where δd_i is leg joint offset, $\mathbf{b}_{si}^p = g_{st}^m(\mathbf{q})(\mathbf{b}_{ti} + \delta\mathbf{b}_{ti})$, and $g_{st}^m(\mathbf{q})$ can be obtained from the measured pose of the end-effector frame T with respect to base frame S, the predicted position vector of A_i can be expressed as:

$$\begin{aligned} \mathbf{a}_{si}^p &= \mathbf{a}_{si} + \delta\mathbf{a}_{si} = \mathbf{a}_{si} + \delta g_{s5}(\mathbf{q}) \cdot \mathbf{a}_{5i} + g_{s5}(\mathbf{q}) \cdot \delta\mathbf{a}_{5i} \\ &= \mathbf{a}_{si} + \left(\delta g_{s5}(\mathbf{q}) \cdot g_{s5}(\mathbf{q})^{-1}\right) \cdot \mathbf{a}_{si} + g_{s5}(\mathbf{q}) \cdot \delta\mathbf{a}_{5i} \end{aligned} \tag{17}$$

where the error matrix $\delta g_{s5}(\mathbf{q}) \cdot g_{s5}(\mathbf{q})^{-1}$ can be calculated according to the equation of (13).

According to the identifiability anylysis of He [8], the maximum number of the identifiable parameters in a serial robot with r revolute joints and t prismatic joints is $6r+3t+6$. Since we have 2 prismatic joints, 2 revolute joints and there is no pose measurement from the tip point of carriage, the independent identifiable parameters provided by serial carriage is equal to 18. Furthermore, each location of the spherical joint A_i and B_i provides 3 fixed coordinate error parameters, and each leg provides 1 leg joint offset error, thus the number of identification parameters from the Hexa-WH is 42.

IV. Parameter Identification Using Differential Evolution

Based on the calibration model (16), a nonlinear objective function can be formulated as the form of (19). The idea behind this nonlinear optimization method is to minimize the deviations between the measured and predicted values based on the Euclidean norm. The task of the parameter identification step is to search for a set of optimum solution

$$\mathbf{x} = (\delta\xi_1, \delta\xi_2, \delta\xi_3, \delta\xi_4, \delta d_i, \delta\mathbf{a}_i, \delta\mathbf{b}_i), \tag{18}$$

to minimize

$$f(\mathbf{x}) = \sum_{j=1}^{N}\sum_{i=1}^{6}(d_{i,j}^m - d_{i,j}^p)^2 \tag{19}$$

where $d_{i,j}^m$ and $d_{i,j}^p$ represent the i^{th} measured and predicted leg length in the j^{th} measurement configuration, N is the number of measurement configurations.

From the calibration model we can see it is a continuous, high nonlinear and multi-dimensional model which has a lot of local minima. To solve this problem, the global optimization methods have to be employed. Differential evolution (DE) has proven to be a promising candidate for minimizing real-valued, nonlinear, and multi-modal objective functions. It is a population-based optimization algorithm, and belongs to the class of evolutionary algorithms which utilizing mutation, crossover and selection operation to find out the optimum solutions.

In this work, the number of identification variables is equal to 60. The variables can be represented in DE as an individual vector $\mathbf{x}=(x_1, x_2, \cdots, x_{60})$. And the population for each generation G can be represented as a matrix $\mathbf{X}_{i,G} \in \Re^{60\times Np}$, where $i=1,2, \cdots, Np$ is the population index.

To establish a starting point for the optimization process, an initial population has to be created. Typically, the element of the population can be randomly generated within the feasible boundary of the variable parameters as

$$x_{j,i,G=0} = x_{j,i}^L + rand_j(0,1) \cdot (x_{j,i}^U - x_{j,i}^L) \tag{20}$$

where $j=1,2\cdots,60$ is the parameter index, i is the population index, and $x_{j,i}^L$, $x_{j,i}^U$ are the lower and upper bounds of the j^{th} parameters respectively.

