Summer 2019

Volumetric error compensation for industrial robots and machine tools

Le Ma

VOLUMETRIC ERROR COMPENSATION FOR INDUSTRIAL ROBOTS AND MACHINE TOOLS

by

LE MA

A DISSERTATION

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillments of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

MECHANICAL ENGINEERING

2019

Approved by:

Douglas A. Bristow, Co-advisor

Robert G. Landers, Co-advisor

Ming Leu

Xiaoping Du

V. Samaranayake

James B. Castle

PUBLICATION DISSERTATION OPTION

This dissertation consists of three articles that have been published or will be submitted for publications. Formatting changes have been made according to the dissertation formatting requirements of Missouri University of Science and Technology.

Paper I: pages 9 to 45 are intended for submission to the ASME Journal of Manufacturing Science and Engineering.

Paper II: pages 46 to 94 have been published in Robotics and Computer-Integrated Manufacturing.

Paper III: pages 95 to 140 are intended for submission to Robotics and Computer-Integrated Manufacturing.

ABSTRACT

A more efficient and increasingly popular volumetric error compensation method for machine tools is to compute compensation tables in axis space with tool tip volumetric measurements. However, machine tools have high-order geometric errors and some workspace is not reachable by measurement devices, the compensation method suffers a curve-fitting challenge, overfitting measurements in measured space and losing accuracy around and out of the measured space. Paper I presents a novel method that aims to uniformly interpolate and extrapolate the compensation tables throughout the entire workspace. By using a uniform constraint to bound the tool tip error slopes, an optimal model with consistent compensation capability is constructed. In addition to machine tools, industrial robots, are also becoming popularly used in manufacturing field. However, typical robot volumetric error compensation methods only consider constant errors such as link length and assembly errors while neglecting complicated kinematic errors such as strain wave gearing and out of rotating plane errors. Paper II presents a high-order joint-dependent model which describes both simple and complicated robot kinematic errors. A laser tracker with advantages of rapid data collection and a self-oriented position retroreflector are used for data collection. The experimental results show that nearly 20% of the robot kinematic errors are joint-dependent which are successfully captured by the proposed method. Paper III continues using the high-order joint-dependent robot error model while utilizing a new retroreflector with the ability of measuring robot position and orientation information simultaneously. More than 60% of measurement time is saved. Both position and orientation accuracy are also further improved.

ACKNOWLEDGEMENTS

I would like to express my deepest appreciation and gratitude to my advisors, Dr. Robert G. Landers and Dr. Douglas A. Bristow, who have provided tremendous mentoring and great guidance throughout my PhD studies. Their patience, support, expertise and passion have always encouraged me to dig more and realize my potential. I am forever appreciative for all the opportunities, platform and help they have given to me to become a better researcher.

I would like to gratefully acknowledge the financial support from the National Science Foundation (NSF, grant CMMI-1335340) and the Center for Aerospace Manufacturing Technologies at Missouri University of Science and Technology. I would also like to thank my dissertation committee, Dr. Ming Leu, Dr. Xiaoping Du, Dr. V. Samaranayake and Dr. James B. Castle, for their insightful comments. Thank my colleagues from the Precision Motion Control Lab.

Thank my friends in Rolla. Specifically, I would like to thank Dr. Muchen Sun, Dr. Xindi Sun and He Li. My time in Rolla could not have been easier and happier without them. It is my fortune to meet and be friends with them and I will treasure all the moments we have together for life.

Finally, thanks for the support from all my relatives and friends in China. Special thanks are due to my parents, without whose love I could not have achieved everything I have. Their integrity, diligence and affection have always been an invaluable spiritual motivation to me and kept inspiring me to become a better person.

TABLE OF CONTENTS

Page

LIST OF ILLUSTRATIONS

PAPER II

PAPER III

LIST OF TABLES

PAPER III

1. INTRODUCTION

Volumetric error compensation techniques have been well developed and increasingly adopted in industrial field, aiming to improve the absolute positioning accuracy of machine tools and industrial robots. While the compensation techniques are more mature and standardized for machine tools, the manufacturing of large monolithic parts keeps demanding for a more uniform and accurate compensation over a wider machine tool workspace. For industrial robot, its increasing applications in precision manufacturing also push the compensation process to become more accurate and efficient. This section serves to introduce basic volumetric error compensation techniques and challenges for machine tools and industrial robots. Improvements that this dissertation has made to this area will also be introduced.

1.1. VOLUMETRIC ERROR COMPENSATION FOR MACHINE TOOLS

One of the largest error source of machine tool volumetric inaccuracy comes from geometric errors, which include imprecise link length, offsets, imperfect assembly and wear of linkages. Those errors are typically corrected by mechanically adjusting the machine or through compensation. Based on previous research work, three-axis machine tools have 21 basic geometric errors [1] and five-axis machine tools have 41 basic geometric errors [2]. With the development of metrology equipment, techniques for identifying and compensating those errors have been well studied and developed. Typically, there are three steps to implement a compensation for machine tools: kinematic modeling, measurement of axis errors and error compensation [3]. In the second step, depending on

the way of using the metrology instruments to measure and identify the basic geometric errors, two types of compensation methods are classified, direct measurement methods and indirect measurement methods. In direct measurement methods, each single geometric error such as linear positioning error, straightness error and angular error of individual axes, is directly measured. The key point in the process of direct measurement methods is to set up the instrument (e.g., laser interferometer) appropriately such that the measured axis is isolated without the involvement of other axes' motion. While the details of direct measurement methods are well reviewed in [4, 5] and most of them are widely adopted by machine tool manufacturers [6], a complete compensation process is always very time-consuming. Unlike direct measurement methods, indirect measurement methods measure and analyze the tool tip volumetric inaccuracy which is treated as a contribution of all geometric errors. The advantage of indirect measurement methods is that all geometric errors can be identified simultaneously. Also, indirect measurement methods can build the kinematic error model in many different ways such as using the homogeneous transformation matrix (HTM) [1], using screw theory [7] and product of exponential models [8]. Further, a wider array of metrology instrument options is available for indirect measurement methods including ball bars [9, 10], R-test [11, 12], touch-trigger probes [13, 14], laser trackers [15, 16] and machining tests [17]. The details of indirect measurement methods are also well described in [3, 4]. Since there are more choices of error models and metrology instruments comparing with direct measurement methods, indirect measurements methods are more popularly studied in academic field.

While the volumetric error compensation techniques have been well developed in both industrial and academic fields, the trend of manufacturing large monolithic parts is

continuously giving challenges to this topic [18]. The demand for a larger compensation of machine tool workspace is thus arising. Laser tracker systems, which consists of a laser tracker and retroreflectors, with advantages of rapid data collection and ability to maximumly measure the machine tool workspace, are often utilized for measurement collection especially for large machine tools. Figure 1.1 shows a setup of a laser tracker and a retroreflector in a machine tool work cell.

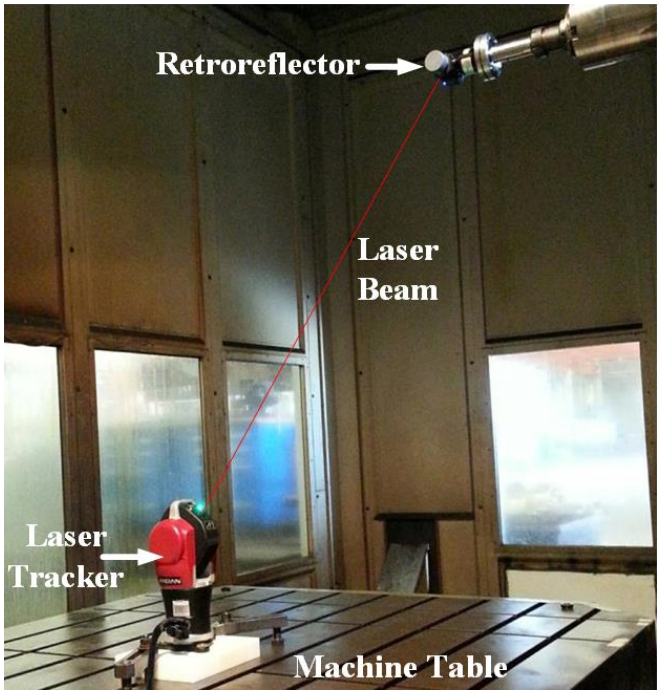

Figure 1.1. Setup of laser tracker and retroreflector in a machine tool work cell.

The laser tracker system enables a large measurement space within the machine tool workspace. However, some workspace is still not measurable due to the avoidance of physical contact and break of laser beam. So the error information from the unmeasured space cannot be identified with actual measurements. It is known that machine tool error

identification is basically a curve fitting process. For direct measurement methods, it is an explicit process as error is directly measured and fitted for each individual one. However, for indirect measurement methods, this process becomes implicit and obfuscated as all errors are identified simultaneously. Since some space does not have measurements, appropriate interpolation and extrapolation are needed to extend the identified error models to the entire machine tool workspace. Further, machine tools generally have high-order geometric errors. In [19], the angular errors are modeled with second-order polynomial functions for a three-axis machine tool. In [20], errors are fitted to third order polynomials as a function of axis position. In the two proposed models described in [18], error models with the sixth and eighth order polynomials give the best performance, respectively. Inappropriate interpolation and extrapolation of high-order polynomial will cause Runge's phenomenon [21] and lead to poor error description over the unmeasured workspace. The error model and thus the compensation accuracy will be inconsistent. Since a uniform compensation accuracy is needed especially for large monolithic part manufacturing, a method of constructing an optimally-fitted error model over the entire workspace is needed. Paper I in this dissertation gives such a method to address this problem. By investigating the relationship between single geometric errors and tool tip volumetric errors, and using a typical five-axis machine tool kinematic error model, an optimal method of interpolating and extrapolating kinematic error model for machine tools is proposed. Experimental results are also presented.

1.2. VOLUMETRIC ERROR COMPENSATION FOR INDUSTRIAL ROBOTS

Industrial robots have often been considered and used to build flexible automation platforms for many applications with its advantages of low cost and high flexibility [22]. While they are primarily applied for repeatable tasks such as palletizing and packaging, they are now increasingly used in many light machining tasks such as deburring and light drilling. The difference is that some tasks (e.g., palletizing and packaging) use the high repeatability property of robots while some other tasks (e.g., deburring and light machining) need the robot to have a high positioning accuracy. However, previous experiments have shown that robot accuracy can be an order of magnitude worse than its repeatability [23]. Thus, an effective method of compensating robot errors is needed.

The majority of robot inaccuracy comes from kinematic errors and research on this topic has been well studied for decades. As summarized in [24], there are three types of kinematic error compensation methods for robots, which are open-loop, closed-loop and screw axis measurement methods. Open-loop methods require an external metrology system to measure robot poses. All kinematic errors will be identified simultaneously by a nonlinear optimization with the measurements. This is similar as the indirect measurement methods for machine tools. Close-loop methods, in contrast to open-loop methods, do not require any external metrology instruments. Robot end effector will be attached to the ground to form a closed kinematic chain and error model parameters will be identified by reading joint angles. The third method, screw-axis measurement method, is to determine the actual kinematic relationship between consecutive axes. No complicated nonlinear optimization is needed.

Most kinematic error compensation methods focus on open-loop methods and screw measurement methods. As summarized in [25], a complete robot calibration process consists of four sequential steps, which are modeling, measurement, identification and compensation. This process is also similar as the calibration process for machine tools. Since robot does not have classified basic errors and robot kinematic errors are not directly measured, identification is regarded as an individual step in robot calibration. The first step, kinematic error modeling, is to mathematically describe the actual kinematic motion of robot with error parameters. Many model structures have been established by researchers including Denavit and Hartenberg (DH) model [26], Hayati model [27], S-model [28] and product of exponential model [29]. The second step, measurement, is to collect robot end effector position and orientation information at different robot poses. Different measurement devices have been used for data collection. These include Coordinate Measuring Machines (CMM) [30], telescoping ball bars [31], camera-based system [32, 33] and laser tracker systems [34, 35]. Those measurements will then be matched with the kinematic error model and the error parameters will be identified in the identification step. Different optimization algorithms such as least square [23], Levenberg-Marquardt [36] and maximum likelihood estimation [37] have been successfully used by researchers. The last step is to implement actual compensation with the identified error model. Unlike most machine tools controllers which offer a function of online compensation, most robot controllers can only do off-line programming, which is to add corrections to the nominal joint command to reach to the desired position and orientation. Since the inverse kinematics of the error model is super complicated to be solved analytically, numerical algorithms

have to be used. Inverse Jacobian method, a representative gradient search method, is often used to compute compensated joint commands [23].

Although a variety of modeling methods have been proposed, there are still limitations. First, link length error, assembly error and joint zero-reference offsets are summarized to be the main sources of kinematic errors [38]. However, those errors have not been appropriately described in one model. Also, a majority of the work only considers the simple kinematic errors such as link length and alignment errors which are often modeled as joint-independent. A representative method is Circle Point Analysis, which determines the offsets of DH and Hayati model parameters [39, 40]. Many joint-dependent errors such as strain wave gearing errors, in and out of joint rotating plane errors and backlash errors are often ignored, limiting the compensation accuracy. Those errors also play important roles in reducing robot accuracy. Thus, a generalized model, which considers both simple and complicated kinematic errors is needed. Paper II in this dissertation poses such a robot kinematic error model, describing both joint-independent and joint-dependent errors. By classifying and modeling different kinds of basic robot kinematic error sources, a generalized model is then proposed. A laser tracker system including a laser tracker and an Active Target is used to measure robot data. A maximum likelihood estimator and the inverse Jacobian method are used to identify modeled errors and compute compensated joint commands. Experimental results are presented, showing that 20% of robot kinematic errors are joint-dependent.

The proposed method in Paper II has not only been demonstrated in lab tests, but also achieved a great success in compensating robots for industrial collaborators. From 2013 to 2016, this method had been applied to compensate 6 robots from 5 companies

including Bell Helicopter Textron Inc., Toyota Bodine, Automated Precision Inc. (API), FANUC America and GE Power, Bangor. Before using the proposed calibration method, Bell Helicopter had to spend 2.5 weeks teaching a robot to debur a gear. After applying the calibration method and using off-line programming, only hours are needed to finish deburring a gear while satisfying the engineering specifications. Also, a calibration software package of the proposed method in Paper II is being commercialized with API to expand its contribution.

Paper III continues improving robot calibration accuracy and efficiency based on Paper II. In previous work with utilizing laser tracker for data collection, Spherical Mounted Retroreflectors (SMRs) or Active Targets (ATs) are attached to the robot end effector [34, 41]. However, they can only determine position information. To acquire robot orientation information, each robot pose has to be measured multiple times where the SMR or AT has to be placed at a different location on the end effector each time. This dramatically increases the measurement time. Extra fixturing and measurement errors are also introduced for each measurement set. A new device that can measure robot position and orientation information simultaneously is thus needed. Paper III in this dissertation presents a robot error compensation method with such a new device, which is called SmartTRACK Sensor (STS). A new kinematic error model is presented which considers the entire closed-loop measurement system including a laser tracker, robot and STS. Both deterministic errors (robot kinematic errors) and stochastic errors (robot repeatability, position and orientation measurement errors) are described in the proposed model. Experimental results are presented, showing that the error model accuracy is further improved while the measurement time is also reduced.

PAPER

I. INTERPOLATION AND EXTRAPOLATION OF OPTIMALLY-FITTED KINEMATIC ERROR MODEL FOR FIVE-AXIS MACHINE TOOLS

Le Ma, Douglas A. Bristow and Robert G. Landers

ABSTRACT

Machine tool geometric errors are frequently corrected by populating compensation tables that contain position-dependent offsets to each commanded axis position. While each offset can be determined by directly measuring the individual geometric error at that location, it is increasingly popular and potentially more efficient to compute the compensation using a volumetric error model derived from measurements across the entire axis space. Interpolation and extrapolation of measurements, once explicit in direct measurement methods, become implicit and obfuscated in the curve fitting process of volumetric error methods. The drive to maximize model accuracy while minimizing measurement sets can lead to significant model errors in portions of the workspace at or beyond the range of metrology equipment. In this paper, a novel method of constructing machine tool volumetric error models is presented in which interpolation and extrapolation errors are constrained. Using a typical five-axis machine tool compensation methodology, a constraint bounding the tool tip modeled error slope is added to the error model identification process. By including this constraint over the whole space, the geometric errors over the interpolation space are still well-identified. Also, the extrapolated model

performance is improved to be consistent with the behavior of the geometric error model over the interpolation space. The methodology is applied to an industrial five-axis machine tool. In the experimental implementation, for 25 measurements outside of the measured region, an unconstrained model increases the mean residual from 0.321 to 0.451 mm, while the constrained model reduces the mean residual to 0.191 mm, a 40.5% reduction.

1. INTRODUCTION

The accuracy of machine tools is critically important in many industrial applications. Generally, accuracy is achieved by using various metrology instruments to measure link lengths, offsets, and alignments that generate errors in the kinematic models, and then correct the errors by mechanically adjusting the machine, altering the kinematic model, or generating compensating position command algorithms [1–3]. While the kinematic offsets and alignments can be directly measured, it is increasingly common and efficient to use multiple measurements to indirectly identify the offsets and alignments as groups using various curve-fitting methods. Indirect measurement methods measure and analyze tool tip volumetric inaccuracy which is treated as a contribution of all geometric error sources. Using mathematical optimization algorithms, all errors can be identified simultaneously. For five-axis machine tools, typical metrology instruments for indirect methods include ball bars [4, 5], R-test [6, 7], touch-trigger probes [8, 9], laser trackers [10, 11] and machining tests [12]. The above indirect measurement methods are reviewed and discussed in [13].

Curve-fitting is an essential element in indirect measurement methods for kinematic compensation, although not a topic well addressed in the literature. In curve fitting, the finite measurements are extended across the entire working volume by the fitted curve, interpolating to regions surrounded by measurements and extrapolating to regions outside of the point cluster. While most curve fitting methods focus on the accuracy of the fitted curve at the measurement locations, accuracy in interpolated and extrapolated regions is highly dependent on the curve fitting method. In this paper a method of controlling the fitting process to ensure the fitted curves generate realistic solutions over the interpolation and extrapolation spaces is proposed.

Extrapolation is essential for machine tool compensation because geometric errors can only be identified over a limited range of the machine tool's workspace due to the design and size of the metrology instruments. For devices such as Ball bar, R-test, and touch-trigger probes, they have to maintain contact with the machine tool's spindle and table, limiting the space for measurement collection. For other metrology instruments such as laser interferometers and laser trackers, no contact between the spindle and the instrument is needed and, thus, measurements can be collected over a much larger volume. However, limitations still exist. Figures 1 and 2 give two examples. In Figure 1, the minimum commanded Z axis when machining is lower than the minimum commanded Z axis when collecting data. Thus, all of the compensation tables having Z axis values less than the minimum commanded Z axis when collecting data cannot be populated unless the geometric error functions are simply extrapolated. The minimum commanded Z axis can be decreased, as shown in Figure 2, if the machine tool spindle is attached to an A or B rotary axis. However, the cutting tool will be in a different orientation. Therefore, the

measured and unmeasured “spaces” for machine tools with more than three axes is more easily visualized in the joint space as opposed to the physical space. In addition to the collision avoidance, the measured region is also limited due to line of sight constraints when a laser tracker or an interferometer is used for data collection. Figure 3 shows the example from [14] that a five-axis machine tool measurements in some space are not collected due to collision avoidance and line of sight constraint. As shown in Figure 3, the top plot illustrates the consequence of Figure 2 and the bottom plot illustrates the consequence when the laser beam is blocked by the machine tool spindle.

The compensation table over the unmeasured “space” can be simply set to zero. In this case the geometric errors that exist at these points will be uncompensated, resulting in part geometry errors. The geometric error model can also be simply extrapolated by the interpolated curves. However, machine tool geometric error models are often described by polynomials. In [15], the angular errors are modeled with second-order polynomial functions for a three-axis machine tool. In [16], errors are fitted to third order polynomials as a function of axis position. In the two proposed models described in [14], error models with the sixth and eighth order polynomials give the best performance, respectively. Polynomial curve fitting possesses good interpolation characteristics. But it also has very poor extrapolation properties [17]. The poor extrapolation phenomenon is especially obvious for high-order polynomial curve fitting. Although polynomial curve fitting provides small error approximation to the data, the fitting accuracy deteriorates rapidly outside the range of the data [18] due to the increase of the magnitude of the fitted curve slope. In [19], the authors found that a larger slope is a main factor that lowers the extrapolation accuracy. An illustrative example is given in Figure 4 in which 13 simulated

measurements (with measurement errors) are fitted with a continuous modeled function. The slope of the modeled function changes rapidly and unrealistically when extrapolated, causing the inaccuracy to continually increase as the modeled function is further extrapolated. Therefore, a method of controlling the poor extrapolating behavior and thus improving extrapolated model accuracy is needed for machine tool geometric error models.

In addition to the unmeasured "space" limitation, the curve fitting also suffers an interpolation process since indirect measurement methods use continuous polynomials to fit discrete measurements. Low order polynomials will make the model less accurate on the collected data while high order polynomials are easy to overfit the measurements and thus lose accuracy. An example of using low and high order functions fitting 13 simulated measurements are shown in Figure 5. When a low order function is used, the fitting accuracy is not desired. When a high order function is used, the actual measurements can be fitted very well. However, for the space between the actual measurements, the fitted function has an oscillation which is known as Runge's phenomenon when high order function is used for interpolation [20]. Thus, the interpolation of machine tool geometric error models between actual measurements should also be carefully considered and treated.

This paper proposes a method to interpolate and extrapolate machine tool geometric error models throughout the entire machine tool workspace. Based on an error model proposed in [14], an analytical form of the tool tip modeled error slope is described using all of the single axis error slopes. By using a uniform constraint to bound the magnitude of the tool tip modeled error slope, all single axis errors derivatives and thus magnitudes will be constrained. A systematic methodology is given to determine the constraint value.

Compared with the unconstrained model, the proposed constrained model provides a more uniform error description over the entire machine tool workspace.

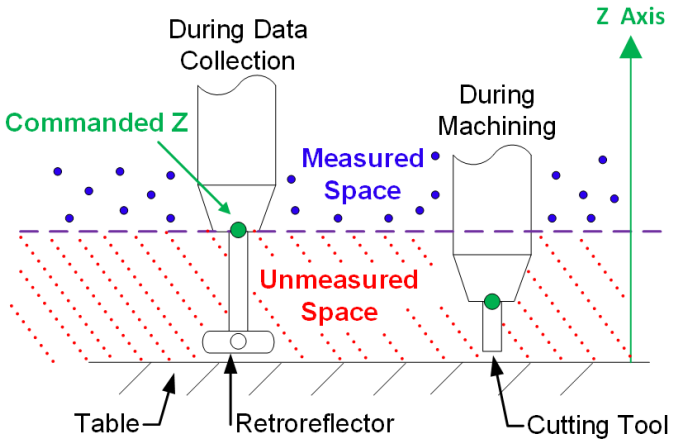

Figure 1. Illustration of Z axis unmeasured space for a three axis machine tool.

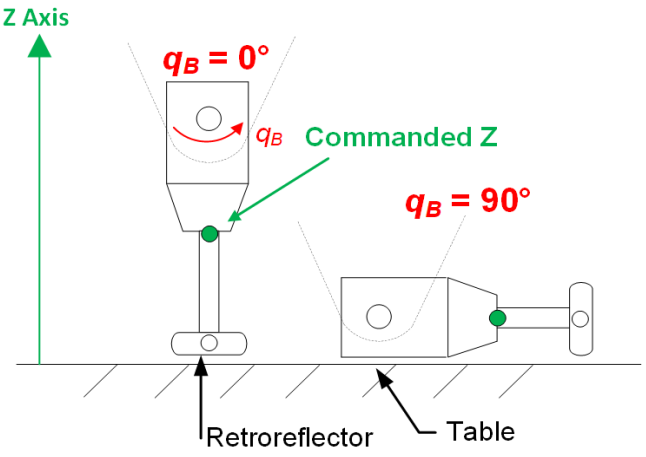

Figure 2. Illustration of coupling between B and Z Axes for a five axis machine tool.

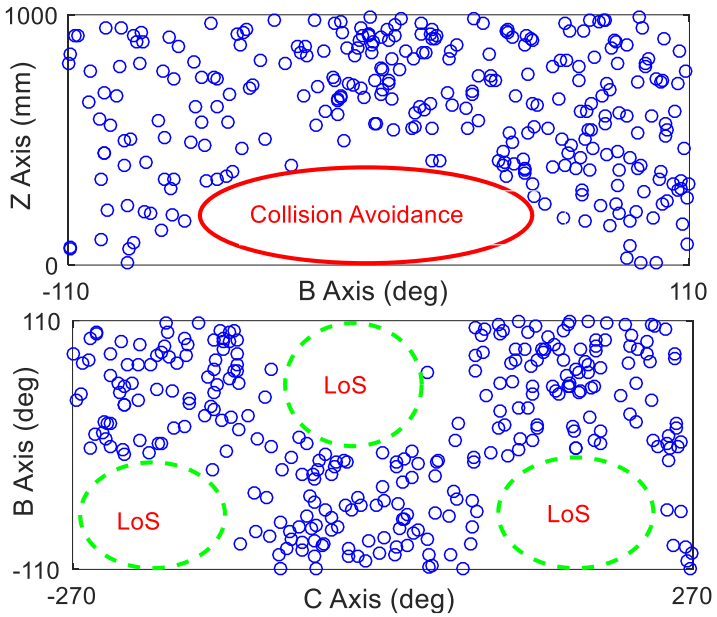

Figure 3. Illustration of collision avoidance and line of sight constraints.

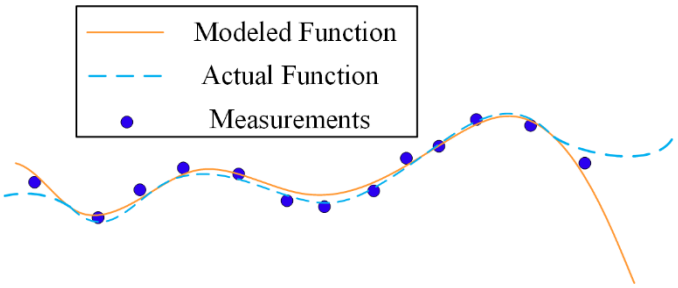

Figure 4. Function fitting with appropriate order.

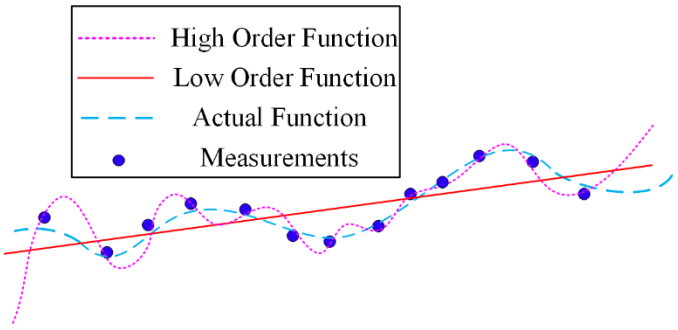

Figure 5. Low and high order functions fitting.

The rest of this paper is organized as follows. Section 2 gives the geometric error compensation methodology used in this paper. Section 3 proposes a constraint design used to construct constrained models. Section 4 gives the experimental results implemented on an industrial 5-axis machine tool, and a comparison between the unconstrained and constrained models is described and analyzed. The paper is summarized and conclusions are drawn in Section 5.

2. GEOMETRIC ERROR COMPENSATION BACKGROUND

While there are many methods to compensate machine tool geometric errors, this paper utilizes a table-based compensation methodology given in [14]. Most machine tool controllers offer a set of compensation tables that map a single axis commands to small corrections to a single axis positions in real time. This section describes the model used for table-based compensation which is named as axis perturbation model. The description of actual measurements and the identification of model parameters are also given.

2.1. GEOMETRIC ERROR MODEL CONSTRUCTION

Using the zero reference model in [21] to describe the Linear Homogeneous Transformations (LHTs) [22] between the coordinate systems of two axes, the nominal kinematics of an *n*-axis machine tool is

$$\mathbf{F}_{nom}(\mathbf{q}) = \mathbf{T}_1(q_1)\mathbf{T}_2(q_2)\ldots\mathbf{T}_n(q_n), \tag{1}$$

where $\mathbf{F}_{nom}$ describes the nominal orientation and position of the machine tool's last frame with respect to the machine tool base frame, $\mathbf{q} = [q_1\ q_2\ \ldots\ q_n]^T$ is the nominal axis command

vector, and $\mathbf{T}_i$ is the nominal LHT from the $(i-1)^{th}$ axis coordinate system to the i^{th} axis coordinate system. The compensation tables offered by the machine tool controller are look-up tables that depend on the input nominal axis commands and contain a small adjustment for the nominal axis commands [14]. The axis perturbation model is such a kind of kinematic error model that can be used to efficiently generate the compensation tables. The axis perturbation kinematic model is given by,

$$\mathbf{F}_{AP}(\mathbf{q}) = \mathbf{F}_{nom}(\mathbf{q} + \boldsymbol{\delta}\mathbf{q}(\mathbf{q})), \tag{2}$$

where $\mathbf{F}_{AP}$ describes the uncompensated machine and $\boldsymbol{\delta}\mathbf{q}(\mathbf{q}) = [\delta q_1(\mathbf{q})\ \delta q_2(\mathbf{q}) \ldots\ \delta q_n(\mathbf{q})]^T$ is a perturbation in the command vector with δq_i as the error for command q_i. Compensation tables can be trivially generated as $-\boldsymbol{\delta}\mathbf{q}(\mathbf{q})$ such that the machine recovers nominal kinematics using the compensated input $\hat{\mathbf{q}} = \mathbf{q} - \boldsymbol{\delta}\mathbf{q}(\mathbf{q})$, or $\mathbf{F}_{AP}(\hat{\mathbf{q}}) = \mathbf{F}_{nom}(\mathbf{q})$.

Based on the structure of common machine tool compensation tables, each correction is described as the sum of n perturbation functions,

$$\delta q_i(\mathbf{q}) = f_{i1}(q_1) + f_{i2}(q_2) + \cdots + f_{in}(q_n), \tag{3}$$

where $f_{ij}(q_j)$ is a table function (if it exists) that maps the error of axis j onto the correction for the command position of axis i. To capture both constant and complex errors, each function is mathematically described with a set of sufficient order polynomials. Here, a basis set of functions called Chebyshev polynomials given on a normalized scale are used. Given a parameter l in the interval [-1 1], a Chebyshev polynomial has the form,

$$C(\lambda) = a_0 g_0(\lambda) + a_1 g_1(\lambda) + a_2 g_2(\lambda) + \cdots + a_m g_m(\lambda), \tag{4}$$

where

$$g_0(\lambda)=1,\ g_1(\lambda)=\lambda,\ g_2(\lambda)=2\lambda^2-1,\ldots,\ g_{m+1}(\lambda)=2\lambda g_m(\lambda)-g_{m-1}(\lambda), \tag{5}$$

m is the Chebyshev polynomial order and a_0, a_1, a_2, ..., a_m are the polynomial coefficients. Thus, each perturbation function can be represented by m^{th} order Chebyshev polynomials as

$$f_{ij}(\bar{q}_j)=a_{0,ij}+a_{1,ij}g_1(\bar{q}_j)+a_{2,ij}g_2(\bar{q}_j)+\cdots+a_{m,ij}g_m(\bar{q}_j), \tag{6}$$

where

$$\bar{q}_j=\frac{2\left(q_j-q_{j,\min}\right)}{\left(q_{j,\max}-q_{j,\min}\right)}-1, \tag{7}$$

is the j^{th} linearly mapped axis command by scaling the axis range to the interval [-1 1] and $q_{j,min}$ and $q_{j,max}$ are the minimum and maximum j^{th} axis commands, respectively.

2.2. MEASUREMENT

A laser tracker coupled with an active retroreflector attached to the machine tool spindle is used to measure machine position. To describe the measurements, a complete closed kinematic loop between the laser tracker and retroflector is needed. Considering the measurement errors and the potential axis positioning errors, the measured machine position can be described as,

$$\mathbf{p}_a(\mathbf{q})=\mathbf{T}_{mf}\mathbf{E}_{mf}\mathbf{F}_{AP}(\mathbf{q}+\boldsymbol{\nu})\mathbf{p}_{tl}+\boldsymbol{\xi}, \tag{8}$$

where $\mathbf{T}_{mf}$ is the nominal transformation from the laser tracker frame to the machine tool's base frame, $\mathbf{E}_{mf}$ is the correction of $\mathbf{T}_{mf}$, $\boldsymbol{\nu}$ is a stochastic axis positioning error vector, $\mathbf{p}_{tl}$ is a tool length vector from the machine to the retroreflector, and $\boldsymbol{\xi}$ is a measurement error vector. Figure 6 gives a schematic description of the actual measurement with respect to

the laser tracker frame. The base frame correction, $\mathbf{E}_{mf}$, is described with a fixed six degree of freedom error matrix,

$$\mathbf{E}_{mf} = \begin{bmatrix} 1 & -\varepsilon_{Z,0} & \varepsilon_{Y,0} & \delta_{X,0} \\ \varepsilon_{Z,0} & 1 & -\varepsilon_{X,0} & \delta_{Y,0} \\ -\varepsilon_{Y,0} & \varepsilon_{X,0} & 1 & \delta_{Z,0} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{9}$$

where $\varepsilon_{X,0}$, $\varepsilon_{Y,0}$, $\varepsilon_{Z,0}$ are small rotations and $\delta_{X,0}$, $\delta_{Y,0}$, $\delta_{Z,0}$ are small translations about the X, Y and Z axis. The fixed tool length vector, $\mathbf{p}_{tl}$, is

$$\mathbf{p}_{tl} = \begin{bmatrix} 0 & 0 & l_t + \delta_t & 1 \end{bmatrix}^T, \tag{10}$$

where l_t is the measured tool length between the origin of the machine tool's last frame and the tool tip and δ_t is the correction of l_t. The retroreflector can only determine 3-D positional information. To compensate for orientation error, two sets of measurements are taken, each time with the retroreflector mounted on a tool with a different length. Each group of two measurements uses the same axis commands and, thus, lie on the same spindle axis orientation.

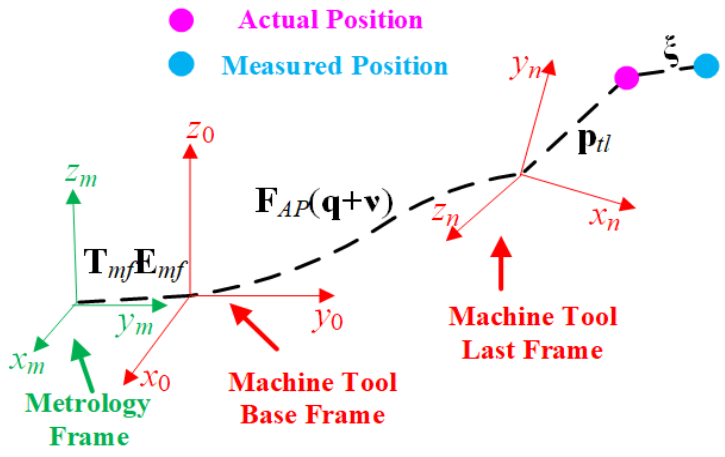

Figure 6. Measurement model schematic showing frames.

2.3. MODEL IDENTIFICATION

An optimization algorithm based on the implicit loop method [23] is used to determine the error model parameters. An advantage of this method is that axis positioning errors and measurement errors can be directly incorporated. They are assumed to be independent and follow normal distributions. These errors are identified in the optimization process and, thus, are separate from the machine tool's geometric errors.

Let $\mathbf{\Sigma}_{\mathbf{v}}$ and $\mathbf{\Sigma}_{\xi}$, respectively, be the covariance matrices for the axis positioning and measurement errors, respectively. Considering N measured machine tool poses, the kinematic errors are identified by minimizing the following function [24],

$$\left[\mathbf{v}_1^*, \ldots, \mathbf{v}_N^*, \mathbf{b}^*\right] = \min_{\mathbf{v}_1,\ldots,\mathbf{v}_N,\mathbf{b}} \sum_{k=1}^{N} \left(\mathbf{v}_k^T \mathbf{\Sigma}_{\mathbf{v}}^{-1} \mathbf{v}_k + \xi_k^T \mathbf{\Sigma}_{\xi}^{-1} \xi_k \right), \tag{11}$$

subject to the implicit loop constraint,

$$\xi_k = \begin{bmatrix} \xi_{k,1} \\ \xi_{k,2} \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{a,k,1}(\mathbf{q}_k) - \mathbf{T}_{mf}\mathbf{E}_{mf}(\mathbf{b})\mathbf{F}_{nom}\left(\mathbf{q}_k + \delta\mathbf{q}_k(\mathbf{b},\mathbf{q}_k) + \mathbf{v}_{k,1}\right)\mathbf{p}_{tl,1}(\mathbf{b}) \\ \mathbf{p}_{a,k,2}(\mathbf{q}_k) - \mathbf{T}_{mf}\mathbf{E}_{mf}(\mathbf{b})\mathbf{F}_{nom}\left(\mathbf{q}_k + \delta\mathbf{q}_k(\mathbf{b},\mathbf{q}_k) + \mathbf{v}_{k,2}\right)\mathbf{p}_{tl,2}(\mathbf{b}) \end{bmatrix}^T, \tag{12}$$

where $\xi_k = \begin{bmatrix} \xi_{k,1}^T & \xi_{k,2}^T \end{bmatrix}^T$ is the measurement error vector and $\mathbf{v}_k = \begin{bmatrix} \mathbf{v}_{k,1}^T & \mathbf{v}_{k,2}^T \end{bmatrix}^T$ is the axis positioning error vector for the k^{th} pose measured by the retroreflector mounted on two tools with different lengths, $\mathbf{b}$ is the error parameter vector including the polynomial coefficients in the modeled machine tool kinematics, six static errors in $\mathbf{E}_{mf}$, and the tool length errors $\delta_{l,1}$ and $\delta_{l,2}$ corresponding to the two tool length vectors $\mathbf{p}_{tl,1}$ and $\mathbf{p}_{tl,2}$, respectively.

3. CONSTRAINED GEOMETRIC ERROR MODEL

As previously introduced, larger slopes are a main factor that reduces extrapolation accuracy. In Figure 4, when extrapolating the modeled function, the magnitude of the modeled function slope becomes larger and the model accuracy becomes poor. In addition, in Figure 5, when a high order modeled function is used, the magnitude of the modeled function slope is also larger than the actual function and the model accuracy is also poor. In [19], the authors found that lowering slopes will decrease the magnitude of the extrapolated function and improve extrapolation reliability. Thus, while using high order model functions will improve the description for the collected measurements, a constraint on the modeled function slope is needed to guarantee the accuracy of the model function over the entire workspace. In this section, a constraint that aims to control all perturbation function slopes over the whole workspace will be developed. A procedure to utilize this constraint for geometric error model construction will also be given.

3.1. CONSTRAINT DESIGN

Let the nominal and modeled transformations from the machine tool's base frame to the machine tool's last frame, respectively, be

$$\mathbf{F}_{nom} = \begin{bmatrix} \mathbf{R}_{nom} & \mathbf{p}_{nom} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{13}$$

$$\mathbf{F}_{AP} = \begin{bmatrix} \mathbf{R}_{AP} & \mathbf{p}_{AP} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{14}$$

where $\mathbf{R}_{nom}$ and $\mathbf{R}_{AP}$ are the nominal and modeled rotations from the machine tool's base frame to the machine tool's last frame, respectively, $\mathbf{p}_{nom}$ and $\mathbf{p}_{AP}$ are nominal and modeled positions of the machine tool's last frame with respect to the machine tool's base frame, respectively. Let the measured tool length vector be

$$\bar{\mathbf{p}}_{tl} = \begin{bmatrix} 0 & 0 & l_t \end{bmatrix}^T . \tag{15}$$

To connect the rotation and position information in one representation, the nominal and modeled tool tip positions with respect to the machine tool's base frame used,

$$\mathbf{p}_{t,nom} = \mathbf{R}_{nom}\bar{\mathbf{p}}_{tl} + \mathbf{p}_{nom} , \tag{16}$$

$$\mathbf{p}_{t,AP} = \mathbf{R}_{AP}\bar{\mathbf{p}}_{tl} + \mathbf{p}_{AP} . \tag{17}$$

For a five-axis machine tool with axis sequence XYZCB, taking the gradient of (16) and (17) with respect to the axis command vector $\mathbf{q}$, respectively,

$$\nabla_{\mathbf{q}}\mathbf{p}_{t,nom} = \begin{bmatrix} \dfrac{\partial \mathbf{p}_{t,nom}}{\partial q_X} & \dfrac{\partial \mathbf{p}_{t,nom}}{\partial q_Y} & \dfrac{\partial \mathbf{p}_{t,nom}}{\partial q_Z} & \dfrac{\partial \mathbf{p}_{t,nom}}{\partial q_C} & \dfrac{\partial \mathbf{p}_{t,nom}}{\partial q_B} \end{bmatrix}, \tag{18}$$

$$\nabla_{\mathbf{q}}\mathbf{p}_{t,AP} = \begin{bmatrix} \dfrac{\partial \mathbf{p}_{t,AP}}{\partial q_X} & \dfrac{\partial \mathbf{p}_{t,AP}}{\partial q_Y} & \dfrac{\partial \mathbf{p}_{t,AP}}{\partial q_Z} & \dfrac{\partial \mathbf{p}_{t,AP}}{\partial q_C} & \dfrac{\partial \mathbf{p}_{t,AP}}{\partial q_B} \end{bmatrix}, \tag{19}$$

For the i^{th} axis,

$$\frac{\partial \mathbf{p}_{t,nom}}{\partial q_i} = \frac{\partial \mathbf{R}_{nom}\bar{\mathbf{p}}_{tl}}{\partial q_i} + \frac{\partial \mathbf{p}_{nom}}{\partial q_i} = \mathbf{S}\left(\mathbf{J}_{\boldsymbol{\omega},nom,i}\right)\mathbf{R}_{nom}\bar{\mathbf{p}}_{tl} + \mathbf{J}_{\mathbf{v},nom,i} , \tag{20}$$

$$\frac{\partial \mathbf{p}_{t,AP}}{\partial q_i} = \frac{\partial \mathbf{R}_{AP}\bar{\mathbf{p}}_{tl}}{\partial q_i} + \frac{\partial \mathbf{p}_{AP}}{\partial q_i} = \mathbf{S}\left(\mathbf{J}_{\boldsymbol{\omega},AP,i}\right)\mathbf{R}_{AP}\bar{\mathbf{p}}_{tl} + \mathbf{J}_{\mathbf{v},AP,i} , \tag{21}$$

where $\mathbf{S}(\bullet)$ is a skew symmetric matrix operator, $\mathbf{J}_{\boldsymbol{\omega},nom,i}$ and $\mathbf{J}_{\boldsymbol{\omega},AP,i}$ are the i^{th} columns of the angular velocity portions of the nominal and modeled Jacobian matrices, respectively, $\mathbf{J}_{\mathbf{v},nom,i}$ and $\mathbf{J}_{\mathbf{v},AP,i}$ are the i^{th} columns of the linear velocity portions of the nominal and

modeled Jacobian matrices, respectively. The machine tool Jacobian matrix is a matrix relates the small changes between the axes positions in joint coordinates and the end effector positions in Cartesian coordinates [25]. For a vector **a** = [a_1,a_2,a_3], the skew symmetric matrix operator of **a** is,

$$\mathbf{S}(\mathbf{a}) = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}. \tag{22}$$

To combine the nominal and model information, the difference between (20) and (21) is taken which gives the slope of the tool tip modeled error, $\mathbf{e}_t$, with respect to the i^{th} axis,

$$\begin{aligned} \frac{\partial \mathbf{e}_t}{\partial q_i} &= \frac{\partial\left(\mathbf{p}_{t,nom} - \mathbf{p}_{t,AP}\right)}{\partial q_i} \\ &= \mathbf{S}\left(\mathbf{J}_{\boldsymbol{\omega},nom,i}\right)\mathbf{R}_{nom}\bar{\mathbf{p}}_{tl} + \mathbf{J}_{\mathbf{v},nom,i} - \mathbf{S}\left(\mathbf{J}_{\boldsymbol{\omega},AP,i}\right)\mathbf{R}_{AP}\bar{\mathbf{p}}_{tl} - \mathbf{J}_{\mathbf{v},AP,i} \\ &= \left(\mathbf{S}\left(\mathbf{J}_{\boldsymbol{\omega},nom,i}\right)\mathbf{R}_{nom} - \mathbf{S}\left(\mathbf{J}_{\boldsymbol{\omega},AP,i}\right)\mathbf{R}_{AP}\right)\bar{\mathbf{p}}_{tl} + \left(\mathbf{J}_{\mathbf{v},nom,i} - \mathbf{J}_{\mathbf{v},AP,i}\right) \end{aligned} . \tag{23}$$

Assuming the difference between $\mathbf{R}_{nom}$ and $\mathbf{R}_{AP}$ is small enough, such that,

$$\mathbf{R}_{nom} \approx \mathbf{R}_{AP}, \tag{24}$$

then (23) can be written as

$$\begin{aligned} \frac{\partial \mathbf{e}_t}{\partial q_i} &= \left(\mathbf{S}\left(\mathbf{J}_{\boldsymbol{\omega},nom,i}\right)\mathbf{R}_{nom} - \mathbf{S}\left(\mathbf{J}_{\boldsymbol{\omega},AP,i}\right)\mathbf{R}_{nom}\right)\bar{\mathbf{p}}_{tl} + \left(\mathbf{J}_{\mathbf{v},nom,i} - \mathbf{J}_{\mathbf{v},AP,i}\right) \\ &= \left(\mathbf{S}\left(\mathbf{J}_{\boldsymbol{\omega},nom,i} - \mathbf{J}_{\boldsymbol{\omega},AP,i}\right)\right)\mathbf{R}_{nom}\bar{\mathbf{p}}_{tl} + \left(\mathbf{J}_{\mathbf{v},nom,i} - \mathbf{J}_{\mathbf{v},AP,i}\right) \\ &= \mathbf{S}\left(\Delta\mathbf{J}_{\boldsymbol{\omega},i}\right)\mathbf{R}_{nom}\bar{\mathbf{p}}_{tl} + \Delta\mathbf{J}_{\mathbf{v},i} \end{aligned} , \tag{25}$$

where $\Delta\mathbf{J}_{\boldsymbol{\omega},i}$ is the difference between the i^{th} columns of the angular velocity portions of the nominal and modeled Jacobian matrices, $\Delta\mathbf{J}_{\mathbf{v},i}$ is the difference between the i^{th} columns of the linear velocity portions of the nominal and modeled Jacobian matrices.

3.2. JACOBIAN MATRIX

To save space, the detailed derivation of $\mathbf{J}_{\boldsymbol{\omega},AP}$, $\mathbf{J}_{\boldsymbol{\omega},nom}$, $\mathbf{J}_{\mathbf{v},AP}$ and $\mathbf{J}_{\mathbf{v},nom}$ are shown in the appendix. Using the derived formulations, the differences in the linear and angular Jacobian matrices, respectively, are

$$\mathbf{J}_{\mathbf{v},AP} = \begin{bmatrix} 1+\dfrac{df_{XX}(q_X)}{dq_X} & \dfrac{df_{XY}(q_Y)}{dq_Y} & \dfrac{df_{XZ}(q_Z)}{dq_Z} & \dfrac{df_{XC}(q_C)}{dq_C} & \dfrac{df_{XB}(q_B)}{dq_B} \\ \dfrac{df_{YX}(q_X)}{dq_X} & 1+\dfrac{df_{YY}(q_Y)}{dq_Y} & \dfrac{df_{YZ}(q_Z)}{dq_Z} & \dfrac{df_{YC}(q_C)}{dq_C} & \dfrac{df_{YB}(q_B)}{dq_B} \\ \dfrac{df_{ZX}(q_X)}{dq_X} & \dfrac{df_{ZY}(q_Y)}{dq_Y} & 1+\dfrac{df_{ZZ}(q_Z)}{dq_Z} & \dfrac{df_{ZC}(q_C)}{dq_C} & \dfrac{df_{ZB}(q_B)}{dq_B} \end{bmatrix}, \quad (26)$$

$$\mathbf{J}_{\boldsymbol{\omega},AP} = \mathbf{A}\begin{bmatrix} \dfrac{df_{BX}(q_X)}{dq_X} & \dfrac{df_{BY}(q_Y)}{dq_Y} & \dfrac{df_{BZ}(q_Z)}{dq_Z} & \dfrac{df_{BC}(q_C)}{dq_C} & 1+\dfrac{df_{BB}(q_B)}{dq_B} \\ \dfrac{df_{BX}(q_X)}{dq_X} & \dfrac{df_{BY}(q_Y)}{dq_Y} & \dfrac{df_{BZ}(q_Z)}{dq_Z} & \dfrac{df_{BC}(q_C)}{dq_C} & 1+\dfrac{df_{BB}(q_B)}{dq_B} \\ \dfrac{df_{CX}(q_X)}{dq_X} & \dfrac{df_{CY}(q_Y)}{dq_Y} & \dfrac{df_{CZ}(q_Z)}{dq_Z} & 1+\dfrac{df_{CC}(q_C)}{dq_C} & \dfrac{df_{CB}(q_B)}{dq_B} \end{bmatrix}, \quad (27)$$
$$-\begin{bmatrix} 0 & 0 & 0 & 0 & -\sin(q_C) \\ 0 & 0 & 0 & 0 & \cos(q_C) \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

where

$$\mathbf{A} = \begin{bmatrix} -\sin\left(q_C+\delta q_C(\mathbf{q})\right) & 0 & 0 \\ 0 & \cos\left(q_C+\delta q_C(\mathbf{q})\right) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (28)$$

3.3. CONSTRAINED MODEL CONSTRUCTION

As shown in (26) and (27), the linear and angular Jacobian differences depend on the slope of each of the perturbation functions. Let (25) be rewritten as

$$\mathbf{h}_i = \frac{\partial \mathbf{e}_t}{\partial q_i} = \mathbf{S}\left(\Delta \mathbf{J}_{\boldsymbol{\omega},i}\right)\mathbf{R}_{nom}\bar{\mathbf{p}}_{tl} + \Delta \mathbf{J}_{\mathbf{v},i}\,, \tag{29}$$

which represents the tool tip modeled error change per axis unit. Note that the unit of $\mathbf{h}_i$ is mm/mm for translational axes and mm/deg for rotational axes. A unification of the unit is needed. A unification of the unit is made,

$$\begin{aligned}\bar{\mathbf{h}}_i &= \frac{\partial \mathbf{e}_t}{\partial \bar{q}_i} = \frac{\partial \mathbf{e}_t}{\partial q_i}\frac{\left(q_{i,\max} - q_{i,\min}\right)}{2} \\ &= \mathbf{h}_i \frac{\left(q_{i,\max} - q_{i,\min}\right)}{2} \\ &= \left(\mathbf{S}\left(\Delta \mathbf{J}_{\boldsymbol{\omega},i}\right)\mathbf{R}_{nom}\bar{\mathbf{p}}_{tl} + \Delta \mathbf{J}_{\mathbf{v},i}\right)\frac{\left(q_{i,\max} - q_{i,\min}\right)}{2}\end{aligned}\,, \tag{30}$$

which represents the tool tip modeled error change per half range motion for the i^{th} axis. By using (30) as a constraint during model identification, the magnitude of the all perturbation function slopes will be regulated. Previously, (11) and (12) are used for model identification without any other constraint. To construct constrained geometric error models and control the error function slopes, a new constraint is added to the identification process in addition to (11) and (12) which is designed as

$$\left\|\bar{\mathbf{h}}_i\right\|_2 \le c, \quad \left(i = X, Y, Z, C, B\right), \tag{31}$$

where c is the constraint applied to all machine tool axes throughout the entire space.

3.4. CONSTRAINT VALUE DETERMINATION

To construct a geometric error model with consistent behavior over the entire space, one must carefully determine the constraint value, i.e., the value c used in (31). A large c will fail to constrain the unrealistic model behavior while a small c will over constrain the error slopes, reducing the model accuracy over the whole space. Since boundary space is

an intermediate space between interpolation and extrapolation spaces where the perturbation functions are fitted with few measurements, the identification of function slopes is more sensitive in boundary space. Thus, a subset of data from boundary space is used to determine an appropriate constraint value, *c*, which will give a balance between unconstrained and over-constrained models. To locate the boundary space points in a two-axis space, a database technique named BORDER is used here [26]. Typically, for a point cluster, BORDER uses three steps to determine the boundary space point. The first step is to find the *k*-nearest neighbors (*k*NN) for each point in the data set where *k* is a user defined and tuned integer. The second step is to count the number of reverse *k*-nearest neighbors (R*k*NN) and the last step is to sort the points according to the R*k*NN number. As boundary space points tend to have fewer R*k*NN, user can choose any number of boundary points from the sorted sequence.

An example is given in the following to explain how BORDER works. Figure 7 gives a cluster of 8 points that are labeled with point number. The integer, *k*, is an arbitrarily defined value. Use $k = 2$ for this example and the 2-nearest neighbors for each point are given in Table 1. In the second column of Table 1, $\mathbf{p}_2$ is the 2-nearest neighbor of $\mathbf{p}_1$, $\mathbf{p}_3$ and $\mathbf{p}_4$. Thus, $\mathbf{p}_1$, $\mathbf{p}_3$ and $\mathbf{p}_4$ are the reverse 2-nearest neighbors of $\mathbf{p}_2$. Table 2 lists the reverse 2-nearest neighbors for each point. As $\mathbf{p}_1$, $\mathbf{p}_4$ and $\mathbf{p}_8$ have the fewest number of reverse 2-nearest neighbors, they are identified as the boundary points which are consistent with the visual observation of Figure 7.

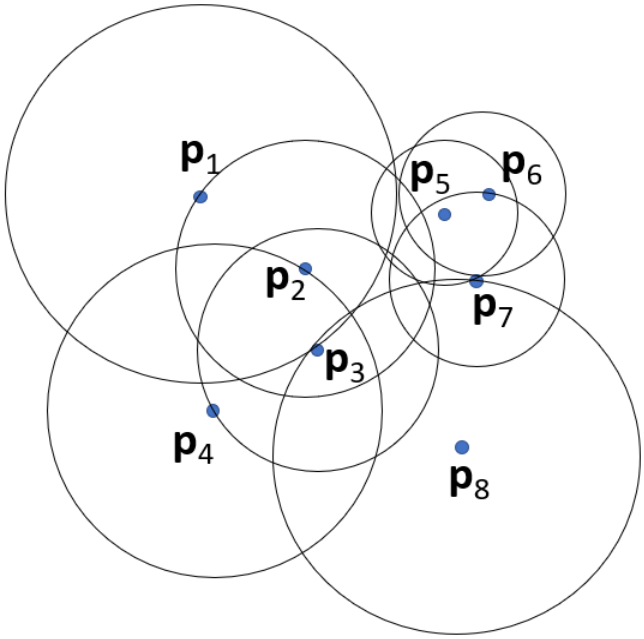

Figure 7. A cluster of 8 random points [26].

Table 1. Two-nearest neighbors of each point in Figure 7.

Query Point	Two-Nearest Neighbors
$\mathbf{p}_1$	$\mathbf{p}_2$, $\mathbf{p}_3$
$\mathbf{p}_2$	$\mathbf{p}_1$, $\mathbf{p}_3$
$\mathbf{p}_3$	$\mathbf{p}_2$, $\mathbf{p}_4$
$\mathbf{p}_4$	$\mathbf{p}_2$, $\mathbf{p}_3$
$\mathbf{p}_5$	$\mathbf{p}_6$, $\mathbf{p}_7$
$\mathbf{p}_6$	$\mathbf{p}_5$, $\mathbf{p}_7$
$\mathbf{p}_7$	$\mathbf{p}_5$, $\mathbf{p}_6$
$\mathbf{p}_8$	$\mathbf{p}_3$, $\mathbf{p}_7$

Table 2. Reverse Two-nearest neighbors of each point in Figure 7.

Query Point	Reverse Two-Nearest Neighbors
$\mathbf{p}_1$	$\mathbf{p}_2$
$\mathbf{p}_2$	$\mathbf{p}_1$, $\mathbf{p}_3$, $\mathbf{p}_4$
$\mathbf{p}_3$	$\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_4$, $\mathbf{p}_8$
$\mathbf{p}_4$	$\mathbf{p}_3$
$\mathbf{p}_5$	$\mathbf{p}_6$, $\mathbf{p}_7$
$\mathbf{p}_6$	$\mathbf{p}_5$, $\mathbf{p}_7$
$\mathbf{p}_7$	$\mathbf{p}_5$, $\mathbf{p}_6$
$\mathbf{p}_8$	N/A

4. EXPERIMENTAL RESULTS

4.1. EXPERIMENTAL SETUP

An industrial five-axis machine tool with axis sequence XYZCB and a Siemens 840D controller is used for the experimental studies conducted in this paper. Figures 8 and 9 give the picture and structural schematic of the machine tool. An Automated Precision Inc., T3 laser tracker is located on the machine tool table and an Active Target (AT) retroreflector is mounted in the spindle. The laser tracker has an accuracy of 5 μm/m and the AT has a deterministic accuracy of 12.5 μm. In Figure 9, l_{Bs} is the length between the B axis rotating center and the machine tool spindle surface. Nominally, l_{Bs} = 98 mm. The lengths between the spindle surface and the AT retroreflector, l_{st} , for the two AT mountings

are 304.9 mm for the short length and 402.9 mm for the long length. Thus, the total tool length is

$$l_t = l_{Bs} + l_{st}, \tag{32}$$

and the two tool total lengths are 402.9 mm (short tool length,98+304.9) and 500.9 mm (long tool length, 98+402.9).

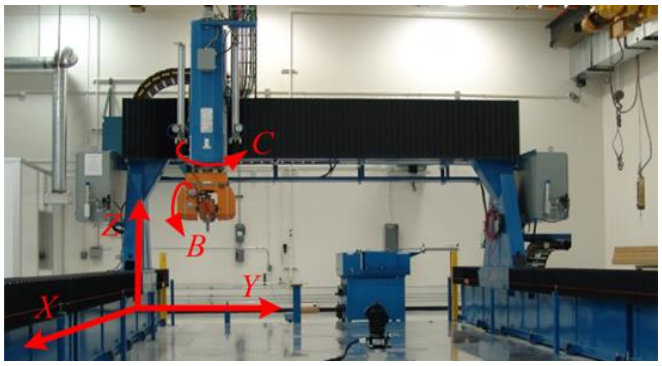

Figure 8. Industrial five-axis machine tool used for experimental studies.