After initialization, the evolution is based on the operations of mutation, crossover and selection. The main objective of mutation operation is to keep a population robust and search new territory. For each individual vector $\mathbf{x}_{i,G}$, a mutant vector $\mathbf{m}_{i,G+1}$ is generated according to

$$\mathbf{m}_{i,G+1} = \mathbf{x}_{r1,G} + F \cdot (\mathbf{x}_{r2,G} - \mathbf{x}_{r3,G}) \tag{21}$$

where the randomly selected integers $r_1,r_2,r_3 \in \{1,2\cdots,Np\}$, and $r_1 \neq r_2 \neq r_3 \neq j$. The mutation scale factor $F>0$.

To increase the diversity of the generated vectors, crossover operation is employed and the trail vector $\mathbf{u}_{i,G+1}=(x_{1,i,G+1}, x_{2,i,G+1}, \cdots, x_{60,i,G+1})$ is generated according to the formula:

$$u_{j,i,G+1} = \begin{cases} m_{j,i,G+1}, if(rand_j[0,1] \prec CR \cup j = j_r) \\ x_{j,i,G}, otherwise \end{cases} \tag{22}$$

where $G=1,2\cdots,G_{max}$ is the generation index, j_r is chosen randomly from the set $\{1,2\cdots,60\}$, which is used to ensure that the trail vector $\mathbf{u}_{i,G+1}$ gets at least one parameter from mutation vector $\mathbf{m}_{i,G+1}$. CR is a crossover rate constant which is a user defined parameter within the range [0, 1].

To determine if the trail vector can be selected as the member of the next generation, the trail vector $\mathbf{u}_{i,G+1}$ is compared to the target vector $\mathbf{x}_{i,G}$ by evaluating the objective function. The vector which has a smaller objective function value will be evolved to the next generation, i.e.

$$\mathbf{x}_{i,G+1} = \begin{cases} \mathbf{u}_{i,G+1}, \quad if \quad f(\mathbf{u}_{i,G+1}) \le f(\mathbf{x}_{i,G}) \\ \mathbf{x}_{i,G}, otherwise \end{cases} \tag{23}$$

Using this selection procedure, the convergence of the algorithm can be guaranteed and all individuals of the next generation will be as good as or better than individuals of the current population.

V. Simulation Analysis

In order to verify the validity and robustness of the calibration method, some computer simulations for the proposed robot are carried out in this section**.** The detailed nominal or designed parameter values are listed in the Table Ⅰ, Table Ⅱ and Table Ⅲ. The simulation procedures are as follows:

1) Randomly generate 100 end-effector measurement poses $g_{st}^{m}(\mathbf{q})$ within the robot workspace. Furthermore, we also generate 100 joint displacements of the carriage actuators, which can be regarded as the nominal joint displacements of the carriage actuators. In practice, the end-effector poses are obtained by the external measuring devices and the joint displacements are collected from the actuator sensor readings.
2) Assuming some preset errors in the twist of carriage (see Table Ⅱ), then they should meet the requirement of $\|\boldsymbol{\omega}_i+\delta\boldsymbol{\omega}_i\|=1$ and $(\boldsymbol{\omega}_i+\delta\boldsymbol{\omega}_i)^T(\mathbf{v}_i+\delta\mathbf{v}_i)=0$ for revolute joint and $\|\mathbf{v}_i+\delta\mathbf{v}_i\|=1$ for prismatic joint. The leg joint offset errors and the coordinate errors of spherical joint A_i and B_i are randomly generated within their tolerance range (see Table Ⅲ).
3) Based on the above nominal kinematic values, generated poses, carriage joint displacements and preset errors, we can calculate the actual leg lengths $d_{i,j}^{m}$ according to (16) and (17). In reality, the leg lengths can be obtained from the linear actuator sensor readings.
4) Take the 60 error parameters as the decision variables in the objective function (19) to calculate the predicted leg lengths $d_{i,j}^{p}$. Then the task of simulation is to employ DE algorithm to search an optimal combination of error parameters to minimize the value of the objective function within some program terminal conditions.

The DE control parameters can be selected according to the scheme of DE/rand-to-best/1 [14]. In our simulation, the DE control parameter *F=λ=0.75, CR=0.95, D=60, Np=600, N=100* and the error bound range [-0.5, 0.5].