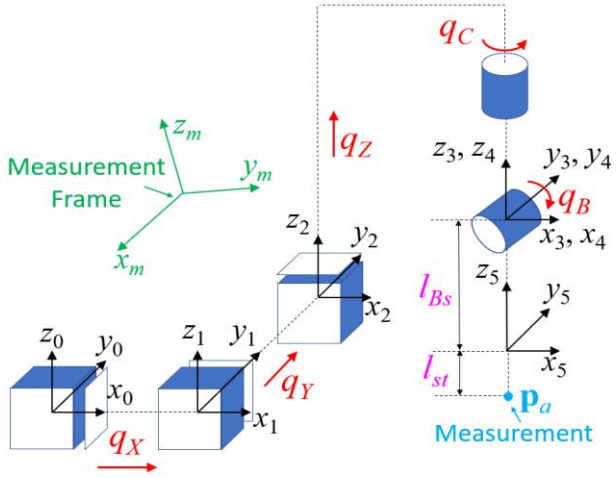

Figure 9. Schematic of industrial five-axis machine tool kinematics.

To create a set of measurements in the axis space with low discrepancy, a quasi-random sequence is used to generate the axis commands [27]. For each tool length, 295 quasi random measurements are collected. Table 3 gives the minimum and maximum axis commands and Figure 10 gives the distribution of the measurements projected in the various two-axis spaces.

The 295 measurements are divided into three sets, shown in the BZ axis space in Figure 11. To analyze the ability of the methodology to extrapolate geometric error models, 25 measurements at the bottom border of the BZ space are taken to be the extrapolation validation set. They will be used to validate the extrapolated model performance. Over the interpolation space, another 25 points are randomly selected to be the interpolation validation set. They will be used to validate the interpolated model performance. The remaining 245 measurements are used as the identification set for model construction.

Table 3. Minimum and maximum axis commands for collected 295 measurements.

Axis	**Minimum**	**Maximum**
X (mm)	83.2	6081.1
Y (mm)	37.2	2557.1
Z (mm)	7.4	988.2
C (degree)	-269.7	269.9
B (degree)	-109.5	109.9

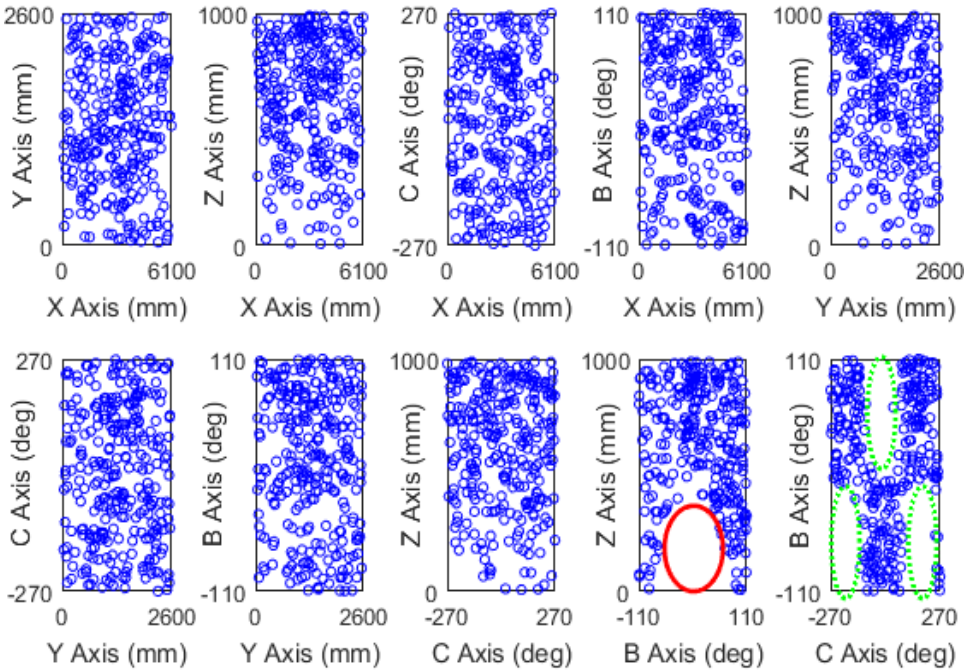

Figure 10. Distribution of 295 measurement points in two-dimensional axis spaces.

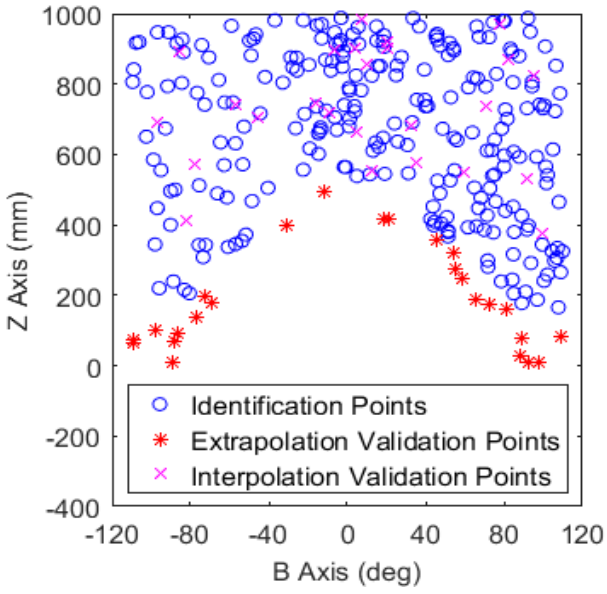

Figure 11. Distribution of identification, extrapolation and interpolation validation points in BZ space.

4.2. UNCONSTRAINED MODEL

As a baseline, a model will be constructed without constraint. This model will be referred to as the unconstrained model. According to [14], perturbation functions described by 6^{th} order polynomials are appropriate for this specific machine tool. Numerical optimization is used to minimize (11) for model identification. Here, the MATLAB optimization solver *fmincon* is used as it is capable of including nonlinear constraints during the optimization process.

The unconstrained model is constructed with the 245 identification points. Table 4 lists the mean and maximum residuals for the identification and validation sets. Figure 12 shows the compensation table functions, which are generated from the geometric error models, over the interpolation and extrapolation spaces. The model reduces the mean residual from 0.307 to 0.038 mm for the identification data set and 0.261 to 0.044 mm for the interpolation validation data set, providing 87.6% and 83.1% reductions, respectively. However, for the extrapolation validation data set, the constrained model increases the mean residual from 0.321 to 0.451 mm and the maximum residual from 0.525 to 1.909 mm. The reason for the poor model performance for the extrapolation validation set can be seen in Figure 12. For the geometric error functions that are dependent on the Z axis position, the behavior in the extrapolation space is not consistent with the behavior in the interpolation space. In this case the error slopes are much larger in the extrapolation space. This results in unrealistic error magnitudes in the extrapolation space and, thus, poor accuracy of the extrapolated model.

Table 4. Mean and maximum residuals for the identification and validation data sets (mm).

	Identification set		Interpolation validation set		Extrapolation validation set	
Model	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**
Nominal	0.307	0.739	0.261	0.539	0.321	0.525
Unconstrained	0.038	0.140	0.044	0.092	0.451	1.909

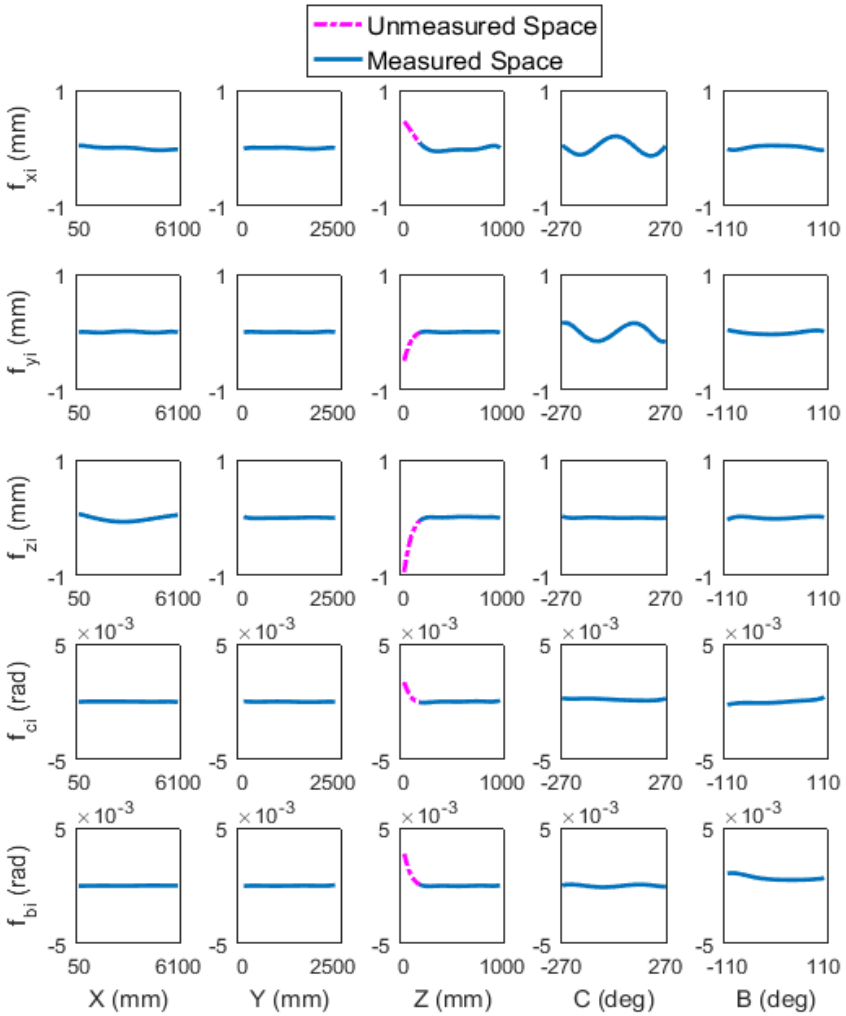

Figure 12. Compensation table functions generated from unconstrained model.

4.3. CONSTRAINED MODEL

In Figure 12, the perturbation functions dependent on the Z axis position behave unrealistic, which is caused by the limitation in Figure 2 and Figure 3. Thus, the boundary space points are located in the BZ space to determine the constraint value *c*. BORDER is applied to the 245 identification points. Here, *k* is tuned and selected as 30. As it is preferred to keep most measurements as identification points and only a few measurements are needed to validate the constrained model performance, the first 10 boundary space points identified by BORDER are used and those 10 points are named as the constraint validation set. The remaining 235 points are defined as the sub identification set. Figure 13 gives the distribution of sub identification and constraint validation points.

Using the sub identification set, constrained models are constructed with different constraint values. Figure 14 shows the performance of the sub identification set and the constraint validation set. As the constraint goes from infinity to 0.4, the maximum residual of the constraint validation set keeps decreasing while the mean residual of the sub identification set is kept nearly the same. This indicates that the interpolation space error curves are refitted to best fit the sub identification points and the error slopes over the extrapolation space are being constrained. When the constraint is set smaller than 0.4, the error slopes are over-constrained. As a result, the model gets worse for both sub identification and constraint validation sets. From the trend of maximum residual of the constraint validation set, c = 0.4 mm/half axis motion is picked as the best constraint for the final constrained model construction. Note here for other machine tools or other tool lengths, the constraint value may not have to be 0.4. Similar test and analysis should be implemented.

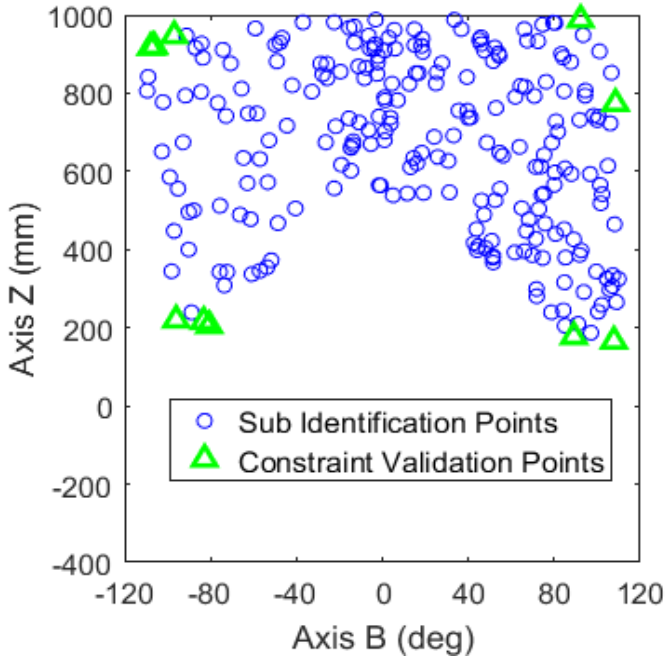

Figure 13. Distribution of sub identification and constraint validation points in BZ space.

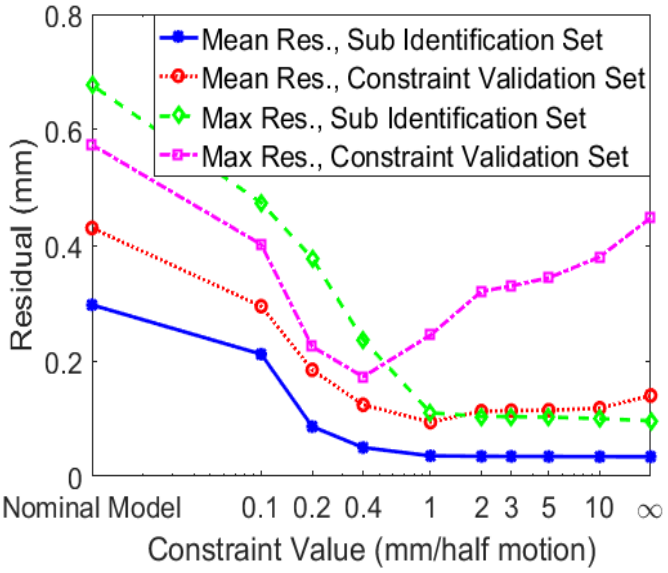

Figure 14. Mean and maximum residuals of the sub identification and constraint validation sets with different constraint values.

4.4. COMPARISON AND ANALYSIS

The final constrained model is constructed with the original 245 identification points and c = 0.4. The model performance and the identified error curves are compared between the unconstrained and constrained models. Figure 15 shows the compensation table functions generated from unconstrained and constrained models.

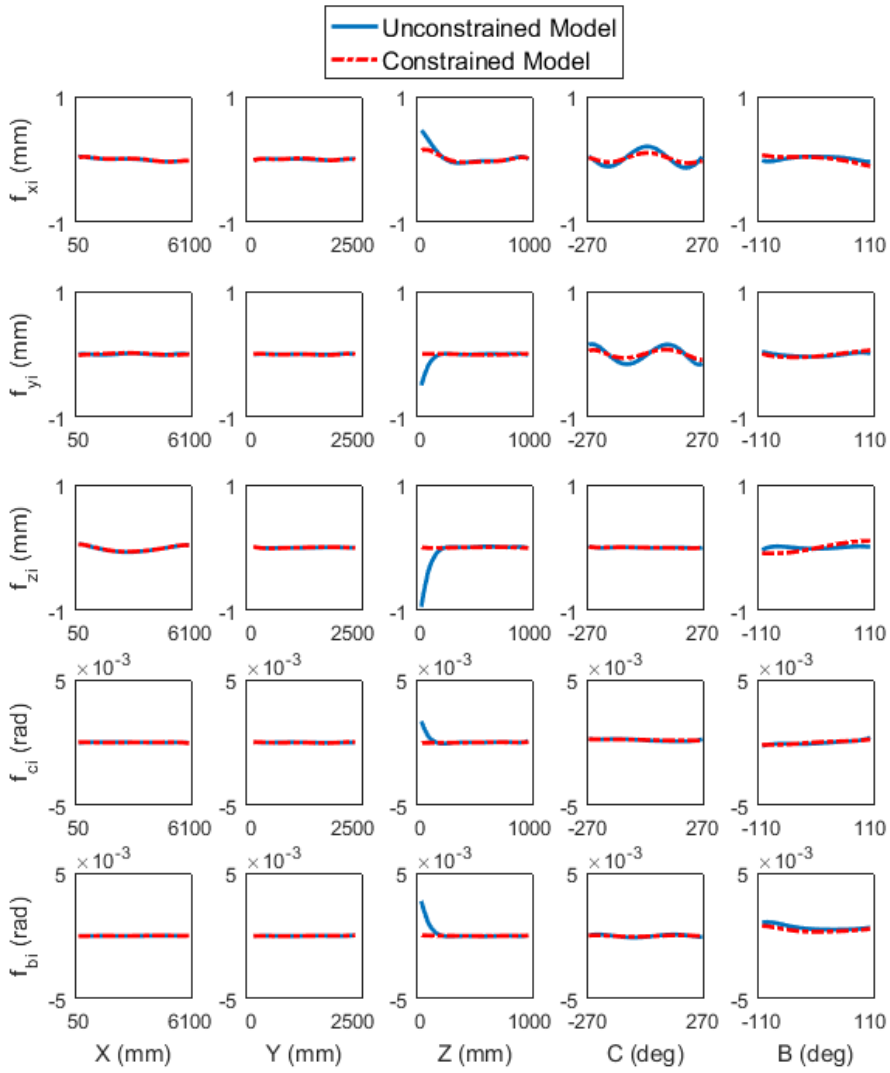

Figure 15. Compensation table functions generated from unconstrained and constrained models.

Table 5 lists the mean and maximum residuals of nominal, unconstrained and constrained models for the identification and validation data sets and Figure 16 gives the mean and maximum residuals of nominal, constrained and unconstrained models for the validation data sets. As shown in Figure 15, the third column errors are identified much flatter. The constraint successfully constrains the error slopes over the whole space. In Table 5 and Figure 16, unlike the poor performance of unconstrained model on the extrapolation validation data set, the constrained model reduces the mean residual from 0.321 mm to 0.191 mm and the maximum residual from 0.525 mm to 0.443 mm, respectively. Although the residuals for the identification and interpolation validation sets increase from the unconstrained to constrained models, the increases on the mean residual (0.016 mm and 0.012 mm) are much smaller comparing to the decrease (0.26 mm) for the extrapolation validation set. The constrained model gives a more uniform and optimal description of the geometric errors.

Table 5. Mean and maximum residuals of nominal, unconstrained and constrained models for identification and validation sets (mm).

	Identification set		**Interpolation validation set**		**Extrapolation validation set**	
Model	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**
Nominal	0.307	0.739	0.261	0.539	0.321	0.525
Unconstrained	0.038	0.140	0.044	0.092	0.451	1.909
Constrained	0.054	0.217	0.056	0.112	0.191	0.443

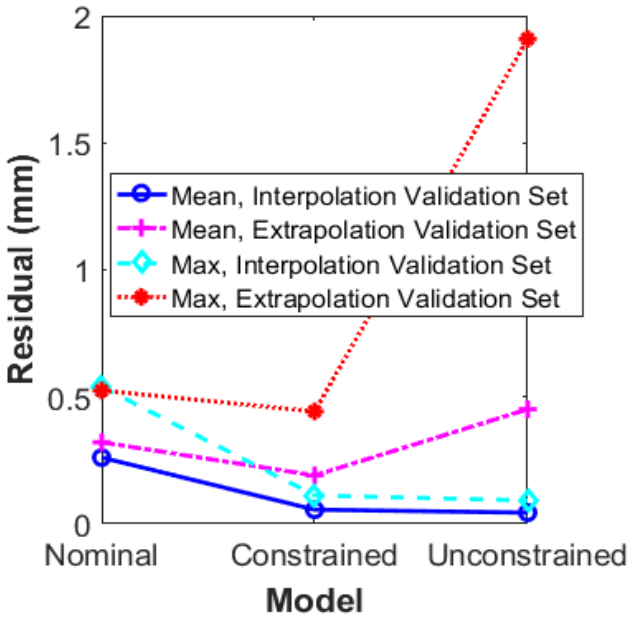

Figure 16. Mean and maximum residuals of nominal, constrained and unconstrained models for interpolation and extrapolation validation sets.

5. SUMMARY AND CONCLUSIONS

The geometric errors of machine tools are frequently corrected by indirect and direct compensation methods with external metrologies. Due to challenge in curve-fitting and the limitation of the measurement devices, the interpolated and extrapolated error models may bring unexpected errors to the actual machining work. A method of interpolating and extrapolating the error model is proposed in this paper. Based on axis perturbation model, the proposed method uses the tool tip modeled error slope, which is formed by the all perturbation function slopes, as a general constraint to correct the unrealistic phenomenon of the model errors. By adding the constraint during the model identification, the error functions are refitted such that the interpolation space errors are still well identified and the extrapolation space errors behave more realistic.

In an experimental study on an industrial five-axis machine tool, an extrapolation validation set is defined to validate the extrapolated model performance. For the unconstrained model, although the mean residuals are reduced with about 85% for the identification and interpolation validation data set, the residuals on the extrapolation validation set are increased. The maximum residual is even increased by 264%. The unconstrained model fails to well describe the boundary and extrapolation space errors. In the construction of constrained models, BORDER is used to select 10 boundary space points to validate model performance with different constraint values. The final constrained model is constructed with an uniform constraint value $c = 0.4$. In the comparison of the unconstrained and constrained models, while the unconstrained model increases the mean residual from 0.321 mm to 0.451 mm for the extrapolation validation set, the constrained model identifies more realistic errors and reduces the mean residual from 0.321 mm to 0.191 mm with a 40.5% reduction. The comparison and analysis demonstrates that the proposed method is able to refit the errors throughout the whole space such that the geometric error model is optimally-fitted. As the interpolated model is kept well, the extrapolated model is also able to improve compensation performance which will be very applicable in actual manufacturing tasks.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support for this work from the National Science Foundation and the Center for Aerospace Manufacturing Technologies at the Missouri University of Science and Technology.

APPENDIX

From [24], the linear velocity portion of the model Jacobian matrix is

$$\mathbf{J}_{\mathbf{v},AP} = \nabla_{\mathbf{q}} \mathbf{p}_{AP} . \tag{33}$$

For an industrial five-axis machine tool with axis sequence XYZCB, the modeled position of the machine tool's last frame with respect to the machine tool's base frame is

$$\mathbf{p}_{AP} = \begin{bmatrix} q_X + \delta q_X(\mathbf{q}) & q_Y + \delta q_Y(\mathbf{q}) & q_Z + \delta q_Z(\mathbf{q}) \end{bmatrix}^T . \tag{34}$$

Thus, using (33), the linear velocity portion of the model Jacobian matrix is

$$\begin{aligned} \mathbf{J}_{\mathbf{v},AP} &= \begin{bmatrix} \frac{\partial \mathbf{p}_{AP}}{\partial q_X} & \frac{\partial \mathbf{p}_{AP}}{\partial q_Y} & \frac{\partial \mathbf{p}_{AP}}{\partial q_Z} & \frac{\partial \mathbf{p}_{AP}}{\partial q_C} & \frac{\partial \mathbf{p}_{AP}}{\partial q_B} \end{bmatrix} \\ &= \begin{bmatrix} 1+\frac{df_{XX}(q_X)}{dq_X} & \frac{df_{XY}(q_Y)}{dq_Y} & \frac{df_{XZ}(q_Z)}{dq_Z} & \frac{df_{XC}(q_C)}{dq_C} & \frac{df_{XB}(q_B)}{dq_B} \\ \frac{df_{YX}(q_X)}{dq_X} & 1+\frac{df_{YY}(q_Y)}{dq_Y} & \frac{df_{YZ}(q_Z)}{dq_Z} & \frac{df_{YC}(q_C)}{dq_C} & \frac{df_{YB}(q_B)}{dq_B} \\ \frac{df_{ZX}(q_X)}{dq_X} & \frac{df_{ZY}(q_Y)}{dq_Y} & 1+\frac{df_{ZZ}(q_Z)}{dq_Z} & \frac{df_{ZC}(q_C)}{dq_C} & \frac{df_{ZB}(q_B)}{dq_B} \end{bmatrix} . \end{aligned} \tag{35}$$

Continuing to consider a five-axis machine tool with the sequence XYZCB, the modeled rotation transformation from the machine tool's base frame to the machine tool's last frame is

$$\mathbf{R}_{AP} = \mathbf{R}_C \mathbf{R}_B , \tag{36}$$

where $\mathbf{R}_C$ and $\mathbf{R}_B$ are the modeled rotation transformations for the C and B axes, respectively,

$$\mathbf{R}_C = \begin{bmatrix} \cos\left(q_C + \hat{q}_C(\mathbf{q})\right) & -\sin\left(q_C + \hat{q}_C(\mathbf{q})\right) & 0 \\ \sin\left(q_C + \hat{q}_C(\mathbf{q})\right) & \cos\left(q_C + \hat{q}_C(\mathbf{q})\right) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{37}$$

$$\mathbf{R}_B = \begin{bmatrix} \cos\left(q_B + \hat{q}_B(\mathbf{q})\right) & 0 & \sin\left(q_B + \hat{q}_B(\mathbf{q})\right) \\ 0 & 1 & 0 \\ -\sin\left(q_B + \hat{q}_B(\mathbf{q})\right) & 0 & \cos\left(q_B + \hat{q}_B(\mathbf{q})\right) \end{bmatrix}. \tag{38}$$

Taking the time derivative of $\mathbf{R}_{AP}$,

$$\begin{aligned} \dot{\mathbf{R}}_{AP} &= \frac{d\mathbf{R}_C}{dt}\mathbf{R}_B + \mathbf{R}_C\frac{d\mathbf{R}_B}{dt} \\ &= \sum_{i=X,Y,Z,C,B}\left(\dot{q}_i\left(\frac{q_C + df_{Ci}(q_i)}{dq_i}\right)\mathbf{S}(\mathbf{k})\mathbf{R}_C\mathbf{R}_B + \dot{q}_i\left(\frac{q_B + df_{Bi}(q_i)}{dq_i}\right)\mathbf{R}_C\mathbf{S}(\mathbf{j})\mathbf{R}_B\right), \\ &= \mathbf{S}\left(\sum_{i=X,Y,Z,C,B}\left(\dot{q}_i\left(\frac{q_C + df_{Ci}(q_i)}{dq_i}\right)\mathbf{k} + \dot{q}_i\left(\frac{q_B + df_{Bi}(q_i)}{dq_i}\right)\mathbf{R}_C\mathbf{j}\right)\right)\mathbf{R}_{AP} \end{aligned} \tag{39}$$

where

$$\mathbf{k} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T, \mathbf{j} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T. \tag{40}$$

Thus, the modeled angular velocity is,

$$\boldsymbol{\omega}_{AP} = \sum_{i=X,Y,Z,C,B}\boldsymbol{\omega}_{AP,i} = \sum_{i=X,Y,Z,C,B}\left(\dot{q}_i\left(\frac{q_C + df_{CX}(q_i)}{dq_i}\right)\mathbf{k} + \dot{q}_i\left(\frac{q_B + df_{Bi}(q_i)}{dq_i}\right)\mathbf{R}_C\mathbf{j}\right), \tag{41}$$

where

$$\boldsymbol{\omega}_{AP,i} = \begin{bmatrix} -\sin\left(q_C + \delta q_C(\mathbf{q})\right)\left(\frac{df_{Bi}(q_i)}{dq_i}\right) \\ \cos\left(q_C + \delta q_C(\mathbf{q})\right)\left(\frac{df_{Bi}(q_i)}{dq_i}\right) \\ \frac{df_{Ci}(q_i)}{dq_i} \end{bmatrix}\dot{q}_i, \; i = X, Y, Z, \tag{42}$$

$$\boldsymbol{\omega}_{AP,C} = \begin{bmatrix} -\sin\left(q_C + \hat{q}_C(\mathbf{q})\right)\left(\dfrac{df_{BC}(q_C)}{dq_C}\right) \\ \cos\left(q_C + \hat{q}_C(\mathbf{q})\right)\left(\dfrac{df_{BC}(q_C)}{dq_C}\right) \\ 1 + \dfrac{df_{CC}(q_C)}{dq_C} \end{bmatrix} \dot{q}_C, \tag{43}$$

$$\boldsymbol{\omega}_{AP,B} = \begin{bmatrix} -\sin\left(q_C + \hat{q}_C(\mathbf{q})\right)\left(1 + \dfrac{df_{BB}(q_B)}{dq_B}\right) \\ \cos\left(q_C + \hat{q}_C(\mathbf{q})\right)\left(1 + \dfrac{df_{BB}(q_B)}{dq_B}\right) \\ \dfrac{df_{CB}(q_B)}{dq_B} \end{bmatrix} \dot{q}_B. \tag{44}$$

The angular velocity portion of the model Jacobian matrix is thus,

$$\mathbf{J}_{\boldsymbol{\omega},AP} = \begin{bmatrix} \dfrac{\boldsymbol{\omega}_{AP,X}}{\dot{q}_X} & \dfrac{\boldsymbol{\omega}_{AP,Y}}{\dot{q}_Y} & \dfrac{\boldsymbol{\omega}_{AP,Z}}{\dot{q}_Z} & \dfrac{\boldsymbol{\omega}_{AP,C}}{\dot{q}_C} & \dfrac{\boldsymbol{\omega}_{AP,B}}{\dot{q}_B} \end{bmatrix}. \tag{45}$$

The linear and angular velocity portions of the nominal Jacobian matrix are given when the errors and slopes are zero in (44) and (54),

$$\mathbf{J}_{\mathbf{v},nom} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \tag{46}$$

$$\mathbf{J}_{\boldsymbol{\omega},nom} = \begin{bmatrix} 0 & 0 & 0 & 0 & -\sin(q_C) \\ 0 & 0 & 0 & 0 & \cos(q_C) \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{47}$$

REFERENCES

[1] Bringmann, B., Besuchet, J. P., and Rohr, L., 2008, "Systematic Evaluation of Calibration Methods," CIRP Annals–Manufacturing Technology, 57(1), pp. 529–532.

[2] ISO, 2012, "Test Code for Machine Tools Part I: Geometric Accuracy of Machine Tools Operating Under No-Load or Quasi-Static Conditions," International Organization for Standardization, Geneva, Switzerland, Standard No. ISO 230-1.

[3] Schwenke, H., Knapp, W., Haitjema, H., Weckenmann, A., Schmitt, R., and Delbressine, F., 2008, "Geometric Error Measurement and Compensation of Machines-An Update," CIRP Annals-Manufacturing Technology, 57(2), pp. 660-675.

[4] Bringmann, B., Besuchet, J. P., and Rohr, L., 2003, "Identification and Compensation of Systematic Deviations Particular to 5-Axis Machining Centers", International Journal of Machine Tools and Manufacture, 43(8), pp. 771–780.

[5] Lei, W. T., Sung, M. P., Liu, W. L., and Chuang, Y. C., 2007, "Double Ballbar Test for the Rotary Axes of Five-Axis CNC Machine Tools," International Journal of Machine Tools and Manufacture, 47(2), pp. 273-285.

[6] Ibaraki, S., Oyama, C., and Otsubo, H., 2011, "Construction of an Error Map of Rotary Axes on a Five-Axis Machining Center by Static R-Test," International Journal of Machine Tools and Manufacture, 51(3), pp. 190-200.

[7] Bringmann, B., and Knapp, W., 2006, "Model-Based 'Chase-the-Ball' Calibration of a 5-Axes Machining Center," CIRP Annals-Manufacturing Technology, 55(1), pp. 531-534.

[8] Erkan, T., and Mayer, J. R. R., 2010, "A Cluster Analysis Applied to Volumetric Errors of 5-Axis Machine Tools Obtained by Probing an Uncalibrated Artifact," CIRP Annals- Manufacturing Technology, 59(1), pp. 539–542.

[9] Ibaraki, S., Iritani, T., and Matsushita, T., 2012, "Calibration of Location Errors of Rotary Axes on 5-Axis Machine Tools by On-the-Machine Measurement Using a Touch-Trigger Probe," International Journal of Machine Tools and Manufacture, 58(1), pp. 44–53.

[10] Aguado, S., Samper, D., Santolaria, J., and Aguilar, J. J., 2012, "Identification Strategy of Error Parameter in Volumetric Error Compensation of Machine Tool Based on Laser Tracker Measurements," International Journal of Machine Tools and Manufacture, 53(1), pp. 160-169.

[11] Wang, J., Guo, J., Zhang, G., Guo, B. A., and Wang, H., 2012, "The Technical Method of Geometric Error Measurement for Multi-Axis NC Machine Tool by Laser Tracker," Measurement science and technology, 23(4), p. 045003.

[12] Hong, C., Ibaraki, S., and Matsubara, A., 2001, "Influence of Position Dependent Error of Rotary Axes on a Machining Test of Cone Frustram by 5-Axis Machine Tools," Precision Engineering, 35(1), pp. 1–11.

[13] Ibaraki, S., and Knapp, W., 2012, "Indirect Measurement of Volumetric Accuracy for 3-Axis and 5-Axis Machine Tools: A Review," International Journal of Automation Technology, 6(2), pp. 110–124.

[14] Creamer, J., Sammons, P. M., Bristow, D. A., Landers, R. G., Freeman, P. L., and Easley, S. J., 2017, "Table-Based Volumetric Error Compensation of Large Five-Axis Machine Tools," ASME Journal of Manufacturing Science and Engineering, 139(2), p. 021011.

[15] Choi, J. P., Min, B. K., and Lee, S. J., 2004, "Reduction of Machining Errors of a Three-Axis Machine Tool by On-Machine Measurement and Error Compensation System," Journal of Materials Processing Technology, 155, pp. 2056-2064.

[16] Xiang, S., and Altintas, Y., 2016, "Modeling and Compensation of Volumetric Errors for Five-Axis Machine Tools," International Journal of Machine Tools and Manufacture, 101, pp. 65-78.

[17] Potgieter, G., and Engelbrecht, A. P., 2007, "Genetic Algorithms for the Structural Optimisation of Learned Polynomial Expressions", Applied Mathematics and Computation, 186(2), pp. 1441–1466.

[18] Aaen, P., Plá, J. A., and Wood, J., 2007. Modeling and Characterization of RF and Microwave Power FETs. Cambridge University Press.

[19] Resat, H., and Mezei, M., 1993, "Studies on Free Energy Calculations. I. Thermodynamic Integration Using a Polynomial Path", The Journal of Chemical Physics, 99(8), pp. 6052–6061.

[20] Runge, C., 1901, ""Uber empirische Funktionen und die Interpolation zwischen ¨aquidistanten Ordinaten", Zeitschrift f ¨ur Mathematik und Physik, 46, pp. 224–243.

[21] Gupta, K.C., 1986, "Kinematic Analysis of Manipulators Using the Zero Reference Position Description," The International Journal of Robotics Research, 5(2), pp. 5-13.

[22] Denavit, J., 1955, "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices," ASME Journal of Applied Mechanics, pp. 215-221.

[23] Wampler, C. W., Hollerbach, J. M., and Arai, T., 1995, “An Implicit Loop Method for Kinematic Calibration and Its Application to Closed-Chain Mechanisms,” IEEE Transactions on Robotics and Automation, 11(5), pp. 710-724.

[24] Ma, L., Bazzoli, P., Sammons, P. M., Landers, R. G., and Bristow, D. A., 2017, “Modeling and Calibration of High-Order Joint-Dependent Kinematic Errors for Industrial Robots,” Robotics and Computer-Integrated Manufacturing, 50, pp. 153–167.

[25] Xia, C., Hsu, W., Lee, M. L., and Ooi, B. C., 2006, “BORDER: Efficient Computation of Boundary Points,” IEEE Transactions on Knowledge and Data Engineering, 18(3), pp. 289-303.

[26] Niederreiter, H., 1988, “Low-Discrepancy and Low-Dispersion Sequences,” Journal of Number Theory, 30(1), pp. 51-70.

II. MODELING AND CALIBRATION OF HIGH-ORDER JOINT-DEPENDENT KINEMATIC ERRORS FOR INDUSTRIAL ROBOTS

Le Ma, Patrick Bazzoli, Patrick M. Sammons, Robert G. Landers and

Douglas A. Bristow

ABSTRACT

Robot positioning accuracy is critically important in many manufacturing applications. While geometric errors such as imprecise link length and assembly misalignment dominate positioning errors in industrial robots, significant errors also arise from non-uniformities in bearing systems and strain wave gearings. These errors are characteristically more complicated than the fixed geometric errors in link lengths and assembly. Typical robot calibration methods only consider constant kinematic errors, thus, neglecting complex kinematic errors and limiting the accuracy to which robots can be calibrated. In contrast to typical calibration methods, this paper considers models containing both constant and joint-dependent kinematic errors. Constituent robot kinematic error sources are identified and kinematic error models are classified for each error source. The constituent models are generalized into a single robot kinematic error model with both constant and high-order joint-dependent error terms. Maximum likelihood estimation is utilized to identify error model parameters using measurements obtained over the measurable joint space by a laser tracker. Experiments comparing the proposed and traditional calibration methods implemented on a FANUC LR Mate 200*i* robot are presented and analyzed. While the traditional constant kinematic error model describes

79.4% of the measured error, the proposed modeling framework, constructed from measurements of 250 poses, describes 97.0% of the measured error. The results demonstrate that nearly 20% of the kinematic error in this study can be attributed to complex, joint-dependent error sources.

Key words: Industrial robots; Strain wave gearing; Calibration; Maximum likelihood estimation

1. INTRODUCTION

Industrial robots are highly flexible and repeatable automation platforms effective for a number of manufacturing tasks [1]. In some applications, a robot is programmed through a “teach” mode [2], in which the robot is manually positioned through a series of points. The robot can return to any of those points, within its repeatability, at any time by recalling them from memory. For these applications, repeatability is the critical design parameter while accuracy is not as critical. In other manufacturing applications, such as deburring and light machining, the robot will be commanded to arbitrary positions and orientations [3], thus, its repeatability and accuracy are both important. However, robot accuracy can be an order of magnitude worse than its repeatability due to various sources of errors such as component manufacturing and assembly errors, as well as joint deflection errors [4-6]. Thus, a rapid and effective method for calibrating robots is essential.

Research regarding robot calibration has been studied and well-developed over the past three decades. While the majority of the work focuses on kinematic model-based calibration, non-kinematic errors (such as elastic deformation) also play important roles in

reducing robot accuracy [7]. In [8], kinematic calibration methods were classified into open-loop, closed-loop and screw-axis measurement methods. In open-loop methods, external metrology systems are used to take measurements. Two examples of open-loop calibration methods are given in [9] and [10], in which a laser tracker and a single telescoping ballbar, respectively, were used for data collection. In closed-loop methods, external measurement devices are not needed. The robot endpoint is attached to the ground such that a mobile closed-loop kinematic chain is formed if the robot is redundant to the endpoint constraint. Then kinematic model parameters are identified using joint angle readings. The methodology and applications of this methodology are given in [11]. In screw-axis measurement methods, kinematic errors are calibrated by determining the actual transformation relationship between consecutive joints. A typical screw-axis measurement method is Circle Point Analysis (CPA) [12], two examples of which are given in [13,14].

Although a wealth of research has been conducted in robot kinematic calibration, a majority of the work only considers ideal rigid body motion and consists of identifying constant joint offsets. While a joint-independent error kinematic model may be sufficient to describe geometric errors resulting from structural errors in the robot assembly (e.g., link-length or alignment errors), many complex kinematic errors, such as periodic gear errors, cannot be sufficiently captured. Strain wave gearings, commonly used in industrial robots due to their high reduction ratio, light weight and compact size [15,16], are known to have complicated position-dependent errors caused by manufacturing tolerances, alignment errors and the gear tooth placement errors on both the circular and flexible splines [17]. Flexing of bearings will also result in non-parallel coupling of gearboxes, causing the end effector to be out of plane, higher at some positions and lower at other

positions [18]. Assembly inaccuracies, gear tooth errors and wear combine to cause position dependent and periodic kinematic errors [19]. The small magnitude of the kinematic errors in strain wave gearings will be amplified by serial links to the end effector, resulting in large, very complex robot errors. More precise models are needed to better describe these complex kinematic errors and, thus, improve post calibration performance.

A new robot kinematic calibration method capable of capturing both fixed and complex kinematic errors is developed in this paper. Six Degree of Freedom (DoF) error transformation matrices between consecutive joints, having joint-dependent error terms modeled by high-order polynomials, are used to construct a joint-dependent kinematic error model capable of describing complex geometric errors [20]. A laser tracker, having the advantages of rapid measurement speed and the ability to gather most, if not all, of the measurements in a single setup, is used for data collection. Then, error model parameters are identified with a maximum likelihood estimation algorithm [21], and a gradient search inverse kinematic compensation algorithm [22] is used for compensation.

The rest of this paper is organized as follows. Section 2 categorizes and models different robot kinematic errors. Section 3 proposes a high-order, joint-dependent kinematic error model. Identification and compensation methods are provided in Section 4. Section 5 provides the experimental results for a FANUC LR Mate 200*i* robot. Circle Point Analysis is also implemented as a representative traditional calibration method. A comparison of CPA with the proposed method is described and analyzed in Section 6. The paper is summarized and conclusions are drawn in Section 7.

2. ROBOT KINEMATIC ERROR MODELING

2.1. CHARACTERIZATION OF ROBOT KINEMATIC ERRORS

Let $\mathbf{T}_i^{i-1}$ represent a transformation from Frame *i*-1 to Frame *i* and parameterize $\mathbf{T}_i^{i-1}$ according to the Denavit-Hartenberg (DH) convention [23] as,

$$\mathbf{T}_i^{i-1} = \mathbf{T}_{RZ}(\theta_i)\mathbf{T}_{TZ}(d_i)\mathbf{T}_{TX}(a_i)\mathbf{T}_{RX}(\alpha_i), \tag{1}$$

where $\mathbf{T}_{Rj}$ is a rotation matrix about axis *j*, $\mathbf{T}_{Tj}$ is a translation matrix along axis *j*, and θ_i, d_i, a_i and α_i are model parameters. Using the DH frame assignment convention, a rotary joint can be written as

$$\mathbf{T}_i^{i-1} = \mathbf{T}_{RZ}(q_i)\mathbf{T}_{d_i,a_i,\alpha_i}, \tag{2}$$

where q_i is the joint command of link *i* and,

$$\mathbf{T}_{d_i,a_i,\alpha_i} = \mathbf{T}_{TZ}(d_i)\mathbf{T}_{TX}(a_i)\mathbf{T}_{RX}(\alpha_i), \tag{3}$$

is a fixed homogeneous transformation. Robot kinematic errors (e.g., link length error, misalignment, pitch error) will cause differences between the actual and nominal transformations. Appropriate mathematical descriptions of those errors are essential in the construction of robot kinematic error models. Several robot kinematic error sources are described and their corresponding error models are constructed as follows.

2.1.1. Rotating Center Offset Errors. The nominal transformation $\mathbf{T}_i^{i-1}$ starts from the rotating center of Frame *i*-1. Existence of assembly errors will cause an offset between the actual and nominal rotating center. In this case, the actual transformation from Frame *i*-1 to Frame *i*, $\mathbf{T}_i^{i-1}$, is

$$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{E}_{RC,i}\mathbf{T}_i^{i-1}(q_i), \tag{4}$$

where $\mathbf{E}_{RC,i}$ is a fixed error translational transformation describing the i^{th} joint rotating center offset,

$$\mathbf{E}_{RC,i} = \begin{bmatrix} 1 & 0 & 0 & \delta_{RC,X,i} \\ 0 & 1 & 0 & \delta_{RC,Y,i} \\ 0 & 0 & 1 & \delta_{RC,Z,i} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{5}$$

and $\delta_{RC,j,i}$ is the translational error along the j^{th} axis. Figure 1(*a*) gives a geometric description of a rotating center offset where Frame $X_{i-1}Y_{i-1}Z_{i-1}$ denotes the nominal Frame i-1 and Frame $X'_{i-1}Y'_{i-1}Z'_{i-1}$ denotes the actual Frame i-1.

2.1.2. Mastering Errors. The location of the zero position, referred to as mastering, is set by aligning the robot through one of several procedures such as zero degree or single axis mastering. However, a robot might lose the mastering data and remastering can introduce a small change in the zero location. With this fixed small change, the actual transformation is

$$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{T}_i^{i-1}(q_i + \Delta q_{i0}) = \mathbf{T}_{RZ}(q_i + \Delta q_{i0})\mathbf{T}_{d_i,a_i,\alpha_i}, \tag{6}$$

where Δq_{i0} is a fixed mastering error for joint i. Figure 1(*b*) shows the transformation due to mastering errors.

2.1.3. Link Length and Assembly Errors. Imprecise manufacturing of link parts and assembly misalignment errors will cause a fixed offset of the nominal link lengths (i.e., d_i and a_i) and angles between joints (i.e., q_i and α_i). The resulting transformation due to the errors in the link lengths and angles between joints can be represented by

$$\begin{aligned}\tilde{\mathbf{T}}_i^{i-1}(q_i) &= \mathbf{T}_{RZ}(q_i+\Delta q_{ia})\mathbf{T}_{TZ}(d_i+\Delta d_i)\mathbf{T}_{TX}(a_i+\Delta a_i)\mathbf{T}_{RX}(\alpha_i+\Delta\alpha_i)\\ &= \mathbf{T}_{RZ}(q_i+\Delta q_{ia})\mathbf{T}_{d_i,a_i,\alpha_i}\mathbf{E}_{LA,i}\end{aligned}, \tag{7}$$

where Δq_{ia}, Δd_i, Δa_i and $\Delta\alpha_i$ are fixed link length and assembly errors and $\mathbf{E}_{LA,i}$ is a fixed link length and assembly error transformation,

$$\mathbf{E}_{LA,i} = \left(\mathbf{T}_{d_i,a_i,\alpha_i}\right)^{-1}\mathbf{T}_{TZ}(d_i+\Delta d_i)\mathbf{T}_{TX}(a_i+\Delta a_i)\mathbf{T}_{RX}(\alpha_i+\Delta\alpha_i). \tag{8}$$

Figure 1(*c*) describes the transformations due to these errors using the DH convention.

2.1.4. Pitch Errors. Pitch error is an error in the gearing that is caused by the runout of the gear flank groove. The pitch error will affect the nominal gear ratio such that the nominal joint command, qi, will be amplified or attenuated. Further, the gear teeth may not be ideally evenly distributed; therefore, the pitch error may also be a function of the gear angle. In this case, the actual transformation will be

$$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{T}_{RZ}\left(r(q_i)q_i\right)\mathbf{T}_{d_i,a_i,\alpha_i}, \tag{9}$$

where $r(q_i)$ is a joint-dependent correcting ratio for pitch error. Figure 1(*d*) illustrates the transformation due to pitch errors.

2.1.5. Strain Wave Gearing Errors. Strain wave gearings are widely used in robotic transmission systems. A strain wave gearing, shown in Figure 2, is comprised of three components: a flexible spline, a wave generator and a circular spline. The wave generator, inserted into the flexible spline, will rotate as the input. Although strain wave gearings have the advantages of compact size, small weight and high gear ratio, they tend to have positional errors as a function of the motor position. The authors in [19] found that strain wave gearing errors consist of a basic kinematic error component and a second position-dependent error component caused by inherent torsional flexibility. This strain

wave gearing position error will induce a joint-dependent planar error for the robot joint output. According to [19], a Fourier series expansion can be used to express the kinematic errors of strain wave gearing as a function of the joint command,

$$\tilde{\mathbf{T}}_i^{i-1}(q_i)=\mathbf{T}_{RZ}\left(q_i+\sum_{j=1}^{k}h_k\cos\left(k\omega q_i+\varphi_k\right)\right)\mathbf{T}_{d_i,a_i,\alpha_i}=\mathbf{T}_{RZ}\left(q_i+\Delta q_i(q_i)\right)\mathbf{T}_{d_i,a_i,\alpha_i}, \quad (10)$$

where h_k, ω and φ_k are the amplitude, frequency and phase shift, respectively, of the Fourier expansion and $\Delta q_i(q_i)$ represents the strain wave gearing error that is dependent on the joint command, q_i. Figure 1(*e*) shows the transformation modeling strain wave gearing errors.

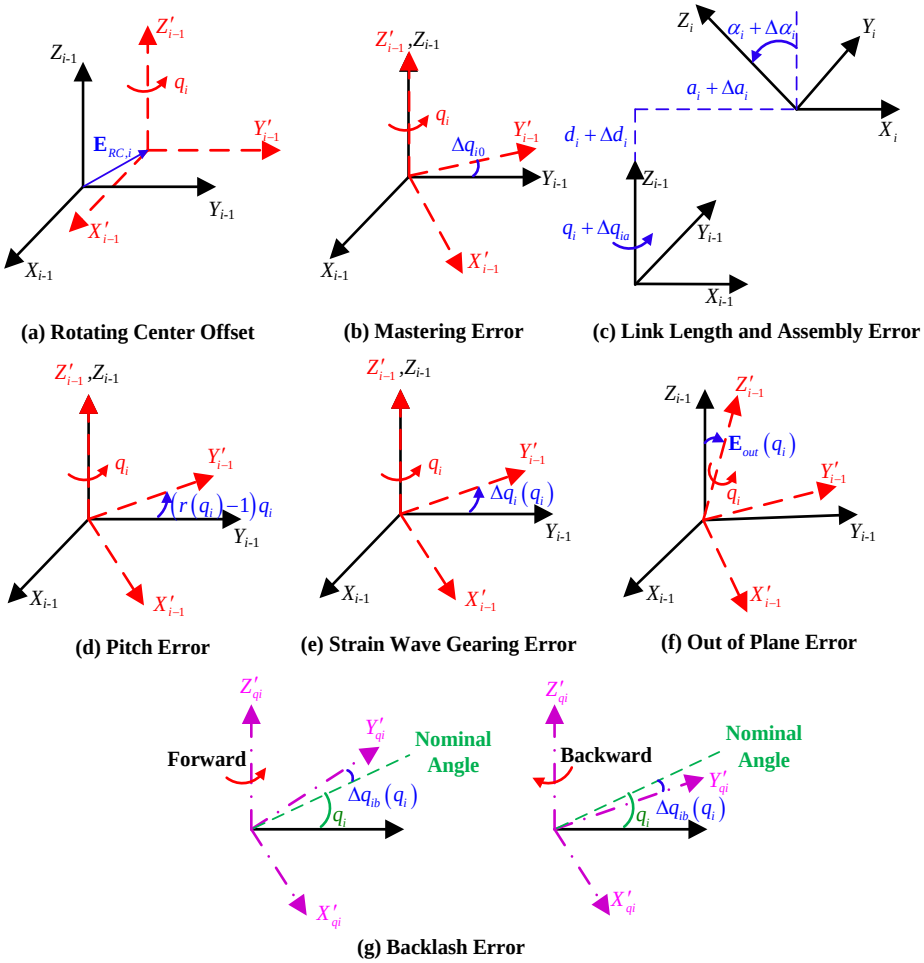

Figure 1. Schematic description of various robot kinematic errors.

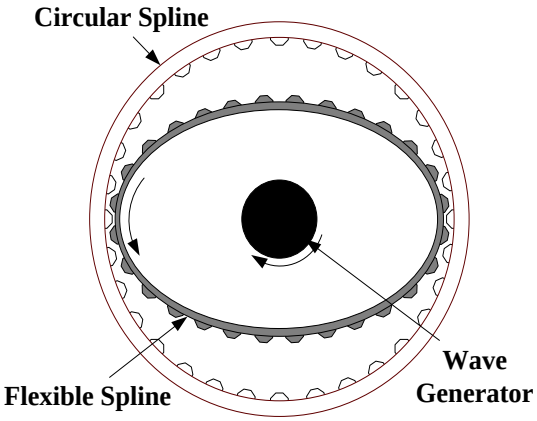

Figure 2. Strain wave gearing schematic.

2.1.6. Out of Plane Errors. Out of plane link deformations exist when a non-uniform load is applied on the output shaft. As a result, the true link end position may be higher than the desired link end position at some angles and lower at other angles. This out of plane error is also joint-dependent and can be described as

$$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{E}_{out}(q_i)\mathbf{T}_{RZ}(q_i)\mathbf{T}_{d_i,a_i,\alpha_i}, \tag{11}$$

where $\mathbf{E}_{out}(q_i)$ is the out of plane error transformation matrix,

$$\mathbf{E}_{out}(q_i) = \begin{bmatrix} \mathbf{R}_{out}(q_i) & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}, \tag{12}$$

and $\mathbf{R}_{out}(q_i)$ describes the out of plane orientation error. Figure 1(*f*) provides a schematic description of the transformation caused by out of plane errors.

2.1.7. Backlash Errors. Backlash is known to occur when the rotating direction changes [24] due to imperfect meshing of gear teeth. For a revolute joint, offsets will occur between the actual and nominal positions. Often, backlash is modeled as a constant error with the same magnitude for both the forward and backward motions. However, backlash errors are often joint-dependent. Figure 3 is a plot of the angular errors, calculated with

respect to the middle of the measurements, when the fifth joint of an industrial robot (described below) is approached from the forward and backward directions. The results illustrate how backlash can change with the joint angle. Thus, transformation due to backlash error is described by

$$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{T}_i^{i-1}\left(q_i + s_i \Delta q_{ib}(q_i)\right) = \mathbf{T}_{RZ}\left(q_i + s_i \Delta q_{ib}(q_i)\right)\mathbf{T}_{d_i,a_i,\alpha_i}, \tag{13}$$

where $\Delta q_{ib}(q_i)$ is the joint-dependent backlash error and

$$s_i = \begin{cases} 1, & \text{forward motion} \\ -1, & \text{backward motion} \end{cases}. \tag{14}$$

Figure 1(*g*) gives the schematic description of the transformation error due to backlash where Frame $X'_{qi}Y'_{qi}Z'_{qi}$ denotes the actual position after the rotation from the nominal position q_i.

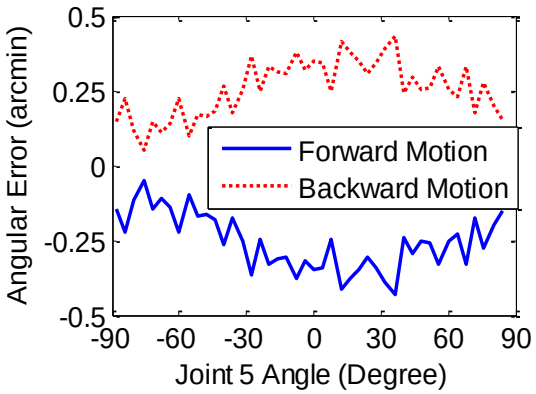

Figure 3. Angular errors from forward and backward motions of Joint 5.

2.2. GENERAL KINEMATIC ERROR MODEL

The robot error sources are summarized in Table 1. Combining the effects of the error sources results in a generalized error model of the form,

$$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{E}_{G,i}(q_i, s_i)\mathbf{T}_i^{i-1}(q_i)\mathbf{E}_{LA,i}, \tag{15}$$

where $\mathbf{E}_{G,i}(q_i, s_i)$ is a generalized joint-position and direction dependent correction.

Table 1. Robot kinematic error sources and corresponding model descriptions.

Kinematic Error	**Model Description**
Rotating Center Offset Errors	$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{E}_{RC,i}\mathbf{T}_i^{i-1}(q_i)$
Mastering Errors	$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{T}_{RZ}(\Delta q_{i0})\mathbf{T}_i^{i-1}(q_i)$
Link Length and Assembly Errors	$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{T}_{RZ}(\Delta q_{ia})\mathbf{T}_i^{i-1}(q_i)\mathbf{E}_{LA,i}$
Pitch Errors	$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{T}_{RZ}(r(q_i)q_i)\mathbf{T}_i^{i-1}(q_i)$
Strain Wave Gearing Errors	$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{T}_{RZ}(\Delta q_i(q_i))\mathbf{T}_i^{i-1}(q_i)$
Out of Plane Errors	$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{E}_{out}(q_i)\mathbf{T}_i^{i-1}(q_i)$
Backlash Errors	$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{T}_{RZ}(s_i\Delta q_{ib}(q_i))\mathbf{T}_i^{i-1}(q_i)$

3. HIGH-ORDER JOINT-DEPENDENT KINEMATIC ERROR MODEL

For an *n*-joint robot, the nominal kinematic model can be represented as

$$\mathbf{F}_n(\mathbf{q}) = \mathbf{T}_1^0(q_1)\mathbf{T}_2^1(q_2)\ldots\mathbf{T}_n^{n-1}(q_n), \tag{16}$$

where $\mathbf{F}_n$ is the nominal transformation from the robot base frame to Frame *n* and $\mathbf{q} = [q_1, q_2,\ldots,q_n]^T$ is the nominal joint command vector. Using the generalized error model (15), the actual transformation of an *n*-joint robot is,

$$\begin{aligned}\mathbf{F}_a(\mathbf{q}) &= \tilde{\mathbf{T}}_1^0\tilde{\mathbf{T}}_2^1\ldots\tilde{\mathbf{T}}_n^{n-1} \\ &= \underbrace{\mathbf{E}_{G,1}(q_1,s_1)\mathbf{T}_1^0(q_1)\mathbf{E}_{LA,1}}_{\text{Joint 1}}\ldots\underbrace{\mathbf{E}_{G,n}(q_n,s_n)\mathbf{T}_n^{n-1}(q_n)\mathbf{E}_{LA,n}}_{\text{Joint } n},\end{aligned} \tag{17}$$

Adjacent error transformations can be combined as

$$\mathbf{E}_i^{i-1}(q_i,s_i) = \mathbf{E}_{LA,i-1}\mathbf{E}_{G,i}(q_i,s_i), \tag{18}$$

yielding the complete kinematic error description,

$$\mathbf{F}_a(\mathbf{q}) = \mathbf{E}_1^0(q_1,s_1)\mathbf{T}_1^0(q_1)\mathbf{E}_2^1(q_2,s_2)\mathbf{T}_2^1(q_2)\ldots\mathbf{E}_n^{n-1}(q_n,s_n)\mathbf{T}_n^{n-1}(q_n), \tag{19}$$

where $\mathbf{E}_{LA,0} = \mathbf{I}$. Note here the last joint correction is removed since it will depend on tool-specific mounting variations. For small kinematic errors, the error kinematics between links can be approximated as [20],

$$\mathbf{E}_i^{i-1}(q_i,s_i) \approx \begin{bmatrix} 1 & -\varepsilon_Z(q_i,s_i) & \varepsilon_Y(q_i) & \delta_X(q_i) \\ \varepsilon_Z(q_i,s_i) & 1 & -\varepsilon_X(q_i) & \delta_Y(q_i) \\ -\varepsilon_Y(q_i) & \varepsilon_X(q_i) & 1 & \delta_Z(q_i) \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{20}$$

where ε_X, ε_Y and ε_Z are small rotations around the X, Y and Z axes, respectively, of Frame i-1 and δ_X, δ_Y and δ_Z are small translations along the X, Y and Z axes, respectively, of Frame i-1. Since axis Z_i is the rotating axis, the effect of backlash errors are included in the rotational error function ε_Z.