To verify the effectiveness of the identification algorithm, we can assume the measurement device is perfect and the measurement errors and noise are omitted. After some evolution generations, a set of globally optimal solution can be achieved under the terminal conditions of maximum generations and convergence precisions. From Table Ⅱ and Table Ⅲ we can see that all of the preset variable values have been successfully recovered. If we simulate the program with different number of measurement poses, it can be seen from Fig. 2 that about 15 measurement poses are required to get a stabilized calibration results.

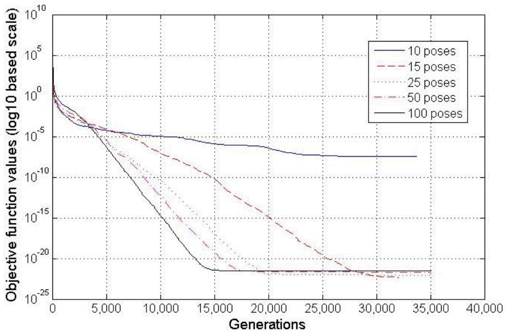


Fig. 2. The best objective function values with generations in different measurement poses.

To simulate the real application conditions and verify the robustness of the calibration algorithm, we also assume that the position and orientation of the end-effector will be measured with a laser tracker. The position and orientation measurement accuracy are in the range of [-0.01, 0.01] *mm* and [-0.00001, 0.0001] *rad.* respectively. The measurement noise may be regarded as a Gaussian distribution, with the ranges obeying the 3σ rule. Then the standard deviations of the position noise orientation noise are 0.003*mm* and 0.000003 *rad.* respectively. The simulation results in the Table Ⅱ and Table Ⅲ show that all of the identification parameters have successfully converged to the preset ones only with a slight difference due to the influence of measurement noises.

TABLE Ⅱ
NOMINAL AND IDENTIFIED PARAMETERS OF CARRIAGE (UNIT: MM)

	Nominal values ξ	Preset errors δξ	Identified without noise	Identified With noise
ω_{1x}	0	–	–	–
ω_{1y}	0	–	–	–
ω_{1z}	0	–	–	–
v_{1x}	1	cos(0.02) – 1	-1.99993×10^{-4}	-1.99418×10^{4}
v_{1y}	0	sin(0.02)	0.0199987	0.0199996
v_{1z}	0	0	1.32017×10^{-17}	-9.3368×10^{-7}
ω_{2x}	0	–	–	–
ω_{2y}	0	–	–	–
ω_{2z}	0	–	–	–
v_{2x}	0	0	-1.79319×10^{-15}	-7.5492×10^{-7}
v_{2y}	0	sin(0.02)	0.0199987	0.0199943
v_{2z}	1	cos(0.02) – 1	-1.99993×10^{-4}	1.99369×10^{-4}
ω_{3x}	0	0	2.39192×10^{-13}	6.7108×10^{-5}
ω_{3y}	1	cos(0.02) – 1	-1.99993×10^{-4}	3.63863×10^{-4}
ω_{3z}	0	sin(0.02)	0.0199987	0.0205498
v_{3x}	628	0.2	0.2	0.199999
v_{3y}	0	0	-9.2727×10^{-17}	1.08622×10^{-7}
v_{3z}	0	0	9.85174×10^{-17}	-4.4995×10^{-7}
ω_{4x}	1	cos(0.02) – 1	-1.99993×10^{-4}	4.29679×10^{-4}
ω_{4y}	0	sin(0.02)	0.0199987	0.0192154
ω_{4z}	0	0	-4.3303×10^{-15}	– 0.001064
v_{4x}	0	0	-2.8376×10^{-16}	-5.53304×10^{-7}
v_{4y}	– 376	0	2.79982×10^{-15}	-1.09392×10^{-6}
v_{4z}	313	0.2	0.2	0.200005

TABLE Ⅲ
NOMINAL AND IDENTIFIED PARAMETERS OF HYBRID ROBOT

	Nominal Values	Preset errors	Identified Without noise	Identified With noise
a_{51x}	231.6	-0.0654	-0.0654	-0.0596501
a_{51y}	-231.9	0.0687	0.0687	0.0693789
a_{51z}	0	0.0928	0.0928	0.0930406
a_{52x}	316.6	0.0448	0.0448	0.0479849
a_{52y}	-84.67	-0.0942	-0.0942	-0.0961375
a_{52z}	0	-0.0731	-0.0731	-0.0724037
a_{53x}	85	0.0229	0.0229	0.0192319
a_{53y}	316.58	0.0133	0.0133	0.0128637
a_{53z}	0	-0.0136	-0.0136	-0.0264649
a_{54x}	-85	-0.0752	-0.0752	-0.077656
a_{54y}	316.58	-0.0976	-0.0976	-0.100211
a_{54z}	0	0.0167	0.0167	0.0057957
a_{55x}	-316.6	0.0576	0.0576	0.0579317
a_{55y}	-84.67	-0.0486	-0.0486	-0.0487524
a_{55z}	0	0.0329	0.0329	0.0309625
a_{56x}	-231.6	-0.0117	-0.0117	-0.0118311
a_{56y}	-231.9	0.0676	0.0676	0.0727291
a_{56z}	0	0.0273	0.0273	0.02563
b_{t1x}	32.5	0.0581	0.0581	0.0636969
b_{t1y}	-125.9	-0.0648	-0.0648	-0.065743
b_{t1z}	0	0.0717	0.0717	0.0688051
b_{t2x}	125.3	0.0847	0.0847	0.0868672
b_{t2y}	34.8	-0.0478	-0.0478	-0.0497133
b_{t2z}	0	0.0324	0.0324	0.0325399
b_{t3x}	92.8	-0.0139	-0.0139	-0.0171407
b_{t3y}	91.1	-0.0266	-0.0266	-0.0277242
b_{t3z}	0	-0.0281	-0.0281	-0.0392927
b_{t4x}	-92.8	-0.0594	-0.0594	-0.0623811
b_{t4y}	91.1	0.0375	0.0375	0.0326712
b_{t4z}	0	0.0088	0.0088	0.000425282
b_{t5x}	-125.3	0.0228	0.0228	0.0222086
b_{t5y}	34.8	-0.0566	-0.0566	-0.0616722
b_{t5z}	0	-0.0368	-0.0368	-0.0416378
b_{t6x}	-32.5	-0.0638	-0.0638	-0.620393
b_{t6y}	-125.9	-0.0087	-0.0087	-0.00470459
b_{t6z}	231.6	-0.0736	-0.0736	-0.0752284
d_1	0	-0.3794	-0.3794	-0.382394
d_2	0	-0.0895	-0.0895	-0.0897172
d_3	0	0.165	0.165	0.16719
d_4	0	-0.3048	-0.3048	-0.302824
d_5	0	0.3233	0.3233	0.319244
d_6	0	0.0774	0.0774	0.0781733

VI. CONCLUSIONS

This paper presents a POE-based calibration method for a novel serial-parallel hybrid robot with redundant degrees of freedom. An identification model with 60 independent error parameters which has the ability to account for the geometric error sources originated from fabricating and assembling processes is derived. Using the DE algorithm, the 60 error parameters of the robot are successfully identified. It can be seen from the simulation results that the DE-based parameter identification method has a very strong stochastic searching ability, and it is reliable and can be easily used to identify error parameters in highly nonlinear kinematic models. The simulation results also verified the effectiveness and robustness of the proposed calibration method for serial-parallel hybrid robot.

ACKNOWLEDGMENT(S)

This work, supported by the European Union under the contract of association between EURATOM and Finnish Tekes, was carried out within the framework of the European Fusion Development Agreement.

REFERENCES

[1] J. Denavit, R. S. Hartenberg, "Kinematic modeling for robot calibration, " *Trans. ASME Journal of Applied Mechanics, Vol. 22,* June 1955, 215-221.

[2] S. Hayati, M. Mirmirani, "Improving the absolute positioning accuracy of robot manipulators, "Journal *of Robotic Systems, Vol. 2, No. 2,* 397-413.

[3] H. W. Stone, *Kinematic modeling, identification and control of robotic manipulator,* Ph.D. Thesis, Robotic Institute, Carnegie Mellon University.

[4] H. Zhuang, Z.S.Roth, and F.Hamano, "A complete and parametrically continuous kinematic model for robot manipulators," *IEEE Trans. Robot. & Autom.,* Vol. 8, No. 4, pp. 451-463, Aug. 1992.