To capture both the fixed and joint-dependent errors, a basis set of sufficient order is used. Here, Chebyshev polynomials on a normalized base are used. Given a parameter λ in the interval [-1 1], a Chebyshev polynomial has the form,

$$C(\lambda) = a_0 + a_1c_1(\lambda) + a_2c_2(\lambda) + \cdots + a_mc_m(\lambda), \tag{21}$$

where

$$\begin{aligned} c_0(\lambda)=1,\ c_1(\lambda)=\lambda,\ c_2(\lambda)=2\lambda^2-1,\ c_3(\lambda)=4\lambda^3-3\lambda, \\ c_4(\lambda)=8\lambda^4-8\lambda^2+1,\ldots,\ c_{m+1}(\lambda)=2\lambda c_m(\lambda)-c_{m-1}(\lambda) \end{aligned}, \tag{22}$$

m denotes the order of the Chebyshev polynomial and a_0, a_1, a_2, …, a_m are the polynomial coefficients. Thus, the error terms can be represented by m^{th} order Chebyshev polynomials as

$$\begin{aligned} \varepsilon_j(\hat{q}_i)=a_{0j}+a_{1j}c_1(\hat{q}_i)+a_{2j}c_2(\hat{q}_i)+\cdots+a_{mj}c_m(\hat{q}_i) \\ \delta_j(\hat{q}_i)=b_{0j}+b_{1j}c_1(\hat{q}_i)+b_{2j}c_2(\hat{q}_i)+\cdots+b_{mj}c_m(\hat{q}_i) \end{aligned}, \tag{23}$$

where j denotes the axis (i.e., X, Y and Z) and

$$\hat{q}_i=\frac{2\left(q_i-q_{i,\min}\right)}{\left(q_{i,\max}-q_{i,\min}\right)}-1, \tag{24}$$

denotes the i^{th} linearly mapped joint command where the joint range is scaled to the interval [-1 1], and $q_{i,\min}$ and $q_{i,\max}$ are the minimum and maximum joint angles, respectively. Thus, the zero order term in (23) can be regarded as the constant error description for joint i-1 and other terms are the joint-dependent errors for joint i. In this framework, modeling of the error kinematics between joints corresponds to selecting a sufficient order m and appropriate model coefficients, a_{0j}, a_{1j}, …, a_{mj}, b_{0j}, b_{1j}, …, b_{mj}.

4. MEASUREMENT, IDENTIFICATION AND COMPENSATION

4.1. MEASUREMENT

A laser tracker is used to obtain the 3-D Cartesian coordinates of a spherical tool attached to the robot end effector. Figure 4 shows the measurement system setup. The kinematic error model in (19) consists of transformations from the robot base frame to

Frame *n*. As measurements of the tool tip are collected with respect to a metrology frame, a static transformation from Frame *n* to the tool tip and a static transformation from the metrology frame to the robot base frame are needed. Two sources of error in the measurement are considered. The first is in the measurement of the tool tip and is treated as a measured Cartesian error. The second is in the robot positioning repeatability error, which is treated as a joint positioning error. Incorporating these elements into the robot kinematic model, the measured position with respect to the metrology frame, $\mathbf{p}_a^m(\mathbf{q})$, is

$$\mathbf{p}_a^m(\mathbf{q}) = \mathbf{T}_0^m \mathbf{F}_a(\mathbf{q}+\mathbf{v})\mathbf{p}_T^n + \boldsymbol{\xi}, \tag{25}$$

where $\mathbf{T}_0^m$ is a transformation from the metrology frame to the robot base frame, $\mathbf{v}$ is a stochastic joint-positioning error, $\mathbf{p}_T^n$ is a tool length vector and $\boldsymbol{\xi}$ is a measured Cartesian error. Figure 5 gives a schematic structure of the measurement model. By setting the metrology frame close to the robot base frame, the transformation, $\mathbf{T}_0^m$, can be described with small fixed rotation and translation errors,

$$\mathbf{T}_0^m = \begin{bmatrix} 1 & -\varepsilon_{Z,0} & \varepsilon_{Y,0} & \delta_{X,0} \\ \varepsilon_{Z,0} & 1 & -\varepsilon_{X,0} & \delta_{Y,0} \\ -\varepsilon_{Y,0} & \varepsilon_{X,0} & 1 & \delta_{Z,0} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{26}$$

The tool length vector, $\mathbf{p}_T^n$, is also fixed, and is

$$\mathbf{p}_T^n = \begin{bmatrix} p_{TX} \\ p_{TY} \\ p_{TZ} \\ 1 \end{bmatrix} = \begin{bmatrix} l_X + \delta_{X,T} \\ l_Y + \delta_{Y,T} \\ l_Z + \delta_{Z,T} \\ 1 \end{bmatrix}, \tag{27}$$

where p_{TX}, p_{TY} and p_{TZ} are the translations along the *X*, *Y* and *Z* axes, respectively, of Frame *n*, l_X, l_Y and l_Z are the estimated translations and $\delta_{X,T}$, $\delta_{Y,T}$ and $\delta_{Z,T}$ are small corrections of

the estimated values. Figure 6 gives the description of the tool length vector in Frame *n*. Since the measurement tool can only collect position information, error measurements using the same set of joint commands are collected in three measurement tool configurations to determine the robot orientation error information. Figure 7 shows the three configurations of the measurement tool with the robot in its zero position.

Figure 4. Measurement system setup.

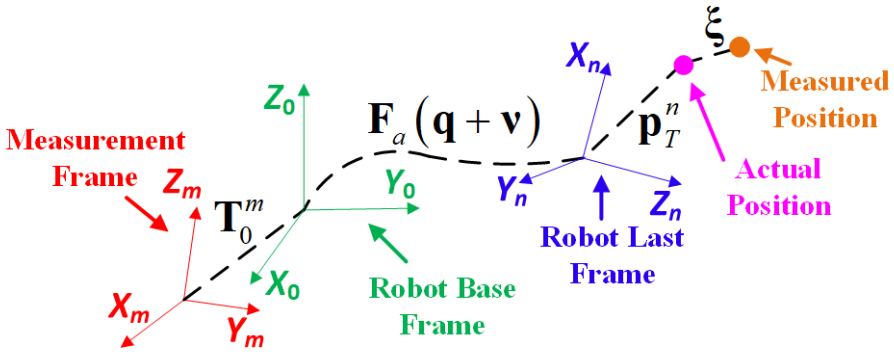

Figure 5. Measurement model schematic structure.

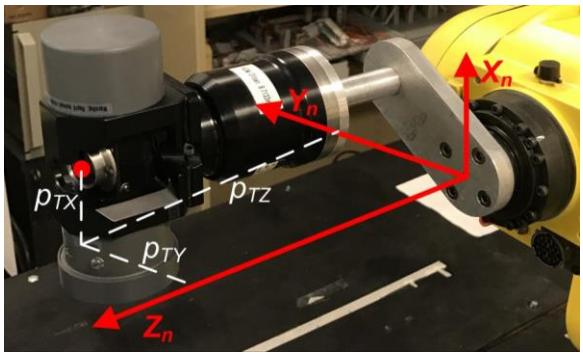

Figure 6. Tool length vector in robot's last frame.

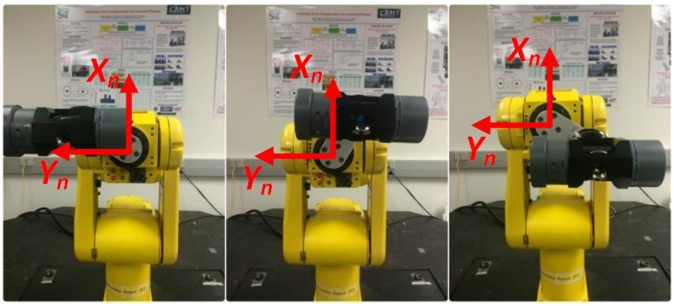

Figure 7. Three measurement tool configurations in robot's zero position.

4.2. MAXIMUM LIKELIHOOD IDENTIFICATION

The method of identifying model error parameters used in this paper is based on the implicit loop algorithm described in [21], using a maximum likelihood estimator. An advantage of this method is that both positioning repeatability and measurement error are considered in order to avoid over-fitting model parameters to the measurement data.

Consider N measured robot poses, $\mathbf{q}_k$, k=1,…,N, acquired with t=1,2,3 tool configurations and denote the collected measurement with respect to the measurement frame, m, as $\mathbf{p}^m_{a,k,t}(\mathbf{q}_k)$. Denote the tool length for each tool configuration, with respect to the end effector frame, n, as $\mathbf{p}^n_{T,t}$, and robot joint positioning errors as $\mathbf{v}_{k,t}$. Collect the base

frame alignment errors in $\mathbf{T}_0^m$ as,

$$\mathbf{b}_{base} = \begin{bmatrix} \varepsilon_{X,0} & \varepsilon_{Y,0} & \varepsilon_{Z,0} & \delta_{X,0} & \delta_{Y,0} & \delta_{Z,0} \end{bmatrix}^T, \tag{28}$$

the tool length errors for each tool configuration in $\mathbf{p}_{T,t}^n$ as,

$$\mathbf{b}_T = \begin{bmatrix} \delta_{X,T,1} & \delta_{Y,T,1} & \delta_{Z,T,1} & \delta_{X,T,2} & \delta_{Y,T,2} & \delta_{Z,T,2} & \delta_{X,T,3} & \delta_{Y,T,3} & \delta_{Z,T,3} \end{bmatrix}^T, \tag{29}$$

and the robot kinematic error parameters in $\mathbf{F}_a$ as,

$$\mathbf{b}_J = \begin{bmatrix} \mathbf{b}_{J1}^T & \mathbf{b}_{J2}^T & \cdots & \mathbf{b}_{Jn}^T \end{bmatrix}^T, \tag{30}$$

where $\mathbf{b}_{Ji}$ are themselves a collection of the Chebyshev polynomials coefficients in $\mathbf{E}_i^{i-1}$. Assume that the model structure developed above contains all kinematic error descriptions of the actual robot. Then, for the appropriate model parameters, the measurement error for all three tool configurations can be collected at each pose as,

$$\boldsymbol{\xi}_k = \begin{bmatrix} \boldsymbol{\xi}_{k,1} \\ \boldsymbol{\xi}_{k,2} \\ \boldsymbol{\xi}_{k,3} \end{bmatrix} = \begin{bmatrix} \mathbf{p}_{a,k,1}^m(\mathbf{q}_k) - \mathbf{T}_0^m \mathbf{F}_a(\mathbf{q}_k + \boldsymbol{\nu}_{k,1}) \mathbf{p}_{T,1}^n \\ \mathbf{p}_{a,k,2}^m(\mathbf{q}_k) - \mathbf{T}_0^m \mathbf{F}_a(\mathbf{q}_k + \boldsymbol{\nu}_{k,2}) \mathbf{p}_{T,2}^n \\ \mathbf{p}_{a,k,3}^m(\mathbf{q}_k) - \mathbf{T}_0^m \mathbf{F}_a(\mathbf{q}_k + \boldsymbol{\nu}_{k,3}) \mathbf{p}_{T,3}^n \end{bmatrix}, \tag{31}$$

where $\mathbf{T}_0^m$, $\mathbf{F}_a$, and $\mathbf{p}_{T,t}^n$ implicitly include the base frame errors, Chebyshev kinematic errors, and tool length errors, respectively. Now, collect joint positioning error for each pose as, $\boldsymbol{\nu}_k = \begin{bmatrix} \boldsymbol{\nu}_{k,1}^T & \boldsymbol{\nu}_{k,2}^T & \boldsymbol{\nu}_{k,3}^T \end{bmatrix}^T$, and assume joint positioning error and meausrement error follow known normal distributions. Let $\boldsymbol{\Sigma}_{\boldsymbol{\nu}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\xi}}$ be the covariance matrices for the collected joint positioning and measurement error at each pose, respectively, which are given by,

$$\begin{aligned} \boldsymbol{\Sigma}_{\boldsymbol{\nu}} &= diag\left(\sigma_{J1}^2, \ldots, \sigma_{Jn}^2, \sigma_{J1}^2, \ldots, \sigma_{Jn}^2, \sigma_{J1}^2, \ldots, \sigma_{Jn}^2\right), \\ \boldsymbol{\Sigma}_{\boldsymbol{\xi}} &= diag\left(\sigma_m^2, \sigma_m^2, \sigma_m^2, 0, \sigma_m^2, \sigma_m^2, \sigma_m^2, 0, \sigma_m^2, \sigma_m^2, \sigma_m^2, 0\right), \end{aligned} \tag{32}$$

where σ_{Ji}, i=1,…,n is the standard deviation of the i^{th} joint positioning error and σ_m is the standard deviation for the measurement error.

The maximum likelihood estimation of the kinematic error model is obtained by minimizing the inverse-covariance weighted stochastic errors as,

$$V\left(\mathbf{v}_1,\ldots,\mathbf{v}_N,\mathbf{b}_{base},\mathbf{b}_J,\mathbf{b}_T\right)=\min_{\substack{\mathbf{v}_1,\ldots,\mathbf{v}_N,\\ \mathbf{b}_{base},\mathbf{b}_J,\mathbf{b}_T}}\sum_{k=1}^{N}\left(\mathbf{v}_k^T\boldsymbol{\Sigma}_{\mathbf{v}}^{-1}\mathbf{v}_k+\boldsymbol{\xi}_k^T\boldsymbol{\Sigma}_{\xi}^{-1}\boldsymbol{\xi}_k\right), \tag{33}$$

subject to the implicit loop constraint, (31). Numerical optimization is used to obtain the minimum, and thus, the kinematic error model. To improve optimization speed and accuracy, an analytical gradient of the optimization function (33) can be utilized. The analytical solution for the gradient is given by,

$$\nabla V=\left[\frac{\partial V}{\partial \mathbf{v}_1}^T\cdots\frac{\partial V}{\partial \mathbf{v}_N}^T\quad\frac{\partial V}{\partial \mathbf{b}_{base}}^T\quad\frac{\partial V}{\partial \mathbf{b}_J}^T\quad\frac{\partial V}{\partial \mathbf{b}_T}^T\right]^T, \tag{34}$$

where each component of the gradient is solved in the following four subsections.

4.2.1. Partial Derivative of Joint Positioning Error. For the i^{th} joint positioning error of the k^{th} command, the partial derivatives of the joint positioning errors for all three measurement tool configurations are

$$\begin{aligned}\frac{\partial V}{\partial \nu_{k,i,1}}&=2\nu_{k,i,1}\sigma_{Ji,1}^2+2\boldsymbol{\xi}_k^T\boldsymbol{\Sigma}_{\xi}^{-1}\begin{bmatrix}-\mathbf{T}_0^m\dfrac{\partial\left(\mathbf{F}_a\left(\mathbf{q}_k+\mathbf{v}_{k,1}\right)\mathbf{p}_{T,1}^n\right)}{\partial\left(\nu_{k,i,1}\right)}\\ -\mathbf{T}_0^m\dfrac{\partial\left(\mathbf{F}_a\left(\mathbf{q}_k+\mathbf{v}_{k,2}\right)\mathbf{p}_{T,2}^n\right)}{\partial\left(\nu_{k,i,1}\right)}\\ -\mathbf{T}_0^m\dfrac{\partial\left(\mathbf{F}_a\left(\mathbf{q}_k+\mathbf{v}_{k,3}\right)\mathbf{p}_{T,3}^n\right)}{\partial\left(\nu_{k,i,1}\right)}\end{bmatrix}\\ &=2\nu_{k,i,1}\sigma_{Ji,1}^2-2\boldsymbol{\xi}_k^T\boldsymbol{\Sigma}_{\xi}^{-1}\begin{bmatrix}\left(\mathbf{T}_0^m\begin{bmatrix}\mathbf{J}_{v,i,1}^{mod}\\ 0\end{bmatrix}\right)^T & \begin{bmatrix}0&0&0&0\end{bmatrix}^T & \begin{bmatrix}0&0&0&0\end{bmatrix}^T\end{bmatrix}^T\end{aligned}, \tag{35}$$

$$\frac{\partial V}{\partial \nu_{k,i,2}} = 2\nu_{k,i,2}\sigma_{Ji,2}^2 - 2\boldsymbol{\xi}_k^T \boldsymbol{\Sigma}_\xi^{-1} \left[\begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \quad \left(\mathbf{T}_0^m \begin{bmatrix} \mathbf{J}_{\nu,i,2}^{mod} \\ 0 \end{bmatrix} \right)^T \quad \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \right]^T, \quad (36)$$

$$\frac{\partial V}{\partial \nu_{k,i,3}} = 2\nu_{k,i,3}\sigma_{Ji,3}^2 - 2\boldsymbol{\xi}_k^T \boldsymbol{\Sigma}_\xi^{-1} \left[\begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \quad \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \quad \left(\mathbf{T}_0^m \begin{bmatrix} \mathbf{J}_{\nu,i,3}^{mod} \\ 0 \end{bmatrix} \right)^T \right]^T, \quad (37)$$

where $\mathbf{J}_{\nu,i,t}^{mod}$ is the linear velocity portion of the model Jacobian with the t^{th} measurement tool orientation. The details of $\mathbf{J}_{\nu,i,t}^{mod}$ are shown in the appendix.

4.2.2. Partial Derivative of Base Frame Error. Let α_j denote the j^{th} element of $\mathbf{b}_{base}$. Then, the partial derivative of the base frame error is

$$\frac{\partial V}{\partial \alpha_j} = -\sum_{k=1}^{N} \left(2\boldsymbol{\xi}_k^T \boldsymbol{\Sigma}_\xi^{-1} \frac{\partial \mathbf{T}_0^m}{\partial \alpha_j} \begin{bmatrix} \mathbf{p}_{k,1}^0 & \mathbf{p}_{k,2}^0 & \mathbf{p}_{k,3}^0 \end{bmatrix}^T \right), \quad (38)$$

where

$$\mathbf{p}_{k,t}^0 = \mathbf{F}_a \left(\mathbf{q}_k + \mathbf{v}_{k,t} \right) \mathbf{p}_{T,t}^n, t = 1,2,3, \quad (39)$$

and $\frac{\partial \mathbf{T}_0^m}{\partial \alpha_j}$ is a straightforward derivative of $\mathbf{T}_0^m$.

4.2.3. Partial Derivative of Joint Kinematic Error. Letting the order of the Chebyshev polynomials used to model the error components be *m*, the error parameters in b_{Ji} is

$$\mathbf{b}_{Ji} = \begin{bmatrix} b_{0,\varepsilon X,i}, \cdots, b_{m,\varepsilon X,i} & b_{0,\varepsilon Y,i}, \cdots, b_{m,\varepsilon Y,i} & \cdots & b_{0,\delta Z,i}, \cdots, b_{m,\delta Z,i} \end{bmatrix}^T. \quad (40)$$

Letting $b_{Ji,j}$ denote the j^{th} element of $\mathbf{b}_{Ji}$, the partial derivative of the joint kinematic error is

$$\frac{\partial V}{\partial b_{Ji,j}} = -\sum_{k=1}^{N} \left(2\boldsymbol{\xi}_k^T \boldsymbol{\Sigma}_\xi^{-1} \mathbf{A}_{i,k,pre} \frac{\partial \mathbf{E}_{i,k}^{i-1}}{\partial b_{Ji,j}} \begin{bmatrix} \mathbf{p}_{i,k,1,post} & \mathbf{p}_{i,k,2,post} & \mathbf{p}_{i,k,3,post} \end{bmatrix}^T \right), \quad (41)$$

where

$$\mathbf{A}_{i,k,pre} = \mathbf{T}_0^m \prod_{h=1}^{i-1} \left(\mathbf{E}_{h,k}^{h-1} \mathbf{T}_{h,k}^{h-1} \right), \tag{42}$$

$$\mathbf{p}_{i,k,t,post} = \mathbf{T}_{h,k}^{h-1} \prod_{h=i+1}^{n} \left(\mathbf{E}_{h,k}^{h-1} \mathbf{T}_{h,k}^{h-1} \right) \mathbf{p}_{T,t}^{n}, \; t = 1,2,3, \tag{43}$$

and $\dfrac{\partial \left(\mathbf{E}_{i,k}^{i-1} \right)}{\partial b_{Ji,j}}$ is a straightforward derivative of $\mathbf{E}_{i,k}^{i-1}$.

4.2.4. Partial Derivative of Tool Length Error. Letting $b_{T,j}$ denote the j^{th} element of b_T, the partial derivative of the tool length error is

$$\frac{\partial V}{\partial b_{T,j}} = -\sum_{k=1}^{N} \left(2 \boldsymbol{\xi}_k^T \boldsymbol{\Sigma}_{\xi}^{-1} \mathbf{W}_k \begin{bmatrix} \dfrac{\partial \mathbf{p}_{T,1}^n}{\partial b_{T,j}} & \dfrac{\partial \mathbf{p}_{T,2}^n}{\partial b_{T,j}} & \dfrac{\partial \mathbf{p}_{T,3}^n}{\partial b_{T,j}} \end{bmatrix}^T \right), \tag{44}$$

where

$$\mathbf{W}_k = \mathbf{T}_0^m \mathbf{F}_a \left(\mathbf{q}_k + \mathbf{v}_{k,t} \right), \; t = 1,2,3. \tag{45}$$

4.3. COMPENSATION

Compensation for the modeled robot errors can be accomplished using the inverse Jacobian method described in [22]. Letting $\mathbf{q}_{nom}$ represent the nominal joint command, a compensated joint command $\mathbf{q}_{new}$ is calculated such that

$$\mathbf{F}_a \left(\mathbf{q}_{new} \right) = \mathbf{F}_n \left(\mathbf{q}_{nom} \right). \tag{46}$$

The initial estimate of $\mathbf{q}_{new}$ is $\mathbf{q}^0 = \mathbf{q}_{nom}$. For the r^{th} estimate, $\mathbf{q}^r$, there exists a residual error transformation, $\boldsymbol{\Omega}^r$

$$\mathbf{\Omega}^r = \left[\mathbf{F}_a\left(\mathbf{q}^r\right)\right]^{-1}\mathbf{F}_n\left(\mathbf{q}_{nom}\right) = \begin{bmatrix} \omega_{11}^r & \omega_{12}^r & \omega_{13}^r & \omega_{14}^r \\ \omega_{21}^r & \omega_{22}^r & \omega_{23}^r & \omega_{24}^r \\ \omega_{31}^r & \omega_{32}^r & \omega_{33}^r & \omega_{34}^r \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{47}$$

Based on (47), an error vector is computed as

$$d\mathbf{e}^r = \left[\begin{array}{ccc|ccc} \omega_{14}^r & \omega_{24}^r & \omega_{34}^r & \frac{1}{2}\left(\omega_{32}^r - \omega_{23}^r\right) & \frac{1}{2}\left(\omega_{13}^r - \omega_{31}^r\right) & \frac{1}{2}\left(\omega_{21}^r - \omega_{12}^r\right) \end{array}\right]^T, \tag{48}$$

where the left and right partitions, respectively, are the position and orientation errors between nominal and actual transformations. This position and orientation error in Cartesian space can be transformed to the joint command error by

$$d\mathbf{q}^r = \mathbf{J}\left(\mathbf{q}_{nom}\right)^{-1} d\mathbf{e}^r, \tag{49}$$

where $d\mathbf{q}^r$ is the corresponding joint command error for $d\mathbf{e}^r$ and $\mathbf{J}\left(\mathbf{q}_{nom}\right)$ is the Jacobian matrix. Then, the updated joint command is

$$\mathbf{q}^{r+1} = \mathbf{q}^r + d\mathbf{q}^r. \tag{50}$$

Equations (47)-(50) are repeated until a suitable tolerance is satisfied

$$\left\| d\mathbf{q}^r \right\|_2 < tol, \tag{51}$$

where *tol* is the stopping tolerance which, for example, can be set to be the joint encoder resolution. When the tolerance is satisfied, the compensated joint command is

$$\mathbf{q}_{new} = \mathbf{q}^f. \tag{52}$$

where $\mathbf{q}^f$ is the final estimated joint command.

5. EXPERIMENTAL RESULTS

5.1. EXPERIMENTAL SETUP

A FANUC LR Mate 200*i* robot with a RJ3 controller is used for the experimental studies conducted in this paper. A photograph of the robot is shown in Figure 8 and a schematic of its kinematic structure is shown in Figure 9. The DH parameters for the FANUC LR Mate 200*i* are listed in Table 2.

Figure 8. Photograph of FANUC LR Mate 200*i* robot side view with links lengths.

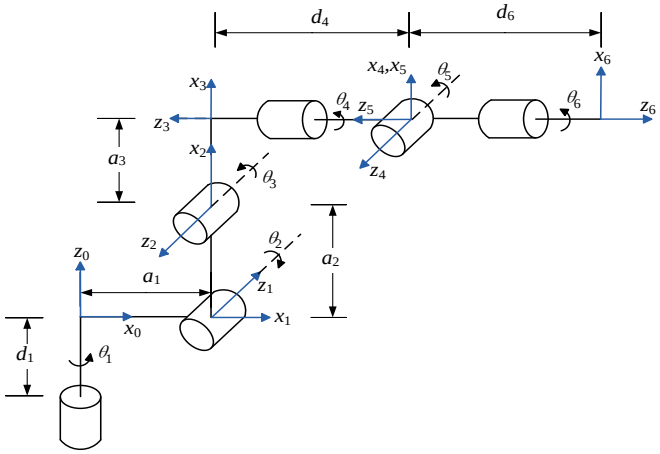

Figure 9. Schematic of FANUC LR Mate 200*i* kinematic structure with joint frames and rotation directions.

Table 2. FANUC LR Mate 200*i* DH parameters.

Frame	θ_i **(deg)**	d_i **(mm)**	a_i **(mm)**	α_i **(deg)**
1	q_1	0	150	-90
2	$q_2 - 90$	0	250	180
3	$q_2 + q_3$	0	75	-90
4	q_4	-290	0	90
5	q_5	0	0	-90
6	q_6	-80	0	180

5.2. MEASUREMENT COLLECTION

Measurements are acquired using an Automated Precision, Inc. model R-20 Radian laser tracker and model AT1 Active Target spherical tool. Manufacturer specifications list the standard deviation of the laser tracker measurement as σ = 2.5 μm/m. The following experiments are conducted with a measurement range of 1.5 to 2.5 m, depending on the position and orientation of the robot. Variation in accuracy over the 1 m range is small enough to be neglected. Therefore, the accuracy of the laser tracker at the average distance of 2 m is used. Further, standard deviation along the measurement frame in the *X*, *Y* and *Z* axes are assumed to be the same and independent. Thus, a standard deviation of σ_m=5×10^{-3} mm is used. The manufacturer specifications for the spherical tool report an accuracy of ±12.5 μm, which is treated as deterministic. Thus, all the experimental data and model results in following sections are treated to have a minimum error of ±12.5 μm.

Measurements are acquired to fully cover the joint space where the tool tip is visible to the laser tracker. Table 3 lists the minimum and maximum limits of the measured space. Across this space, measurements are generated with a quasi-random distribution. Quasi-random numbers, such as the Niederreiter sequence, have low discrepancy, meaning that there are smaller gaps and less clustering of the measurement locations [26] than sequences generated with a pseudo random sequence, especially when the sequence sample size is small. Table 4 gives the estimated tool length vectors described in Figures 6 and 7.

Table 3. Measured range of each joint.

	Joint 1	**Joint 2**	**Joint 3**	**Joint 4**	**Joint 5**	**Joint 6**
Minimum	-100º	-30º	-30º	-180º	-90º	-180º
Maximum	100º	100º	100º	180º	90º	180º

Table 4. Estimated tool length vectors of three measurement tool configurations.

Orientation	**1**	**2**	**3**
l_x (mm)	33.06	57.26	-33.96
l_y (mm)	57.73	-33.86	-57.23
l_z (mm)	161.22	161.22	161.22

5.3. REPEATABILITY

A robot's repeatability is a fundamental limitation of how well the robot can be calibrated. The ISO 9283 standard provides a technical procedure to determine the

repeatability of industrial robots [27]. Following the procedure in the ISO 9283 standard, the repeatability of the FANUC LR Mate 200*i* is determined to be ±0.03 mm.

5.4. ESTIMATION OF JOINT VARIANCE

Joint variance, $\sigma^2_{Ji,t}$, describes the variation of the positioning repeatability when only joint *i* is commanded to the same position repeatedly. In this work, the variance of the normal distribution fitted to the angular errors between each measurement and the measurement's center is taken to be the joint variance. For the i^{th} joint, *M* positions are measured, each of which is measured twice from the same direction of approach. Within the rotation plane, let two measurements for the j^{th} position be denoted ($x_{i,j1}$, $y_{i,j1}$) and ($x_{i,j2}$, $y_{i,j2}$). The center measurement is

$$\left(x_{i,ja}, y_{i,ja}\right) = \left(\frac{x_{i,j1} + x_{i,j2}}{2}, \frac{y_{i,j1} + y_{i,j2}}{2}\right). \tag{53}$$

Thus, the angular errors $\theta_{i,j1}$ and $\theta_{i,j2}$, respectively, between the measurements and center is

$$\begin{aligned} \theta_{i,j1} &= \arctan\left(\frac{y_{i,j1}}{x_{i,j1}}\right) - \arctan\left(\frac{y_{i,ja}}{x_{i,ja}}\right) \\ \theta_{i,j2} &= \arctan\left(\frac{y_{i,j2}}{x_{i,j2}}\right) - \arctan\left(\frac{y_{i,ja}}{x_{i,ja}}\right) \end{aligned}. \tag{54}$$

A normal distribution is then fitted using all of the angular errors for the i^{th} joint. The variance of the fitted normal distribution is the i^{th} joint variance. The same procedure is repeated for the opposite direction of motion. Figure 10 shows the distributions of the fitted angular errors. Table 5 lists the estimated joint variances for each joint in both their forward and backward directions for $M = 9$.