[5] B. W. Mooring, Z. S. Roth, and M. Driels, *Fundamentals of Manipulator Calibration*. New York: John Wiley & Sons, 1991.

[6] K. Okamura and F. C. park, "Kinematic Calibration Using the Product of Exponentials Formula," *Robotica*, Vol. 14, pp. 415-421, 1996.

[7] I. -M. Chen, G. L. Yang, C. T. Tan, and S. H. Yeo, "Local POE Model for Robot Kinematic Calibration," Journal of *Mechanism and Machine Theory*, Vol. 36, pp. 1215-1239, May 2001.

[8] R. B. He, Y. J. Zhao, S. N. Yang, and S. Z. Yang, "Kinematic-parameter Identification for Serial-robot Calibration Based on POE Formula," *IEEE Trans. on Robotics*, Vol. 26, No. 3 pp. 411-423, June 2010.

[9] Y.G. Yang, Y.B.Liu, J. Pi, Y.S.Shi and W.Li, "Calibration of a 6-PRRS parallel manipulator using D-H method combined with vector chain," *The 2009 IEEE/RSJ international conference on Intelligent robots and systems,* Oct. 11-15, St. Louis, USA, pp. 3944-3948, 2009.

[10] G.Yang, I-Ming, Chen, W. K. Lim and S. H. Yeo, "Self-calibration of three-legged modular reconfigurable parallel robots based on leg-end distance errors," *Robotica,* Vol, 19, pp. 187-198, 2001.

[11] X. Zhong, J. Lewis and F. Nagy, "Inverse robot calibration using artificial neural networks," *International Journal of Engineering Applications of Artificial Intelligence, Vol. 9, No. 1,* pp. 83−93, 1996,.

[12] J. U. Dolinsky, I. D. Jenkinson and G.J. Colquhoun, "Application of genetic programming to the calibration of industrial robots, "*International Journal of Computers in Industry,* Vol. 58, pp. 255−264. 2007,

[13] Y. Liu, B. Liang, C. Li, L. Xue, S. Hu and Y. Jiang, "Calibration of a Steward parallel robot using genetic algorithm," in: *Proc. of the 2007 IEEE Inter. Conf. on Mech. and Autom.,* Harbin, China, Aug 5−8, pp. 2495−2500, 2007.

[14] R. Stone, K.Price, *Differential Evolution – a practical approach to global optimization.* Berlin: Springer, 2005.

[15] J. Liu, J. Lampinen, "A fuzzy adaptive differential evolution algorithm," *International Journal of Soft Computing – A Fusion of Foundations, Methodologies and Applications,* Vol. 9, No. 6, pp. 448−462,Jun 2005.

[16] H.P. Wu, H. Handroos, "Utilization of differential evolution in inverse kinematics solution of a parallel redundant manipulator," *Fourth international conference on knowledge-based intelligent engineering systems & allied technologies,* Brighton, UK, Aug. 30-Sept. 1, pp. 812-815, 2000 .

[17] R. M. Murray, Z. X. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Boca Ration, Florida: CRC Press, 1994.

ACTA UNIVERSITATIS LAPPEENRANTAENSIS

459. WINTER, SUSANNA. Network effects: scale development and implications for new product performance. 2011. Diss.

460. JÄÄSKELÄINEN, ANSSI. Integrating user experience into early phases of software development. 2011. Diss.

461. KÄÄRIÄINEN, TOMMI. Polymer surface modification by atomic layer deposition. 2011. Diss.

462. KOCHURA, ALEKSEY. Growth, magnetic and transport properties of InSb and II-IV-As2 semiconductors doped with manganese. 2011. Diss.

463. PUTKIRANTA, ANTERO. Possibilities and challenges of longitudinal studies in operations management. 2011. Diss.

464. HAPPONEN, ARI. Muuttuvaan kysyntään sopeutuva varastonohjausmalli. 2011. Diss.

465. VASAVA, PARITOSH. Application of computational fluid dynamics in modelling blood flow in human thoracic aorta. 2011. Diss.

466. PURO, LIISA. Identification of extractives and polysaccharides as foulants in membrane filtration of pulp and paper mill effluents. 2011. Diss.