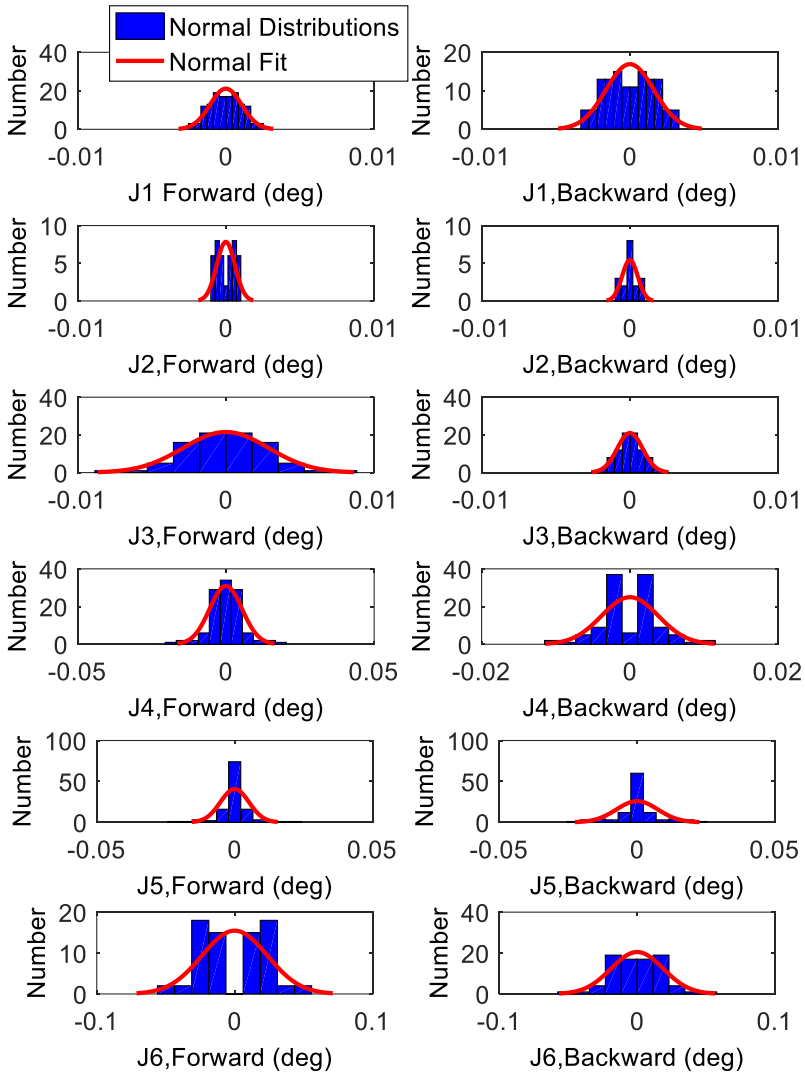

Figure 10. Normal distributions of angular errors for forward and backward motions of each joint.

5.5. MODEL IDENTIFICATION

One issue in model identification is the determination of the order of polynomials that is appropriate to describe the kinematic error functions. Lower order sequences will fail to sufficiently describe the complexity of the complex kinematic errors (e.g., strain wave gearing errors), while higher order sequences will tend to over fit the identification

data set, leading to poor estimation outside of this set. Another issue is the number of measurements that should be collected. If not enough measurements are taken, the kinematic error model parameters cannot be properly estimated; however, taking too many measurements will decrease the calibration procedure efficiency. Note that modeling higher order kinematic errors will necessitate the collection of additional identification data. To determine the appropriate order of the Chebyshev polynomials and the proper number of measurements to take, models are constructed using sequences from 1^{st} to 10^{th} order with 150, 250 and 350 quasi-random measurements as the identification data set. A separate set of 300 quasi-random measurements are collected as a validation set, which is used for validating the model performance. Figure 11 shows the mean residuals for the identification (ID) and validation (Val) sets with different order polynomials and number of identification measurements (ID Meas).

From Figure 11, one can see that while identification set residuals decrease with increasing polynomial order, the same is not true for the validation set residuals. High order models with small data sets have a higher validation residual compared to low order models with the same data set, and indication that the model is over fitting the data set (e.g. 9^{th} order polynomial models compared to 3^{rd} order polynomial models using 150 identification measurements). The polynomial order and identification set are selected to achieve the smallest mean residual in the validation set. For the data set in Fig. 11, a 6^{th} order model with 350 measurements achieves that lowest residual. However, the validation residual for 250 measurements provides nearly the same performance (only 0.006 mm higher residual), while at the same time using significantly less measurement data. Thus, 250 measurements with a 6^{th} order polynomial are selected and used in the following analysis. Note that these

results will likely depend on the specific robot under investigation and a similar analysis will be required to determine the appropriate number of measurements and polynomial order for other robots.

Table 5. Estimated joint variances for FANUC LR Mate 200*i*.

Joint	**Forward (deg^2)**	**Backward (deg^2)**
1	3.6×10^{-6}	2.6×10^{-6}
2	3.6×10^{-7}	2.5×10^{-7}
3	8.4×10^{-6}	2.0×10^{-6}
4	3.0×10^{-5}	1.4×10^{-5}
5	2.7×10^{-5}	5.6×10^{-5}
6	5.6×10^{-4}	3.6×10^{-4}

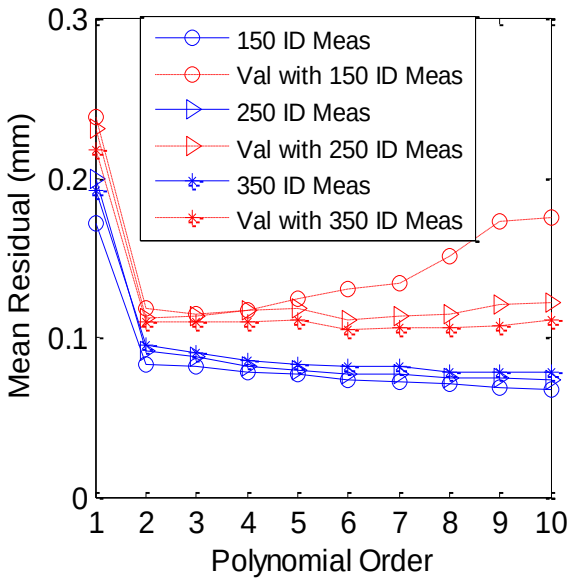

Figure 11. Mean residuals for identification and validation data sets with different number of measurements and error model polynomial orders.

The final model is constructed using three data sets, one for each measurement tool configuration, of 250 measurements for each set using a sequence of 6^{th} order Chebyshev polynomials. Another 300 measurements for the third tool configuration are used as the validation set. Using the identified model error parameters, the joint-dependent error terms are calculated using the identification procedure described above. Figure 12 shows the rotational (ε_X, ε_Y and ε_Z) and translational (δ_X, δ_Y and δ_Z) joint-dependent errors for all six joints. Both rotational and translational errors show joint-dependency. The variation of ε_X and ε_Y, especially for Joints 1 and 2, indicates an obvious joint-dependent out of plane error.

Another rotational error, ε_Z, has a constant offset from zero for all six joints, a result of mastering errors. The joint-dependent components in ε_Z can be described by pitch and strain wave gearing errors. Those two error sources also cause joint-dependent backlash errors, indicating a joint-dependent, direction-dependent in plane error. The translational errors are a combination of rotating center and link length errors. The joint-dependency of translational errors is due to the eccentricity in the strain wave gearings. From the combination of rotational and translational errors, not only do the rotating axes have in plane and out plane joint-dependent errors, the actual rotational centers are also wobbling around their nominal center positions.

Since it is a challenge to find the true maximum residual throughout the measured space and the 300 validation points may not contain that point, a Gamma distribution is fitted to the model residuals. The residual at 99% of the Gamma distribution, which is termed here as the Gamma 99% residual, is used as another measure to show the performance of the validation set. Figure 13 shows the definitions of the mean residual, maximum residual and Gamma 99% residual for the validation data set. Table 6 and

Figures 14 and 15 show the performance of the constructed model. Table 6 lists the residuals for the identification and validation data sets. Figures 14 and 15 show the nominal and modeled residuals for the identification and validation data sets, respectively. The mean residual is reduced 97.8% for the identification set and 97.0% for the validation set.

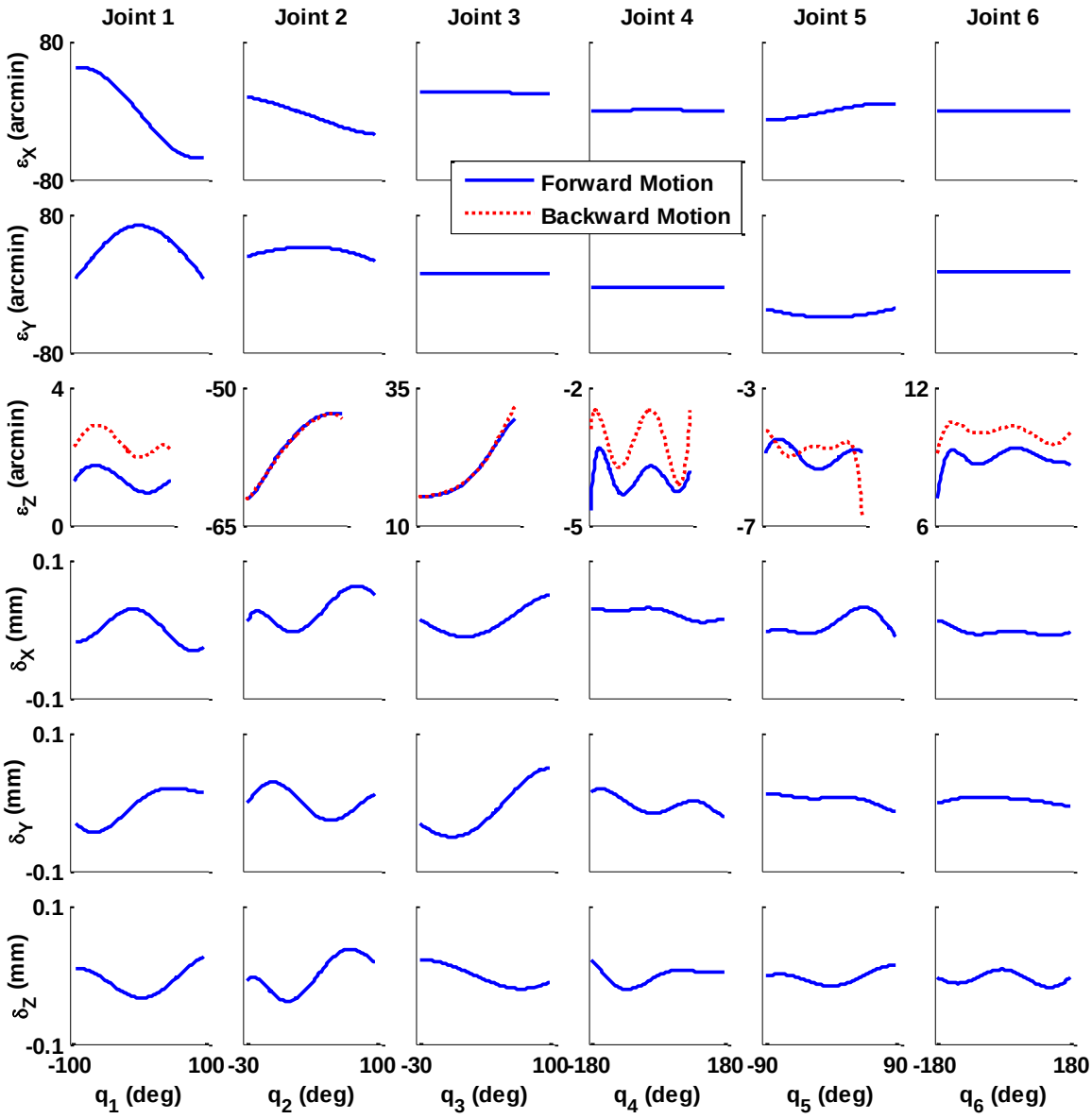

Figure 12. Rotational and translational kinematic error model terms for each joint.

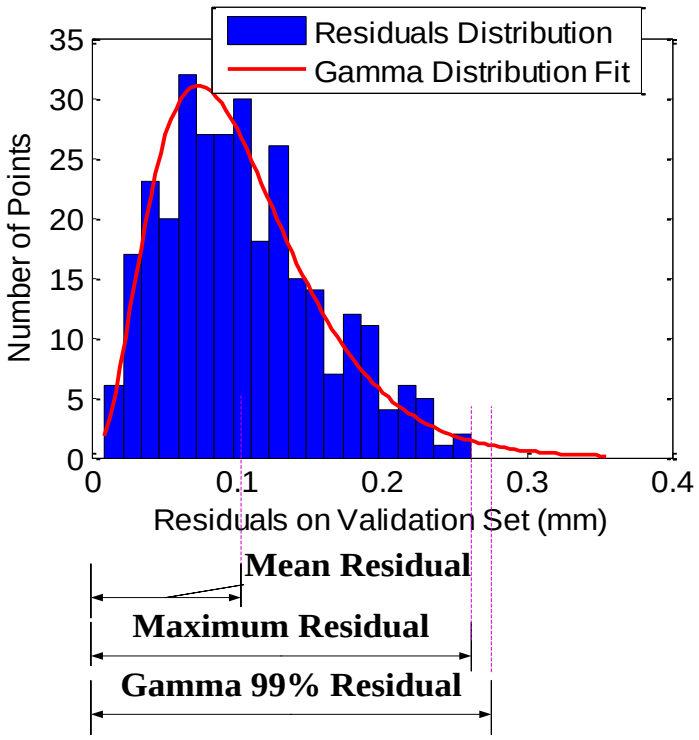

Figure 13. Definitions of mean residual, maximum residual and Gamma 99% residual for validation data set.

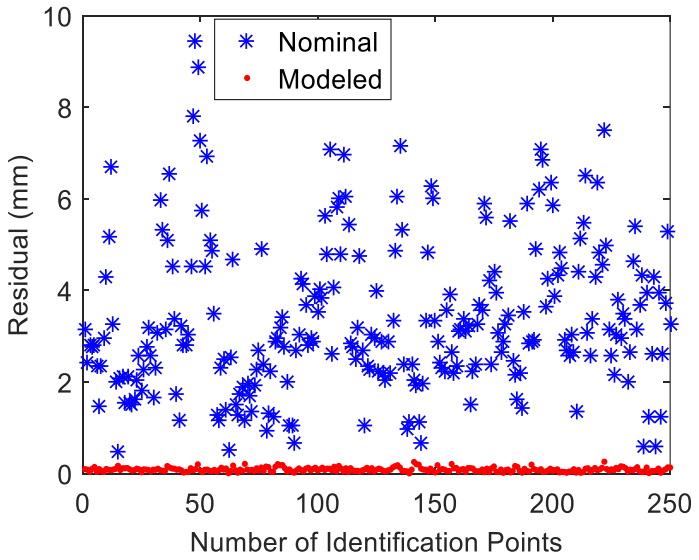

Figure 14. Nominal and modeled residuals for identification data set.

Table 6. Mean, maximum and Gamma 99% residuals for identification and validation data sets.

	Mean		**Maximum**		**Gamma 99%**	
Residual	Value (mm)	Percent Decrease	Value (mm)	Percent Decrease	Value (mm)	Percent Decrease
Identification Set Nominal	3.360	N/A	9.458	N/A	N/A	N/A
Identification Set Modeled	0.074	97.80%	0.216	97.72%	N/A	N/A
Validation Set Nominal	3.418	N/A	8.210	N/A	8.721	N/A
Validation Set Modeled	0.104	96.96%	0.261	96.82%	0.277	96.82%

5.6. COMPENSATION

Compensation is implemented for the 300 points in the validation set. New data are collected using the compensated commands and the residuals between the actual and nominal positions are calculated. Table 7 and Figure 16 show the compensation results for the validation set. The percentage decrease is 96.96% for the mean modeled residual and 97.02% for the mean compensated residual with only 0.06% difference, showing that compensated performance is very consistent with the model results.

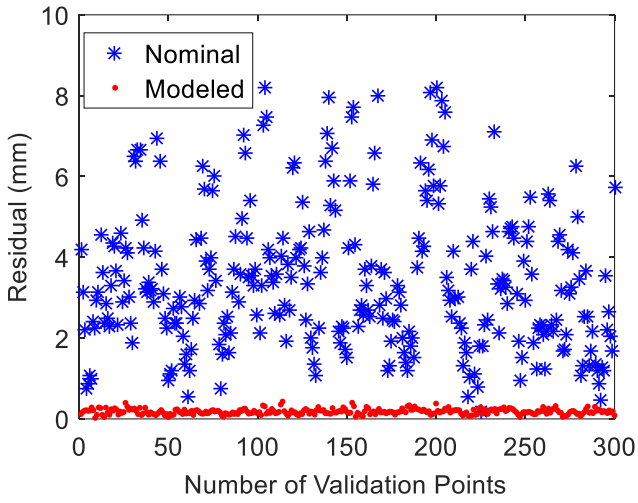

Figure 15. Nominal and modeled residuals for validation data set.

Table 7. Mean, maximum and Gamma 99% nominal, modeled and compensated residuals for validation data set.

	Mean		**Maximum**		**Gamma 99%**	
Residual	Value (mm)	Percent Decrease	Value (mm)	Percent Decrease	Value (mm)	Percent Decrease
Nominal	3.418	N/A	8.210	N/A	8.721	N/A
Modeled	0.104	96.96%	0.261	96.82%	0.277	96.82%
Compensated	0.102	97.02%	0.287	96.50%	0.264	97.00%

6. EFFECTS OF JOINT-DEPENDENT ERRORS

In order to evaluate the magnitude of joint-dependent errors as compared to traditional joint-independent kinematic errors, an alternative calibration methodology for fixed-parameters errors (rotating center, mastering, etc.) is performed.

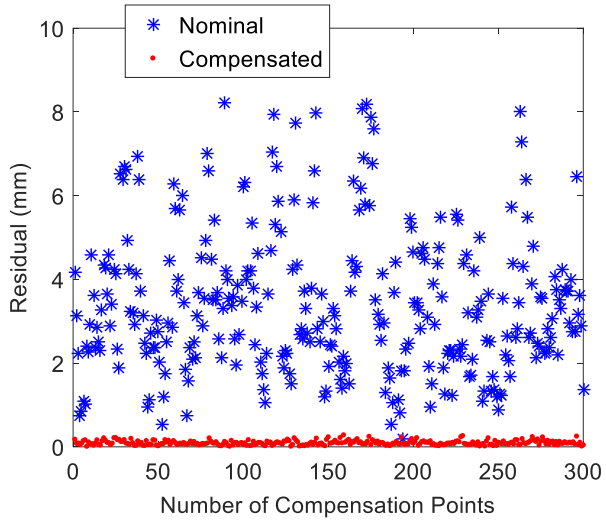

Figure 16. Nominal and compensated residuals for validation data set.

6.1. CIRCLE POINT ANALYSIS

Circle Point Analysis (CPA) is a common calibration method that models fixed kinematic errors. The basic idea of CPA is to determine the actual rotating axis of a joint and measure the kinematic parameters defining the transformation between consecutive joints. In CPA, a circle can be fitted using measurements collected for a single rotating joint. The rotation axis will pass the fitted circle's center and be normal to the circle plane. By constructing new frames for all of the joints' rotation axes, the actual kinematic transformation parameters can be determined between consecutive frames. Following previous work done in [13,14], CPA is implemented on the industrial robot considered in this study.

Measurements are taken for the motion of each joint separately with evenly distributed joint commands. Table 8 lists the measured range, angle step and number of measurements for the CPA identification set. Six circles are fitted and actual frames are constructed. The actual kinematic transformation parameters (i.e., DH and Hayati-

Mirmirani conventions parameters) are calculated and listed in Table 9. With the CPA identified parameters, nominal and modeled residuals are calculated for the CPA identification set and the results are shown in Table 10.

Table 8. Measured range, angle step and number of points in CPA identification data set.

	Joint 1 (deg)	**Joint 2 (deg)**	**Joint 3 (deg)**	**Joint 4 (deg)**	**Joint 5 (deg)**	**Joint 6 (deg)**
Start	-100	-30	-30	-180	-90	-180
End	98	50	100	180	90	180
Angle Step	3	2	2	5	3	5
Points	67	41	66	72	61	72

6.2. CALIBRATION METHODOLOGY COMPARISON AND ANALYSIS

A comparison between the CPA and proposed industrial robot calibration methodologies is conducted for two validation data sets. The first validation set consists of 60 quasi-random points on each circle used in the CPA methodology, which is denoted the CPA validation set. The second validation set, which spans the entire visible joint range, is the validation set of 300 points used above to analyze the performance of the proposed methodology, and is denoted the quasi-random validation set. Table 11 shows the mean and maximum residuals for the CPA validation set, and Figure 17 shows the nominal, CPA and proposed methodology modeled residuals for the CPA validation set. The residuals for the quasi-random validation set for both calibration methodologies are shown in Table 12

Table 9. Nominal and CPA modeled parameters of FANUC LR Mate 200*i*.

Parameter	**Nominal**	**Modeled**
$\boldsymbol{\theta_1}$ **(deg)**	q_1	q_1-0.022
$\boldsymbol{\theta_2}$ **(deg)**	q_2– 90	q_2-90.055
$\boldsymbol{\theta_3}$ **(deg)**	q_3	q_3+0.381
$\boldsymbol{\theta_4}$ **(deg)**	q_4	q_4-0.537
$\boldsymbol{\theta_5}$ **(deg)**	q_5	q_5-0.320
$\boldsymbol{d_1}$ **(mm)**	0	-0.061
$\boldsymbol{d_3}$ **(mm)**	0	0.133
$\boldsymbol{d_4}$ **(mm)**	-290	-290.444
$\boldsymbol{d_5}$ **(mm)**	0	-0.050
$\boldsymbol{a_1}$ **(mm)**	150	149.72
$\boldsymbol{a_2}$ **(mm)**	250	250.058
$\boldsymbol{a_3}$ **(mm)**	75	75.177
$\boldsymbol{a_4}$ **(mm)**	0	0.136
$\boldsymbol{a_5}$ **(mm)**	0	-0.106
$\boldsymbol{\alpha_1}$ **(deg)**	-90	-90.060
$\boldsymbol{\alpha_2}$ **(deg)**	-180	-179.990
$\boldsymbol{\alpha_3}$ **(deg)**	-90	-89.992
$\boldsymbol{\alpha_4}$ **(deg)**	90	89.986
$\boldsymbol{\alpha_5}$ **(deg)**	-90	-90.027
$\boldsymbol{\beta_2}$ **(deg)**	0	-0.008

Table 10. Mean and maximum nominal and CPA modeled residuals for CPA identification data set.

	Nominal Residual (mm)		**CPA Modeled Residual (mm)**	
Joint	**Mean**	**Maximum**	**Mean**	**Maximum**
1	2.571	2.611	0.176	0.329
2	2.506	3.445	0.356	0.807
3	3.488	5.200	0.790	2.380
4	4.493	6.129	0.039	0.076
5	2.841	3.157	0.103	0.262
6	3.003	3.524	0.035	0.043

Table 11. Mean and maximum nominal, CPA and proposed methodology modeled residuals for CPA validation data set.

	Nominal Residual (mm)		**CPA Modeled Residual (mm)**		**Proposed Method Modeled Residual (mm)**	
Joint	**Mean**	**Maximum**	**Mean**	**Maximum**	**Mean**	**Max**
1	2.601	3.641	0.181	0.296	0.074	0.167
2	2.513	3.456	0.350	0.822	0.071	0.294
3	3.389	5.073	0.692	2.212	0.098	0.189
4	4.419	6.121	0.039	0.072	0.082	0.154
5	2.830	3.161	0.102	0.273	0.096	0.355
6	2.981	3.528	0.033	0.043	0.084	0.122

and Figure 18. The CPA methodology has similar performance for the data in the CPA validation data set as it did for the CPA identification data set (see Table 10). However, unlike the CPA methodology, the proposed calibration methodology is able to reduce the residuals on Joints 2 and 3 to the same level as other joints. For the quasi-random validation data set, the proposed method also shows better performance and captures nearly 20% more errors than the CPA methodology regarding the mean residual.

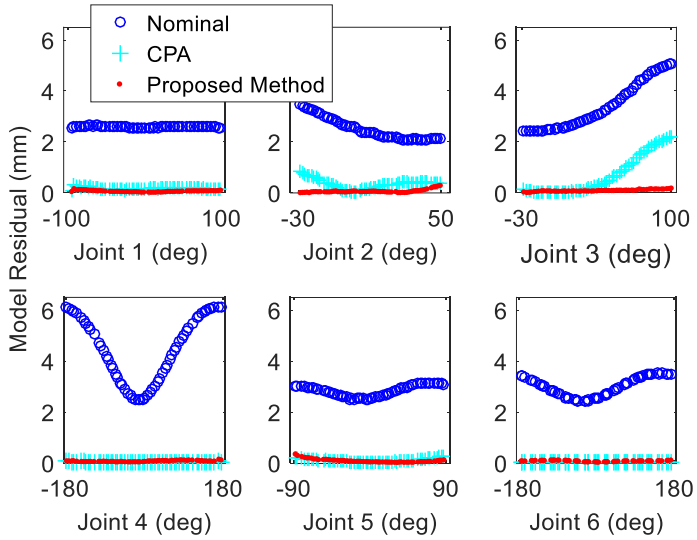

Figure 17. Nominal, CPA and proposed methodology modeled residuals for CPA validation data set.

Table 12. Mean, maximum and Gamma 99% nominal, CPA and proposed methodology modeled residual for the quasi-random validation data set (mm).

	Mean Residual	**Maximum Residual**	**Gamma 99%**
Nominal	3.418	8.210	8.721
CPA Method	0.705	2.061	1.875
Proposed Method	0.104	0.261	0.277

Significant differences exist between the model residuals for the CPA and proposed methods for Joints 2 and 3 (see Figure 17). To analyze these differences, the results for Joint 3 are investigated in detail. Figure 19 gives a plot of the differences between the CPA modeled and measured positions in 3-D space and within the third joint's rotating 2-D

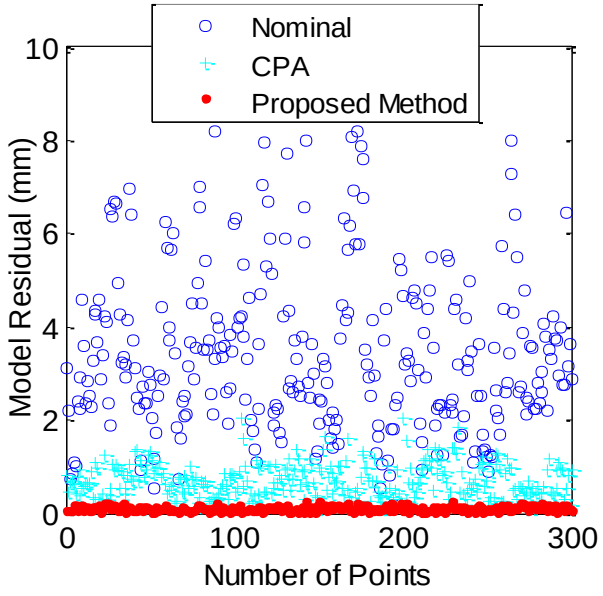

Figure 18. Nominal, CPA and proposed method modeled residuals for the quasi-random validation data set.

plane for the CPA validation set. The data shows that the differences between the CPA modeled and measured positions in 3-D and 2-D spaces are nearly the same, indicating that almost all of the Joint 3 residuals are due to joint-dependent errors within the rotary plane.

The full error transformation for Joint 3 is

$$\mathbf{E}_3^2(q_3,s_3)\approx\begin{bmatrix} 1 & -\varepsilon_Z(q_3,s_3) & \varepsilon_Y(q_3) & \delta_X(q_3) \\ \varepsilon_Z(q_3,s_3) & 1 & -\varepsilon_X(q_3) & \delta_Y(q_3) \\ -\varepsilon_Y(q_3) & \varepsilon_X(q_3) & 1 & \delta_Z(q_3) \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{55}$$

where all of the error terms are described by 6th order Chebyshev polynomials. Now the proposed method is modified by retaining only the constant term of ε_Z. Figure 20 shows the nominal, CPA, full proposed methodology and modified proposed methodology modeled residuals for the CPA validation data set for Joint 3. Retaining only the constant terms of ε_Z, the performance of this modified proposed methodology is nearly the same as the CPA performance. This phenomenon is consistent with Figure 19 which shows that the in plane joint-dependent errors limit the CPA model performance. Referring to the error sources given in Section 1, the joint-dependent error sources can be attributed to pitch and strain wave gearing errors. Thus, joint-dependent errors are an important component of kinematic errors in industrial robots, in this case accounting for nearly 20% of the error.

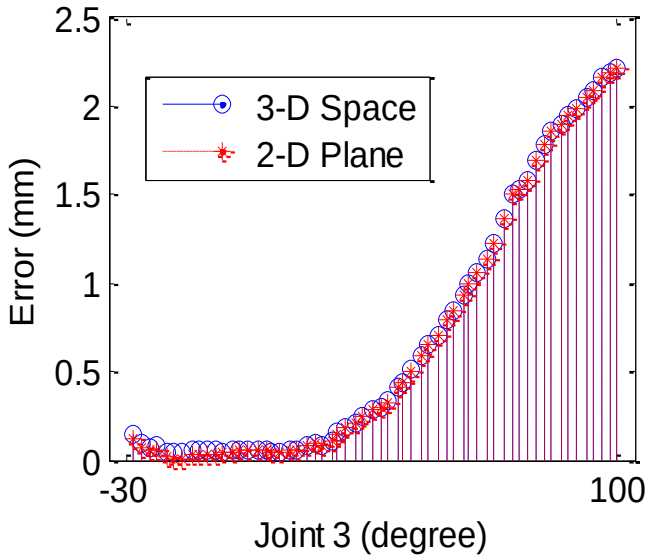

Figure 19. Differences between Joint 3 CPA modeled and measured positions in 3-D space and 2-D rotating plane.

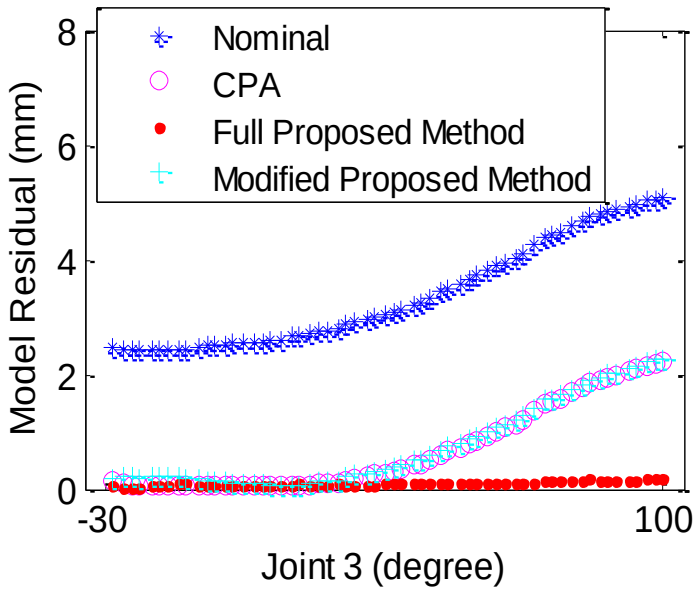

Figure 20. Nominal, CPA, full proposed methodology and modified proposed methodology modeled residuals for CPA validation data set.

7. SUMMARY AND CONCLUSIONS

Robot kinematic errors were classified in this paper into seven categories: rotating center offset, mastering, link length and assembly, pitch, strain wave gearing, out of plane, and backlash. Each error was discussed and it was seen that many errors are joint-dependent and better described by high-order models, as opposed to constant offsets. These kinematic errors were generalized by describing them with error matrices containing high-order Chebyshev polynomials to represent individual error terms. A new robot calibration and compensation methodology was presented. This methodology uses position data measured by a laser tracker from a measurement tool mounted in three different orientations on a robot. Using the robot nominal kinematics augmented with the generalized error matrices, a maximum likelihood estimator is used to simultaneously

estimate the model parameters, joint variances and measurement error for the collected data. A Jacobian-based search method is then used to determine updated joint commands to compensate for the robot's kinematic errors.

A series of experiments were conducted using a FANUC LR Mate 200*i* robot. Robot position measurements were taken with a laser tracker and the robot repeatability, joint variance and measurement variance were analyzed. A variety of number of measurements and polynomial orders for the error terms were analyzed and it was found that using 250 measurements and 6^{th} order polynomials produced accurate models using an efficient process. Applying the calibration procedure, the resulting robot kinematic error model reduced the errors seen in the identification data set by 97.8% and reduced the errors seen in a validation data set by 97%. Further, there was a reduction of 97% in the kinematic errors when applying compensation to the validation data set.

Circle Point Analysis (CPA), a common calibration methodology, was applied to the individual joints. Using separate measurement data sets collected for individual joints, the CPA method was able to account for over 93% of the errors for Joints 1, 4, 5 and 6; however, it could only account for 85.8% and 77.4% of the errors for Joints 2 and 3, respectively. For separate validation data sets collected for each joint, the proposed method was able to reduce the mean errors of all six joints by at least 97%, while the CPA method could only reduce the errors of Joints 2 and 3, respectively, by 86% and 80%. A careful analysis of Joint 3 showed that it had significant in-plane, joint-dependent errors that could be described by pitch and strain wave gearing errors. Also, when the CPA method was applied to the validation data taken over the entire joint space the proposed method was able to capture nearly 20% more of the kinematic errors than the CPA method. Thus, this

paper demonstrated the importance of joint-dependent robot kinematic errors and provided a calibration and compensation methodology capable of describing and eliminating these errors.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support for this work from the National Science Foundation (Award No. CMMI 1335340), the US Department of Education (Award No. P200A120062), Textron/Bell Helicopter, and the Center for Aerospace Manufacturing Technologies at the Missouri University of Science and Technology.

APPENDIX

The robot Jacobian relates velocities in joint space to the end-effector linear and angular velocities in Cartesian space. For an n-link robot with joint commands $\mathbf{q} = [q_1, \ldots, q_n]^T$, let $\mathbf{p}_{end}^{0,nom}$ denote the nominal end-effector position with respect to the robot base frame. In this case the linear velocity portion of the Jacobian $\mathbf{J}_v^{nom}$ is

$$\mathbf{J}_v^{nom} = \nabla_{\mathbf{q}} \mathbf{p}_{end}^{0,nom}. \tag{56}$$

The modeled linear velocity portion of the Jacobian is computed by replacing the nominal end-effector position with the modeled end-effector position,

$$\mathbf{J}_v^{mod} = \nabla_{\mathbf{q}} \mathbf{p}_{end}^{0,mod}, \tag{57}$$

where $\mathbf{J}_v^{mod}$ is the modeled linear velocity portion of the Jacobian and $\mathbf{p}_{end}^{0,mod}$ is the modeled position of the end-effector with respect to the robot base frame.

Let an error transformation matrix be

$$\mathbf{E}_i^{i-1} = \begin{bmatrix} \mathbf{R}_{E,i}^{i-1} & \mathbf{p}_{E,i}^{i-1} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} = \begin{bmatrix} 1 & -\varepsilon_Z(q_i, s_i) & \varepsilon_Y(q_i) & \delta_X(q_i) \\ \varepsilon_Z(q_i, s_i) & 1 & -\varepsilon_X(q_i) & \delta_Y(q_i) \\ -\varepsilon_Y(q_i) & \varepsilon_X(q_i) & 1 & \delta_Z(q_i) \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{58}$$

where $\mathbf{R}_{E,i}^{i-1}$ is a general rotational error matrix and $\mathbf{p}_{E,i}^{i-1}$ is a general translational error vector from Frame *i*-1 to Frame *i*. Let a nominal transformation from Frame *i*-1 to Frame *i* be

$$\mathbf{T}_i^{i-1} = \begin{bmatrix} \mathbf{R}_i^{i-1} & \mathbf{p}_i^{i-1} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{59}$$

where $\mathbf{R}_i^{i-1}$ is a nominal rotation matrix and $\mathbf{p}_i^{i-1}$ is a nominal translation vector. Thus, the model transformation from Frame *i*-1 to Frame *i* is

$$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{E}_i^{i-1}\mathbf{T}_i^{i-1} = \begin{bmatrix} \mathbf{R}_{E,i}^{i-1}\mathbf{R}_i^{i-1} & \mathbf{R}_{E,i}^{i-1}\mathbf{p}_i^{i-1} + \mathbf{p}_{E,i}^{i-1} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}. \tag{60}$$

The model rotation and translation from Frame *i*-1 to Frame *i*, respectively, are

$$\begin{aligned} \hat{\mathbf{R}}_i^{i-1} &= \mathbf{R}_{E,i}^{i-1}\mathbf{R}_i^{i-1} \\ \hat{\mathbf{p}}_i^{i-1} &= \mathbf{R}_{E,i}^{i-1}\mathbf{p}_i^{i-1} + \mathbf{p}_{E,i}^{i-1} \end{aligned}, \tag{61}$$

Thus, the model transformation from the robot base frame to Frame *n*, is

$$\mathbf{F}_a(\mathbf{q}) = \prod_{i=1}^{n} \mathbf{E}_i^{i-1}\mathbf{T}_i^{i-1} = \begin{bmatrix} \hat{\mathbf{R}}_n^0 & \hat{\mathbf{p}}_n^0 \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{62}$$

where

$$\begin{aligned}\hat{\mathbf{R}}_n^0 &= \prod_{i=1}^{n} \hat{\mathbf{R}}_i^{i-1} \\ \hat{\mathbf{p}}_n^0 &= \left(\prod_{i=1}^{n-1} \hat{\mathbf{R}}_i^{i-1}\right)\hat{\mathbf{p}}_n^{n-1} + \left(\prod_{i=1}^{n-2} \hat{\mathbf{R}}_i^{i-1}\right)\hat{\mathbf{p}}_{n-1}^{n-2} + \cdots + \hat{\mathbf{p}}_1^0\end{aligned}, \tag{63}$$

The tool length vector, which is a translation from Frame *n* to the end-effector, is

$$\mathbf{p}_T^n = \begin{bmatrix} \hat{\mathbf{p}}_{end}^n \\ 0 \end{bmatrix}, \tag{64}$$

where $\hat{\mathbf{p}}_{end}^n$ is the XYZ position of the end-effector with respect to Frame *n*. Thus, the modeled position of the end-effector is

$$\mathbf{p}_{end}^{0,mod} = \hat{\mathbf{R}}_n^0 \hat{\mathbf{p}}_{end}^n + \hat{\mathbf{p}}_n^0 = \left(\prod_{i=1}^{n} \hat{\mathbf{R}}_i^{i-1}\right)\hat{\mathbf{p}}_{end}^n + \left(\prod_{i=1}^{n-1} \hat{\mathbf{R}}_i^{i-1}\right)\hat{\mathbf{p}}_n^{n-1} + \left(\prod_{i=1}^{n-2} \hat{\mathbf{R}}_i^{i-1}\right)\hat{\mathbf{p}}_{n-1}^{n-2} + \cdots + \hat{\mathbf{p}}_1^0 . \tag{65}$$

Using Equation (57), the i^{th} column of the linear velocity portion of the modeled Jacobian, $\mathbf{J}_{v,i}^{mod}$ is found by taking the partial derivative of the model position with respect to the i^{th} joint command,

$$\mathbf{J}_{v,i}^{mod} = \frac{\partial \mathbf{p}_{end}^{0,mod}}{\partial q_i} = \frac{\partial\left(\left(\prod_{i=1}^{n} \hat{\mathbf{R}}_i^{i-1}\right)\hat{\mathbf{p}}_{end}^n + \left(\prod_{i=1}^{n-1} \hat{\mathbf{R}}_i^{i-1}\right)\hat{\mathbf{p}}_n^{n-1} + \left(\prod_{i=1}^{n-2} \hat{\mathbf{R}}_i^{i-1}\right)\hat{\mathbf{p}}_{n-1}^{n-2} + \cdots + \hat{\mathbf{p}}_1^0\right)}{\partial q_i} . \tag{66}$$

Expanding Equation (66)

$$\begin{aligned}\mathbf{J}_{v,i}^{mod} &= \frac{\partial\left(\hat{\mathbf{R}}_{i-1}^0 \hat{\mathbf{R}}_i^{i-1} \hat{\mathbf{R}}_n^i\right)\hat{\mathbf{p}}_{end}^n}{\partial q_i} + \frac{\partial\left(\hat{\mathbf{R}}_{i-1}^0 \hat{\mathbf{R}}_i^{i-1} \hat{\mathbf{R}}_{n-1}^i\right)\hat{\mathbf{p}}_n^{n-1}}{\partial q_i} + \frac{\partial\left(\hat{\mathbf{R}}_{i-1}^0 \hat{\mathbf{R}}_i^{i-1} \hat{\mathbf{R}}_{n-2}^i\right)\hat{\mathbf{p}}_{n-1}^{n-2}}{\partial q_i} + \cdots + \frac{\partial \hat{\mathbf{R}}_{i-1}^0 \hat{\mathbf{p}}_i^{i-1}}{\partial q_i} \\ &= \hat{\mathbf{R}}_{i-1}^0 \left(\frac{\partial\left(\hat{\mathbf{R}}_i^{i-1}\right)}{\partial q_i}\left(\hat{\mathbf{R}}_n^i \hat{\mathbf{p}}_{end}^n + \hat{\mathbf{R}}_{n-1}^i \hat{\mathbf{p}}_n^{n-1} + \hat{\mathbf{R}}_{n-2}^i \hat{\mathbf{p}}_{n-1}^{n-2} + \cdots + \hat{\mathbf{p}}_i^{i+1}\right) + \frac{\partial \hat{\mathbf{p}}_i^{i-1}}{\partial q_i} \right) \\ &= \hat{\mathbf{R}}_{i-1}^0 \left(\frac{\partial\left(\hat{\mathbf{R}}_i^{i-1}\right)}{\partial q_i} \mathbf{p}_{end}^{i,mod} + \frac{\partial \hat{\mathbf{p}}_i^{i-1}}{\partial q_i} \right)\end{aligned}, \tag{67}$$

where $\mathbf{p}_{end}^{i,mod}$ is the model position of the end-effector with respect to Frame *i* and

$$\frac{\partial\left(\hat{\mathbf{R}}_i^{i-1}\right)}{\partial q_i}=\frac{\partial\left(\mathbf{R}_{E,i}^{i-1}\mathbf{R}_i^{i-1}\right)}{\partial q_i}=\frac{\partial\mathbf{R}_{E,i}^{i-1}}{\partial q_i}\mathbf{R}_i^{i-1}+\mathbf{R}_{E,i}^{i-1}\frac{\partial\mathbf{R}_i^{i-1}}{\partial q_i}, \tag{68}$$

$$\frac{\partial\hat{\mathbf{p}}_i^{i-1}}{\partial q_i}=\frac{\partial\left(\mathbf{R}_{E,i}^{i-1}\mathbf{p}_i^{i-1}+\mathbf{p}_{E,i}^{i-1}\right)}{\partial q_i}=\frac{\partial\mathbf{R}_{E,i}^{i-1}}{\partial q_i}\mathbf{p}_i^{i-1}+\mathbf{R}_{E,i}^{i-1}\frac{\partial\mathbf{p}_i^{i-1}}{\partial q_i}+\frac{\partial\mathbf{p}_{E,i}^{i-1}}{\partial q_i}. \tag{69}$$

There are four unknown terms in Equations (68) and (69), $\frac{\partial\mathbf{R}_{E,i}^{i-1}}{\partial q_i}$, $\frac{\partial\mathbf{R}_i^{i-1}}{\partial q_i}$, $\frac{\partial\mathbf{p}_i^{i-1}}{\partial q_i}$ and $\frac{\partial\mathbf{p}_{E,i}^{i-1}}{\partial q_i}$.

Taking the derivatives of the rotation and translation errors,

$$\frac{\partial\mathbf{R}_{E,i}^{i-1}}{\partial q_i}=\begin{bmatrix}0 & -\frac{d\varepsilon_Z\left(q_i,s_i\right)}{dq_i} & \frac{d\varepsilon_Y\left(q_i\right)}{dq_i}\\ \frac{d\varepsilon_Z\left(q_i,s_i\right)}{dq_i} & 0 & -\frac{d\varepsilon_X\left(q_i\right)}{dq_i}\\ -\frac{d\varepsilon_Y\left(q_i\right)}{dq_i} & \frac{d\varepsilon_X\left(q_i\right)}{dq_i} & 0\end{bmatrix}=\mathbf{S}\left(\boldsymbol{\varepsilon}_i'\right), \tag{70}$$

$$\frac{\partial\mathbf{p}_{E,i}^{i-1}}{\partial q_i}=\begin{bmatrix}\frac{d\delta_X\left(q_i\right)}{dq_i} & \frac{d\delta_Y\left(q_i\right)}{dq_i} & \frac{d\delta_Z\left(q_i\right)}{dq_i}\end{bmatrix}^T. \tag{71}$$

where **S** is a skew symmetric matrix and

$$\boldsymbol{\varepsilon}_i'=\begin{bmatrix}\frac{d\varepsilon_X\left(q_i\right)}{dq_i} & \frac{d\varepsilon_Y\left(q_i\right)}{dq_i} & \frac{d\varepsilon_Z\left(q_i\right)}{dq_i}\end{bmatrix}^T. \tag{72}$$

For revolute joints, the translation vector for the i^{th} joint is

$$\mathbf{p}_i^{i-1}=\mathbf{R}_i^{i-1}\mathbf{L}_i^{i-1}, \tag{73}$$

where $\mathbf{L}_i^{i-1}$ is a fixed link length vector. Letting Z_{i-1} denote the axis of rotation of Frame *i*-1,

$$\frac{\partial\mathbf{R}_i^{i-1}}{\partial q_i}=\mathbf{S}\left(\mathbf{k}\right)\mathbf{R}_i^{i-1}, \tag{74}$$

where

$$\mathbf{k} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T . \tag{75}$$

The derivative of the translation vector is

$$\frac{\partial \mathbf{p}_i^{i-1}}{\partial q_i} = \frac{\partial \mathbf{R}_i^{i-1} \mathbf{L}_i^{i-1}}{\partial q_i} = \mathbf{S}(\mathbf{k}) \mathbf{R}_i^{i-1} \mathbf{L}_i^{i-1} = \mathbf{S}(\mathbf{k}) \mathbf{p}_i^{i-1} = \mathbf{k} \times \mathbf{p}_i^{i-1} . \tag{76}$$

Combining (70), (71), (74) and (76) with (68) and (69), the i^{th} column of the linear velocity portion of model Jacobian is

$$\mathbf{J}_{v,i}^{mod} = \hat{\mathbf{R}}_{i-1}^{0} \left(\mathbf{R}_{E,i}^{i-1} \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \left(\mathbf{R}_i^{i-1} \mathbf{p}_{end}^{i,mod} + \mathbf{p}_i^{i-1} \right) \right) + \begin{bmatrix} \frac{d\delta_X(q_i)}{dq_i} \\ \frac{d\delta_Y(q_i)}{dq_i} \\ \frac{d\delta_Z(q_i)}{dq_i} \end{bmatrix} \times \left(\mathbf{R}_i^{i-1} \mathbf{p}_{end}^{i,mod} + \mathbf{p}_i^{i-1} \right) + \begin{bmatrix} \frac{d\delta_X(q_i)}{dq_i} \\ \frac{d\delta_Y(q_i)}{dq_i} \\ \frac{d\delta_Z(q_i)}{dq_i} \end{bmatrix} \right) . \tag{77}$$

REFERENCES

[1] J.F. Engelberger, Robotics in practice: Management and applications of industrial robots. Springer Science and Business Media, 2012.

[2] M.H. Ang Jr, L. Wei, L.S. Yong, An industrial application of control of dynamic behavior of robots-a walk-through programmed welding robot. Proceedings of the 2000 IEEE International Conference on Robotics and Automation, 2000, 3:2352-2357.

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

[4] H. Kochekali, A. Nowrouzi, Y.B. Kavina, R.A. Whitaker, Factors affecting robot performance. Industrial Robot, 1991, 18(1):9-13.

[5] Y.H.A. Liou, P.P. Lin, R.R. Lindeke, H.D. Chiang, Tolerance specification of robot kinematic parameters using an experimental design technique—the Taguchi method. Robotics and Computer-Integrated Manufacturing, 1993, 10(3):199-207.

[6] D.E. Whitney, C.A. Lozinski, J.M. Rourke, Industrial robot forward calibration method and results. ASME Journal of Dynamic Systems, Measurement, and Control, 1986, 108(1):1-8.

[7] B. Karan, M. Vukobratovic, Calibration and accuracy of manipulation robot models—an overview. Mechanism and Machine Theory, 1994, 29(3):479-500.

[8] J.M. Hollerbach, C.W. Wampler, The calibration index and taxonomy for robot kinematic calibration methods. International Journal of Robotics Research, 1996, 15(6):573-591.

[9] A. Nubiola, I.A. Bonev, Absolute calibration of an ABB IRB 1600 robot using a laser tracker. Robotics and Computer-Integrated Manufacturing, 2013, 29(1):236-245.

[10] A. Nubiola, M. Slamani, I.A. Bonev, A new method for measuring a large set of poses with a single telescoping ballbar. Precision Engineering, 2013, 37(2):451-460.

[11] D.J. Bennett, S. Member, J.M. Hollerbach, Autonomous calibration of single-loop closed kinematic chains formed by manipulators with passive endpoint constraints. IEEE Transactions on Robotics and Automation, 1991, 7(5):597-606.

[12] M.E. Sklar, Geometric calibration of industrial manipulators by circle point analysis. Proceedings of the 2nd Conference on Recent Advances in Robotics, 1989, 178-202.

[13] J. Santolaria, J. Conte, M. Pueo, C. Javierre, Rotation error modeling and identification for robot kinematic calibration by circle point method. Metrology and Measurement System, 2014, 21(1):85-98.

[14] J. Santolaria, J. Conte, M. Ginés, Laser tracker-based kinematic parameter calibration of industrial robots by improved CPA method and active retroreflector. International Journal of Advanced Manufacturing Technology, 2013, 66:2087-2106.

[15] L.M. Hsia, The analysis and design of harmonic gear drives. Proceedings of the 1988 IEEE International Conference on Systems, Man, and Cybernetics, 1988, 1:616-619.

[16] M. Were, Analysis and control of kinematic error in harmonic gear drive mechanisms. Rice University, 1997.

[17] T.W. Nye, R.P. Kraml, Harmonic drive gear error: characterization and compensation for precision pointing and tracking. In 25th Aerospace Mechanisms Symposium, 1991, 3113:237.

[18] P.M. Sammons, L. Ma, K. Embry, L.H. Armstrong, D.A. Bristow, R.G. Landers, Modeling and compensation of backlash and harmonic drive-induced errors in robotic manipulators. ASME 2014 International Manufacturing Science and Engineering Conference, 2014.

[19] F.H. Ghorbel, P.S. Gandhi, F. Alpeter, On the kinematic error in harmonic drive gears. ASME Journal of Mechanical Design, 2001, 123(1):90-97.

[20] P. Freeman, A novel means of software compensation for robots and machine tools. SAE Technical Paper, 2006, 2006-01-3167.

[21] C.W. Wampler, J.M. Hollerbach, T. Arai, An implicit loop method for kinematic calibration and its application to closed-chain mechanisms. Robotics and Automation, 1995, 11(5):710-724.

[22] B.W. Mooring, Z.S. Roth, M.R. Driels, Fundamentals of manipulator calibration. New York, Wiley, 1991.

[23] M.W. Spong, S. Hutchinson, M. Vidyasagar, Robot modeling and control. New York, Wiley, 2006.

[24] J.Y. Kao, Z.M. Yeh, Y.S. Tarng, Y.S. Lin, A study of backlash on the motion accuracy of CNC lathes. International Journal of Machine Tools and Manufacture, 1996, 36(5):539-550.

[25] J. Creamer, P.M. Sammons, D.A. Bristow, R.G. Landers, P.L. Freeman, S.J. Easley, Table-based volumetric error compensation of large five-axis machine tools. Journal of Manufacturing Science and Engineering, 2017, 139(2):021011.

[26] H. Niederreiter, Low-discrepancy and low-dispersion sequences. Journal of Number Theory, 1988, 30(1):51-70.

[27] ISO 9283, Manipulating industrial robots-Performance criterion and related test methods. International Standardization Organization, 1998.

III. ONLINE ADAPTIVE MODELING OF ROBOTIC KINEMATIC ERRORS USING A SIX DEGREE OF FREEDOM TRACKING SENSOR

Le Ma, Douglas A. Bristow and Robert G. Landers

ABSTRACT

Improving robot's positioning accuracy is important for many industrial applications. Measurements at different robot poses are often collected and utilized for the purpose of error identification and compensation. Typically, a laser tracker and retroreflectors are used for data collection by attaching the retroreflectors to the robot end effector. However, most retroreflectors can only measure the robot position information. To determine robot's orientation information, each robot pose has to be measured multiple times with placing and measuring the retroreflector at different position on the robot end effector. This process dramatically increases the measurement time. Extra fixturing errors, robot repeatability errors and measurement errors are also introduced each time, which lowers the error model and thus the compensation accuracy. In this paper, a six degree of freedom tracking sensor, which is capable of measuring robot position and orientation information simultaneously, is introduced and utilized. A comparison between using the new sensor and position retroreflector methods is made. In addition, traditionally, models with different number of identification measurements have to be constructed and compared to find a smallest but sufficient number of points for an accurate model construction, which also causes inefficiency. In this paper, an online adaptive model identification method is proposed which iteratively updates model parameters with newly collected measurements.

Experiments are implemented on a FANUC LR Mate 200*i* robot. Using the adaptive modeling method, 76% modeling time has been saved to find the most appropriate number of identification measurements comparing with the traditional method. For the new tracking sensor, while the traditional device uses 4 hours to reduce the robot mean positional and angular residual to 0.136 mm and 0.0007 rad, respectively, the proposed sensor is able to reduce the mean error to 0.118 mm and 0.00045 rad with only 1.5 hours.
Key words: Industrial robots; Laser tracker; Retroreflector; Tracking sensor; Adaptive model identification

1. INTRODUCTION

Industrial robots are widely used in highly repeatable tasks such as palletizing and packaging with advantages of low cost and high repeatability. In recent years, industrial robots also start to play more important roles in manufacturing applications such as deburring and light machining, which require the robot to have a more sufficient accuracy [1]. However, robots have kinematic errors that often make their accuracy unacceptable for such tasks. Thus, improving the robot accuracy is essential. Generally, there are four steps to calibrate robot kinematic errors [2]. First, a robot kinematic error model is constructed by mathematically describing the actual geometry and motion of the robot. Second, robots are commanded to different poses while measurement devices are used to measure robot positions and orientations at those poses. Third, the kinematic error parameters introduced in the first step are identified by matching the robot kinematic error model with the measurements and the last step is to utilize the error model for compensation.

Of the four steps, the second step, i.e., measurement, is critical important as the data accuracy will directly affect the error model accuracy. Further, the speed of collecting measurements will also determine how efficient the whole calibration process will be. To achieve a rapid and accurate calibration, a measurement device which can collect data rapidly and accurately is expected. Developments in precision instrumentation have provided wide options for robot metrology. Coordinate Measuring Machines (CMM), telescoping ball-bars, and camera-based systems have been successfully used in robot calibration [3-8]. Also, laser trackers, having the advantages of single setup, large measurement range and rapid data collection, are becoming widely adopted for data collection [9-13]. When a laser tracker is utilized for measurement, retroflectors such as Spherical Mounted Retroreflectors (SMRs) or Active Targets (ATs) are often attached to the robot end effector for position measurement. However, these retroflectors cannot measure orientation information. Multiple laser trackers measuring one retroflector simultaneously, or one laser tracker measuring each robot pose multiple times, where a retroreflector is located at a different position on the end effector for each pose are needed to obtain the orientation information. In [14], multiple SMRs are attached to the robot end effector. In [15], three AT configurations at each robot pose are measured. This dramatically reduces the efficiency and accuracy of the calibration process. Extra fixturing and measurement errors are introduced each time the retroreflector or the AT is reattached and measured. So, to improve the calibration efficiency and accuracy, a new device with the capability of determining robot position and orientation information simultaneously is needed. In this paper, a device with such capability, called SmartTRACK Sensor (STS), is introduced and utilized for robot calibration.

STS is a six Degree of Freedom (DoF) measurement device that is designed based upon AT. By incorporating two rotary encoders and a level sensor on the AT, the new device is capable of determining its position and orientation simultaneously. Laser tracker measures position information and the two encoders and level sensor measures orientation information. As measurement devices, both laser tracker and STS bring measurement errors. To consider all fixed robot errors and random measurement errors, this paper proposes a kinematic error model that treats the whole measurement system as a closed loop and considers both fixed robot kinematic errors and stochastic errors in the system, e.g., robot repeatability errors, laser tracker positioning errors and STS encoder and level sensor errors. For the three parts in the measurement system, i.e., robot, laser tracker and STS, a high-order joint-dependent kinematic error model is utilized. Then, by describing the nominal kinematics of laser tracker and STS, the measured position, encoders and level sensor angles are determined and modeled with the utilized robot kinematic error model. Also, the stochastic positioning errors from laser tracker and the stochastic angular errors from the two encoders and level sensor are characterized.

Another challenge in the measurement step is to determine a least number of measurements which is enough for an accurate model construction. Traditionally, models with different number of measurements have to be constructed [15]. By comparing the modeling accuracy of a same validation set between different models, the most appropriate number of measurements which takes less time but gives desired model accuracy is determined. However, measurements have to be collected first and then models can be constructed. Extra data and unnecessary models reduce the measurement and modeling efficiency. In this paper, an online adaptive modeling method is proposed. Using a model

constructed with the first dozens of measurements, when a few measurements are collected, this initial model can be adaptively updated with less time comparing with constructing a new model. This enables the measurement and modeling step to be parallelly implemented. The whole calibration efficiency is thus improved.

The rest of this paper is organized as follows. Section 2 introduces the background of a high-order joint-dependent kinematic error model for robot. Section 3 describes the nominal kinematics for laser tracker and STS, and then proposes a position and orientation model for the whole measurement system. An online adaptive model identification method and the characterization of position and orientation measurement errors are provided in Section 4. Section 5 provides the experimental results for a FANUC LR Mate 200*i* robot. A method with AT measurements is also implemented. A comparison between AT measurement method and the proposed sensor measurement method is described and analyzed in Section 6. The paper is summarized and conclusions are drawn in Section 7.

2. ROBOT KINEMATIC ERROR MODEL BACKGROUND

Robots have different sources of errors that will cause difference between actual and nominal kinematics. As summarized in [15], a robot suffers seven sources of simple and complicated kinematic errors including rotating center offset errors, mastering errors, link length and assembly errors, pitch errors, strain wave gear errors, out of plane and backlash errors. In this section, the kinematic error model proposed in [15] that compensates all those simple and complicated errors will be introduced and used as the fundamental work in this paper.