467. LAPPALAINEN, PIA. Socially Competent Leadership – predictors, impacts and skilling in engineering. 2012. Diss.

468. PLAMTHOTTATHIL, ANSHY OONNITTAN. Application of electrokinetic Fenton process for the remediation of soil contaminated with HCB. 2012. Diss.

469. EBRAHIMI, FATEMEH. Synthesis of percarboxylic acids in microreactor. 2012. Diss.

470. JANTUNEN, SAMI. Making sense of software product requirements. 2012. Diss.

471. VILKO, JYRI. Approaches to supply chain risk management: identification, analysis and control. 2012. Diss.

472. TANSKANEN, VESA. CFD modelling of direct contact condensation in suppression pools by applying condensation models of separated flow. 2012. Diss.

473. HUHTANEN MIKKO. Software for design of experiments and response modelling of cake filtration applications. 2012. Diss.

474. PARJANEN, SATU. Creating possibilities for collective creativity
Brokerage functions in practice-based innovation. 2012. Diss.

475. KUKKONEN, SAKU. Generalized differential evolution for global multi-objective optimization with constraints. 2012. Diss.

476. LAAKSONEN, JONNA. Tactile-proprioceptive robotic grasping. 2012. Diss.

477. KALLIO, ANNE. Enhancing absorptive capacity in a non-research and development context
An action research approach to converting individual observations into organizational awareness. 2012. Diss.

478. LÄTTILÄ, LAURI. Improving transportation and warehousing efficiency with simulation based decision support systems. 2012. Diss.

479. OYOMNO, WERE. Usable privacy preservation in mobile electronic personality. 2012. Diss.

480. LINNALA, MIKKO. Simulation and optimization tools in paper machine concept design. 2012. Diss.

481. KORPIJÄRVI, JUHA. Aging based maintenance and reinvestment scheduling of electric distribution network. 2012. Diss.

482. KORHONEN, JUHAMATTI. Active inverter output filtering methods. 2012. Diss.

483. KLODOWSKI, ADAM. Flexible multibody approach in bone strain estimation during physical activity: quantifying osteogenic potential. 2012. Diss.

484. VUORENMAA, MARKKU. Osaamisen johtaminen pk-yrityksen kansainvälisen kasvun elinkaarella. 2012. Diss.

485. RAUTIAINEN, MARITA. Dynamic ownership in family business systems – a portfolio business approach. 2012. Diss.

486. LILIUS, REIJO. THE FINNISH IT INDUSTRIES IN TRANSITION Defining and measuring the Finnish software product and IT services industries by applying theoretical frameworks . 2012. Diss.

487. TUOMINEN, PASI. The purpose of consumer co-operation: implications for the management and governance of co-operatives. 2012. Diss.

488. SAARI, ESA. Suurnopeus-turbokoneroottoreiden termodynaaminen ja mekaaninen mallinnus sekä rakenneanalyysi. 2012. Diss.

489. PAANANEN, MIKKO. On innovative search: the use of internal and external sources of innovation among Finnish innovators. 2012. Diss.

490. BELOVA, POLINA. Quasiclassical approach to the vortex state in iron-based superconductors. 2012. Diss.

491. HIETANEN, IIRO. Design and characterization of large area position sensitive radiation detectors. 2012. Diss.

492. PÄSSILÄ, ANNE. A reflexive model of research-based theatre Processing innovation of the cross-road of theatre, reflection and practice-based innovation activities. 2012. Diss.

493. RIIPINEN, TOMI. Modeling and control of the power conversion unit in a solid oxide fuel cell environment. 2012. Diss.

494. RANTALAINEN, TUOMAS. Simulation of structural stress history based on dynamic analysis. 2012. Diss.

495. SALMIMIES, RIINA. Acidic dissolution of iron oxides and regeneration of a ceramic filter medium. 2012. Diss.

496. VAUTERIN, JOHANNA JULIA. The demand for global student talent: Capitalizing on the value of university-industry collaboration. 2012. Diss.

497. RILLA, MARKO. Design of salient pole PM synchronous machines for a vehicle traction application. 2012. Diss.

498. FEDOROVA, ELENA. Interdependence of emerging Eastern European stock markets. 2012. Diss.

499. SHAH, SRUJAL. Analysis and validation of space averaged drag model for numerical simulations of gas-solid flows in fluidized beds. 2012. Diss.