Let $\mathbf{T}_i^{i-1}$ represent a transformation from Frame *i*-1 to Frame *i* and parameterize $\mathbf{T}_i^{i-1}$ according to the Denavit–Hartenberg (DH) convention [16] as

$$\mathbf{T}_i^{i-1} = \mathbf{Rot}_z(\theta_i)\mathbf{Trans}_z(d_i)\mathbf{Trans}_x(a_i)\mathbf{Rot}_x(\alpha_i), \tag{1}$$

where $\mathbf{Rot}_j$ is the rotation matrix about the j^{th} axis, $\mathbf{Trans}_j$ is the translation matrix along the j^{th} axis and θ_i, d_i, α_i and a_i are the four DH parameters for the i^{th} joint. The nominal kinematics for an *n*-joint robot can thus be represented as

$$\mathbf{F}_n(\mathbf{q}) = \mathbf{T}_1^0(q_1)\mathbf{T}_2^1(q_2)\ldots\mathbf{T}_n^{n-1}(q_n), \tag{2}$$

where $\mathbf{F}_n$ is the nominal transformation from the robot's base frame to robot's last frame and $\mathbf{q} = [q_1, q_2, \ldots, q_n]^T$ is the joint command vector.

A six Degree of Freedom (DoF) joint-dependent error matrix [17] is used to describe robot simple and complicated error sources,

$$\mathbf{E}(q_i) = \begin{bmatrix} 1 & -\varepsilon_Z(q_i) & \varepsilon_Y(q_i) & \delta_X(q_i) \\ \varepsilon_Z(q_i) & 1 & -\varepsilon_X(q_i) & \delta_Y(q_i) \\ -\varepsilon_Y(q_i) & \varepsilon_X(q_i) & 1 & \delta_Z(q_i) \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

where ε_X, ε_Y and ε_Z are small rotations about the *x*, *y* and *z* axesnof Frame *i*-1 and δ_X, δ_Y and δ_Z are small translations along the *x*, *y* and *z* axes of Frame *i*-1. Incorporating (3) into the nominal transformation matrix for each joint, the modeled transformation from Frame *i*-1 to Frame *i* is

$$\tilde{\mathbf{T}}_i^{i-1}(q_i) = \mathbf{E}_i^{i-1}(q_i)\mathbf{T}_i^{i-1}(q_i). \tag{4}$$

Thus, the modeled transformation from the robot's base frame to the robot's last frame is

$$\mathbf{F}_a(\mathbf{q}) = \tilde{\mathbf{T}}_1^0(q_1)\tilde{\mathbf{T}}_2^1(q_2)\ldots\tilde{\mathbf{T}}_n^{n-1}(q_n). \tag{5}$$

To capture both simple and complicated robot errors, Chebyshev polynomials, as a set of basis functions, are used to describe each rotational and translational errors given in (3). The constant terms can be regarded as descriptions of the simple robot errors rotating offset error, mastering error, link length and assembly errors and the other terms are descriptions for the complicated errors such as in plane and out of plane errors. Given a parameter λ in the interval [-1 1], a Chebyshev polynomial is

$$C(\lambda) = b_0 c_0(\lambda) + b_1 c_1(\lambda) + \cdots + b_m c_m(\lambda), \tag{6}$$

where

$$c_0(\lambda) = 1, c_1(\lambda) = \lambda, \ldots, c_{m+1}(\lambda) = 2\lambda c_m(\lambda) - c_{m-1}(\lambda), \tag{7}$$

b_0, b_1, b_2, …, b_m are the polynomial coefficients and m is the polynomial order. The joint-dependent rotational and translational errors can thus be described as

$$\begin{aligned} \varepsilon_j(\bar{q}_i) &= b_{0,ij,\varepsilon} + b_{1,ij,\varepsilon} c_1(\bar{q}_i) + \cdots + b_{m,ij,\varepsilon} c_m(\bar{q}_i) \\ \delta_j(\bar{q}_i) &= b_{0,ij,\delta} + b_{1,ij,\delta} c_1(\bar{q}_i) + \cdots + b_{m,ij,\delta} c_m(\bar{q}_i) \end{aligned}, \tag{8}$$

where j denotes the axis (i.e., x, y and z) and $\bar{q}$ denotes the i^{th} linearly mapped joint command by scaling the joint command range to [-1 1],

$$\bar{q}_i = \frac{2(q_i - q_{i,\min})}{(q_{i,\max} - q_{i,\min})} - 1, \tag{9}$$

and $q_{i,\min}$, $q_{i,\max}$ are the minimum and maximum i^{th} joint commands, respectively.

3. MEASUREMENT SYSTEM KINEMATIC ERROR MODEL

A laser tracker, with advantages of rapid data collection and the ability to measure robot workspace in one single setup, is used to collect robot's data. Unlike attaching multiple SMRs [14] to the robot end effector or measuring an Active Target [15] multiple times for the determination of robot's orientation information, a new tracking sensor, called SmatTRACK Sensor (STS), is used to measure robot's position and orientation information simultaneously. In this section, the working principle of the laser tracker and STS will be introduced and then a kinematic error model of the measurement system including the robot, laser tracker and STS will be developed.

3.1. LASER TRACKER

A laser tracker is an instrument that measures points in 3-D space rapidly. The laser tracker is driven by two independent rotation axes, denoted azimuth and elevation, which are measured by two rotary encoders. Figure 1 provides a picture of an API Radian laser tracker and its schematic structure.

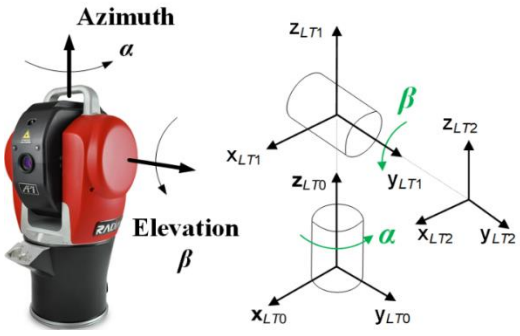

Figure 1. API Radian laser tracker with azimuth and elevation axes and its schematic structure.

In Figure 1, Frame $\mathbf{x}_{LT0}\mathbf{y}_{LT0}\mathbf{z}_{LT0}$ is the laser tracker base frame. Frames $\mathbf{x}_{LT1}\mathbf{y}_{LT1}\mathbf{z}_{LT1}$ and $\mathbf{x}_{LT2}\mathbf{y}_{LT2}\mathbf{z}_{LT2}$ are the resulting frames after the azimuth and elevation rotations, respectively. Unlike robot kinematic error model, where the nominal joint command vector is the input and known, the azimuth and elevation angles, α and β, can only be determined by treating the laser tracker having a perfectly nominal kinematics. So, error matrix could not be added to the laser tracker kinematics. The nominal laser tracker kinematics are

$$\begin{aligned}\mathbf{F}_{LT} &= \mathbf{Rot}_z(\alpha)\mathbf{Rot}_y(\beta) \\ &= \begin{bmatrix} \cos(\alpha)\cos(\beta) & -\sin(\alpha) & \cos(\alpha)\sin(\beta) & 0 \\ \sin(\alpha)\cos(\beta) & \cos(\alpha) & \sin(\alpha)\sin(\beta) & 0 \\ -\sin(\beta) & 0 & \cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.\end{aligned} \tag{10}$$

3.2. SMARTTRACK SENSOR

STS is designed and developed by equipping a retroreflector with two rotary encoders and a level sensor. Figure 2 gives a photograph of an STS.

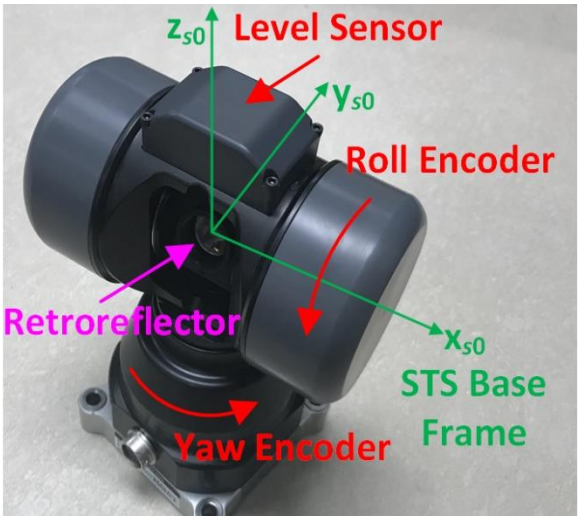

Figure 2. Photograph of SmartTRACK Sensor.

In Figure 2, $\mathbf{x}_{s0}\mathbf{y}_{s0}\mathbf{z}_{s0}$ is the STS base frame which is the determined orientation information. When an STS is connected with a laser tracker through the laser beam emitted from the laser tracker, the two independent encoders, i.e., the roll and yaw encoders, will automatically rotate as the STS moves, thereby adjusting the STS orientation to ensure the laser is always tracking the retroflector. Also, the level sensor, which is a microelectromechanical system (MEMS) accelerometer, will measure the inclination angle of the STS head with respect to the gravity vector. Then, the angle measurements from the encoders and level sensor and the position measurement from the retroreflector will be transferred to a Software Development Kit (SDK) to determine and output the location and orientation of Frame $\mathbf{x}_{s0}\mathbf{y}_{s0}\mathbf{z}_{s0}$. Thus, both position and orientation information are obtained simultaneously. Yaw and roll encoders have measurement ranges of [-180° 180°] and [-55° 55°], respectively. The level sensor has a measured range of [-60° 60°]. The limitation of the level sensor measured range is due to the fact that the accelerometer will become insensitive when the measured inclination is approaching the direction of the gravity vector [18]. A picture of an STS and its schematic structure is given in Figure 3.

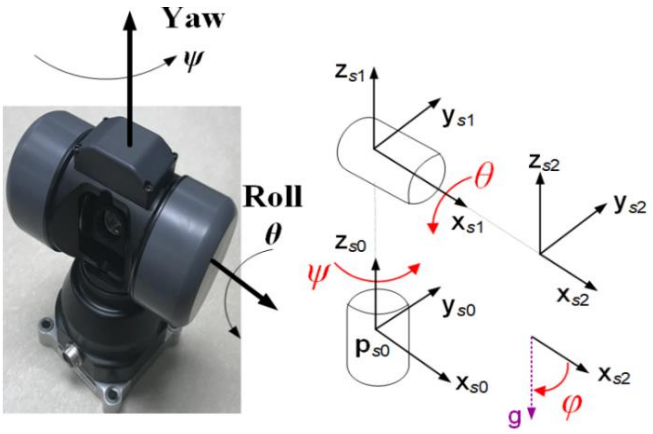

Figure 3. STS with yaw and roll axes and its schematic structure.

In Figure 3, Frame $\mathbf{x}_{s0}\mathbf{y}_{s0}\mathbf{z}_{s0}$ is the STS base frame. Frames $\mathbf{x}_{s1}\mathbf{y}_{s1}\mathbf{z}_{s1}$ and $\mathbf{x}_{s2}\mathbf{y}_{s2}\mathbf{z}_{s2}$ are the two frames resulting from yaw and roll rotations, respectively. Note the level sensor is assembled along axis $\mathbf{x}_{s2}$; therefore, the level sensor angle, φ, is the angle between axis $\mathbf{x}_{s2}$ and the gravity vector $\mathbf{g}$. Assuming there is no offset error caused by the two rotations and the two encoders are independent of one another, the nominal STS kinematics are

$$\begin{aligned}\mathbf{F}_{STS} &= \mathbf{Rot}_z(\psi)\mathbf{Rot}_x(\theta) \\ &= \begin{bmatrix} \cos(\psi) & -\sin(\psi)\cos(\theta) & \sin(\psi)\sin(\theta) & 0 \\ \sin(\psi) & \cos(\psi)\cos(\theta) & -\cos(\psi)\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.\end{aligned} \tag{11}$$

3.3. MEASUREMENT SYSTEM KINEMATICS

To use an STS and laser tracker for robot measurement collection, the STS is attached to the robot end effector and the laser tracker is placed in front of the robot. Figure 4 gives a setup of the measurement system including a robot, an STS and a laser tracker. The schematic structure of the closed measurement system is shown in Figure 5. In Figure 5, the measurement frame is an arbitrary frame that is placed close to the robot's base frame in software environment (e.g., New River Kinematics Spatial Analyzer). STS is attached to the robot, giving a fixed transformation from the robot's last frame to the STS base frame. Note that axis $\mathbf{z}_{LT2}$ and axis $\mathbf{y}_{s2}$ are parallel with the laser beam. Therefore, these two axes are collinear.

In the measurement software environment, the measured encoders and level sensor angles will not be saved. The STS measurement is exported as a transformation from the laser tracker base frame to the STS base from,

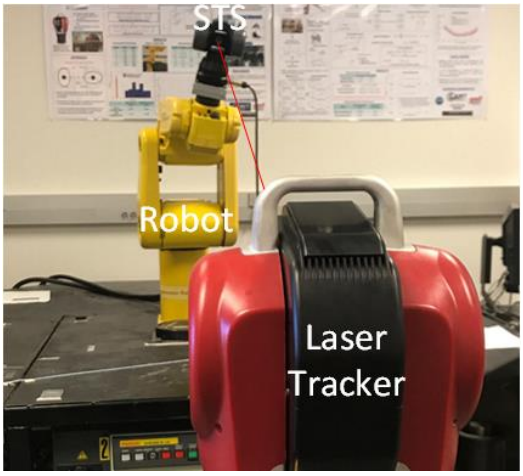

Figure 4. Measurement system consisting of a robot, STS and laser tracker.

$$\mathbf{T}_{s0}^{LT0} = \begin{bmatrix} \mathbf{R}_{s0}^{LT0} & \mathbf{p}_{s0}^{LT0} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{12}$$

where $\mathbf{R}_{s0}^{LT0}$ is the measured rotation from the laser tracker base frame to the STS base from and $\mathbf{p}_{s0}^{LT0}$ is the measured position from the laser tracker base frame to the STS base frame.

Using the robot kinematic error model, the modeled position from the laser tracker base frame to the STS base frame, $\hat{\mathbf{p}}_{s0}^{LT0}(\mathbf{q})$, is

$$\begin{bmatrix} \hat{\mathbf{p}}_{s0}^{LT0}(\mathbf{q}) \\ 1 \end{bmatrix} = \mathbf{T}_{m}^{LT0}\mathbf{E}_{0}^{m}\mathbf{F}_{a}(\mathbf{q})\hat{\mathbf{T}}_{s0}^{n}\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \tag{13}$$

where $\mathbf{T}_{m}^{LT0}$ is a known transformation from the laser tracker base frame to the measurement frame, $\mathbf{E}_{0}^{m}$ is a fixed transformation from the measurement frame to robot's base frame, $\hat{\mathbf{T}}_{s0}^{n}$ is a fixed transformation from robot's last frame to the STS base frame,

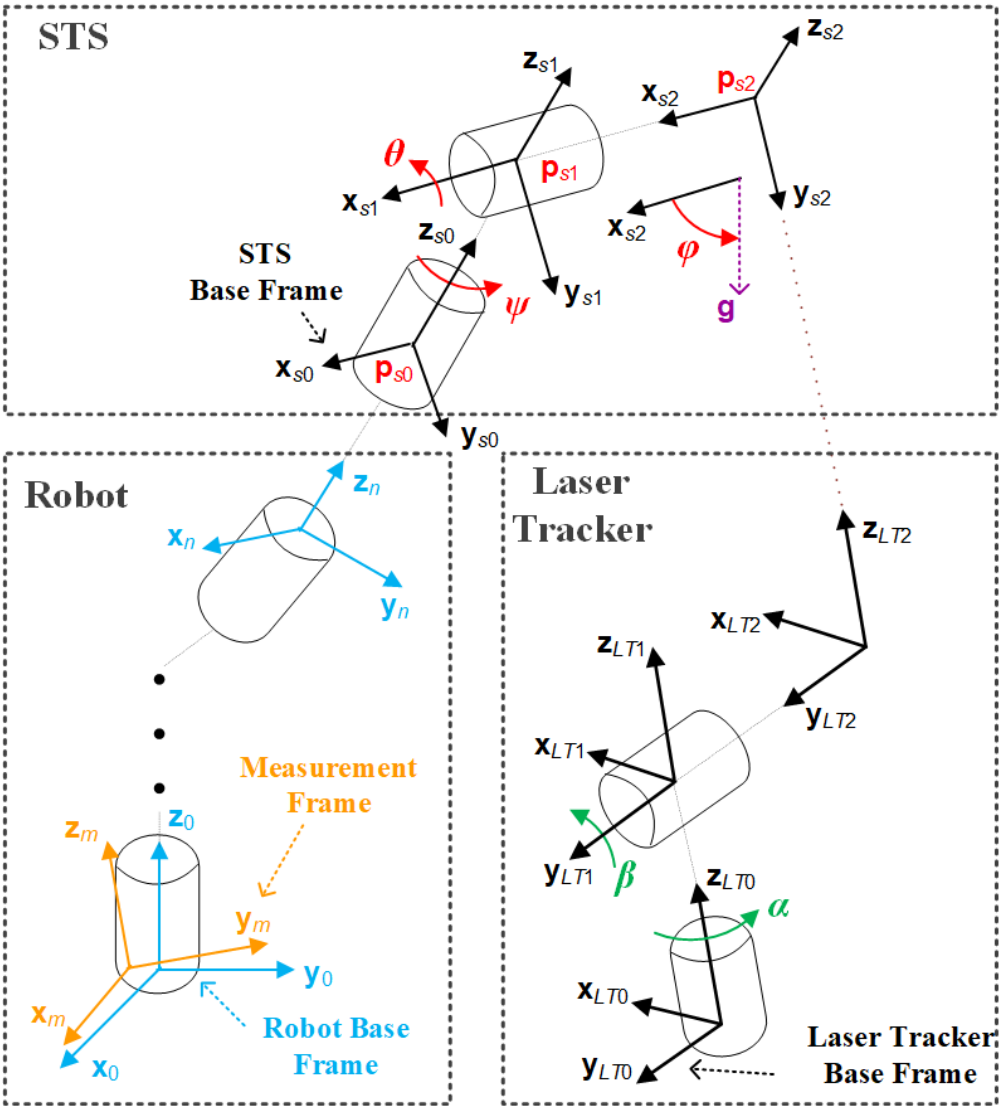

Figure 5. Schematic structure of measurement system.

$$\mathbf{T}_m^{LT0} = \begin{bmatrix} \mathbf{R}_m^{LT0} & \mathbf{p}_m^{LT0} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}, \tag{14}$$

$$\mathbf{E}_0^m = \begin{bmatrix} 1 & -\varepsilon_{Z,0} & \varepsilon_{Y,0} & \delta_{X,0} \\ \varepsilon_{Z,0} & 1 & -\varepsilon_{X,0} & \delta_{Y,0} \\ -\varepsilon_{Y,0} & \varepsilon_{X,0} & 1 & \delta_{Z,0} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{15}$$

$$\hat{\mathbf{T}}_{s0}^{n} = \mathbf{T}_{s0}^{n}\mathbf{E}_{s0}^{n} = \mathbf{T}_{s0}^{n}\begin{bmatrix} 1 & -\varepsilon_{Z,s0} & \varepsilon_{Y,s0} & \delta_{X,s0} \\ \varepsilon_{Z,s0} & 1 & -\varepsilon_{X,s0} & \delta_{Y,s0} \\ -\varepsilon_{Y,s0} & \varepsilon_{X,s0} & 1 & \delta_{Z,s0} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{16}$$

and $\mathbf{R}_{s0}^{LT0}$, $\mathbf{p}_{s0}^{LT0}$ are known the rotation and translation from the laser tracker base frame to the measurement frame, respectively, $\mathbf{T}_{s0}^{n}$ is a measured transformation from the robot's last frame to the STS base frame and $\mathbf{E}_{s0}^{n}$ is a fixed correction for $\mathbf{T}_{s0}^{n}$.

Since STS outputs its base frame orientation, (14) to (16) can also be used to directly describe the measured orientation information, i.e., $\mathbf{R}_{s0}^{LT0}$. However, there are two reasons that a better way of modeling the orientation information is needed. First, the laser tracker specification gives the accuracies of α, β and the accuracy of the measured distance between the laser tracker and the tracked position such that the measured position accuracy can be characterized. However, for STS, the specification doesn't give the accuracies of the encoders, level sensor and $\mathbf{R}_{s0}^{LT0}$. It remains to determine how accurate the measured orientation is and how to characterize the accuracy. Second, the STS orientation matrix is determined with the three independent measured angles, i.e., two encoder angels and one level sensor angle. Even if the accuracies of the three angles are known, the accuracy of the orientation matrix will be a combination of three angular errors, which gives a challenge to convert the accuracies of three independent angles to the accuracy of an orientation matrix. Thus, using the three angles, i.e., two encoder angels and one level sensor angle, which are ultimately the orientation error sources, as the orientation measurement information is a much easier way for orientation accuracy characterization. In this work, a method of inversing $\mathbf{R}_{s0}^{LT0}$ back to encoders and level sensor angles is developed. Thus, the

output measurement in (12) is transformed back to one position and three angles. Correspondingly, kinematic error models for describing the three angles with robot kinematic errors are constructed.

In Figure 5, assuming no center offset error is caused by the two encoder rotations, $\mathbf{p}_{s0}^{LT0} = \mathbf{p}_{s1}^{LT0} = \mathbf{p}_{s2}^{LT0}$. Since axis $\mathbf{z}_{LT2}$ and axis $\mathbf{y}_{s2}$ are collinear with the laser beam, axis $\mathbf{y}_{s2}$ expressed in STS base frame is

$$\mathbf{y}_{s2}^{s0} = -\mathbf{z}_{LT2}^{s0} = -\mathbf{R}_{LT0}^{s0}\mathbf{z}_{LT2}^{LT0} = -\left(\mathbf{R}_{s0}^{LT0}\right)^{-1}\frac{\mathbf{p}_{s0}^{LT0}}{\left\|\mathbf{p}_{s0}^{LT0}\right\|_2}. \tag{17}$$

From (11), $\mathbf{y}_{s2}^{s0}$ can be written as,

$$\mathbf{y}_{s2}^{s0} = \begin{bmatrix} -\sin(\psi)\cos(\theta) \\ \cos(\psi)\cos(\theta) \\ \sin(\theta) \end{bmatrix}. \tag{18}$$

Thus, from (17) and (18),

$$\begin{bmatrix} -\sin(\psi)\cos(\theta) \\ \cos(\psi)\cos(\theta) \\ \sin(\theta) \end{bmatrix} = -\left(\mathbf{R}_{s0}^{LT0}\right)^{-1}\frac{\mathbf{p}_{s0}^{LT0}}{\left\|\mathbf{p}_{s0}^{LT0}\right\|_2} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}. \tag{19}$$

Solving (19) for the two encoder angles

$$\begin{matrix} \psi = \operatorname{atan2}(-a_1, \quad a_2) \\ \theta = \operatorname{asin}(a_3) \end{matrix}. \tag{20}$$

With ψ known, the level sensor angle is

$$\varphi = \arccos\left(\mathbf{x}_{s2}^{LT0} \cdot \mathbf{g}^{LT0}\right), \tag{21}$$

where $\mathbf{g}^{LT0}$ is the gravity vector expressed in the laser tracker base frame, which is measured by a laser tracker function called “virtual level” and

$$\mathbf{x}_{s2}^{LT0} = \mathbf{R}_{s0}^{LT0}\mathbf{x}_{s2}^{s0} = \mathbf{R}_{s0}^{LT0}\begin{bmatrix}\cos(\psi)\\ \sin(\psi)\\ 0\end{bmatrix}. \tag{22}$$

To model the three angles, the measured rotation and position in (19) and (22) are replaced with modeled rotation and position. Corresponding to (19), the relationship between the modeled encoder angles and robot model is

$$\begin{bmatrix}-\sin(\hat{\psi}(\mathbf{q}))\cos(\hat{\theta}(\mathbf{q}))\\ \cos(\hat{\psi}(\mathbf{q}))\cos(\hat{\theta}(\mathbf{q}))\\ \sin(\hat{\theta}(\mathbf{q}))\end{bmatrix} = -\left(\hat{\mathbf{R}}_{s0}^{LT0}(\mathbf{q})\right)^{-1}\frac{\hat{\mathbf{p}}_{s0}^{LT0}(\mathbf{q})}{\left\|\hat{\mathbf{p}}_{s0}^{LT0}(\mathbf{q})\right\|_2} = \begin{bmatrix}\hat{a}_1\\ \hat{a}_2\\ \hat{a}_3\end{bmatrix}. \tag{23}$$

Thus, the modeled yaw and roll encoder angles are

$$\begin{aligned}\hat{\psi}(\mathbf{q}) &= \operatorname{atan2}(-\hat{a}_1, \quad \hat{a}_2)\\ \hat{\theta}(\mathbf{q}) &= \operatorname{asin}(\hat{a}_3)\end{aligned}. \tag{24}$$

Corresponding to (21) and (22), the modeled level sensor angle is

$$\hat{\varphi}(\mathbf{q}) = \arccos\left(\hat{\mathbf{x}}_{s2}^{LT0}(\mathbf{q})\cdot\hat{\mathbf{g}}^{LT0}\right), \tag{25}$$

where

$$\hat{\mathbf{x}}_{s2}^{LT0} = \hat{\mathbf{R}}_{s0}^{LT0}(\mathbf{q})\begin{bmatrix}\cos(\hat{\psi}(\mathbf{q}))\\ \sin(\hat{\psi}(\mathbf{q}))\\ 0\end{bmatrix}, \tag{26}$$

and $\hat{\mathbf{g}}^{LT0}$ is the modeled gravity vector in laser tracker base frame,

$$\hat{\mathbf{g}}^{LT0} = \begin{bmatrix}g_x & g_y & \sqrt{1-g_x^2-g_y^2}\end{bmatrix}^T, \tag{27}$$

and g_x, g_y are the gravity vector elements projected on the x and y axis of laser tracker base frame, respectively.

4. ONLINE ADAPTIVE MODEL IDENTIFICATION AND MEASUREMENT ERROR CHARACTERIZATION

4.1. ONLINE ADAPTIVE MODEL IDENTIFICATION

Let $\mathbf{e}(\mathbf{q})$ be the difference between the measured and modeled positions and orientations,

$$\mathbf{e}(\mathbf{q}) = \mathbf{w}(\mathbf{q}) - \hat{\mathbf{w}}(\mathbf{q}), \tag{28}$$

where

$$\mathbf{w}(\mathbf{q}) = \begin{bmatrix} \left(\mathbf{p}_{s0}^{LT0}(\mathbf{q})\right)^T & \psi(\mathbf{q}) & \theta(\mathbf{q}) & \varphi(\mathbf{q}) \end{bmatrix}^T, \tag{29}$$

$$\hat{\mathbf{w}}(\mathbf{q}) = \begin{bmatrix} \left(\hat{\mathbf{p}}_{s0}^{LT0}(\mathbf{q})\right)^T & \hat{\psi}(\mathbf{q}) & \hat{\theta}(\mathbf{q}) & \hat{\varphi}(\mathbf{q}) \end{bmatrix}^T. \tag{30}$$

Considering N robot poses and treating the robot joint positioning error (single joint repeatability), position measurement error, and orientation measurement (i.e., encoders and level sensor) errors as having normal distributions, the optimal model parameters are [19],

$$\left[\mathbf{v}_1^*, \ldots, \mathbf{v}_N^*, \mathbf{b}_N^*\right] = \underset{\mathbf{v}_1,\ldots,\mathbf{v}_N,\mathbf{b}}{\arg\min} \left(\sum_{k=1}^{N} \mathbf{v}_k^T \boldsymbol{\Sigma}_{\mathbf{v}}^{-1} \mathbf{v}_k + \mathbf{e}_k^T \boldsymbol{\Sigma}_{\mathbf{e}}^{-1} \mathbf{e}_k \right) + \mathbf{b}_N^T \boldsymbol{\Sigma}_{\mathbf{b}}^{-1} \mathbf{b}_N, \tag{31}$$

where $\mathbf{v}_k$ is the joint positioning error for the k^{th} pose, $\mathbf{b}_N$ is the error parameter vector, $\boldsymbol{\Sigma}_{\mathbf{v}}$ is the covariance matrix for robot joint positioning accuracy, $\boldsymbol{\Sigma}_{\mathbf{e}}$ is the covariance matrix for position and orientation measurement accuracies and $\boldsymbol{\Sigma}_{\mathbf{b}}$ is the covariance matrix for the error parameters. Let $\boldsymbol{\Sigma}_{\mathbf{e}}$ be

$$\boldsymbol{\Sigma}_{\mathbf{e}} = \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{e},p} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{\Sigma}_{\mathbf{e},\psi,\theta,\varphi} \end{bmatrix}, \tag{32}$$

where $\boldsymbol{\Sigma}_{\mathbf{e},p}$ is the covariance matrix for position measurement accuracy and $\boldsymbol{\Sigma}_{\mathbf{e},\psi,\theta,\varphi}$ is the covariance matrix for orientation measurement accuracy. To determine these covariance

matrices, the method of measuring $\boldsymbol{\Sigma}_{\mathbf{v}}$ introduced in [15] is applied in this work. The determination of $\boldsymbol{\Sigma}_{\mathbf{b}}$ is a challenge. Larger values of $\boldsymbol{\Sigma}_{\mathbf{b}}$ tend to give a tighter model, but also cause more challenges and issues in the convergence of (31) due to numerical sensitivity [20]. Here, $\boldsymbol{\Sigma}_{\mathbf{b}}$ is treated as a tuning variable that can control the convergence of (31) and the modeling accuracy.

In measurement step, an appropriate number of measurements which are sufficient to build an accurate model as well as to save measurement time needs to be determined. Traditionally, to determine the measurement number, models with different number of measurements have to be constructed after the measurement step. Models performance will then be compared. However, each of these models is constructed independently which costs much time. To improve the measurement and modeling efficiency, (31) is modified and an adaptive model identification technique is introduced to update the previous model information during the measurement process. In (31), $\mathbf{v}_1^*,\ldots,\mathbf{v}_N^*,\mathbf{b}_N^*$ are the identified joint positioning errors and error model parameters with N measurements. When M new measurements are collected, the new model is constructed with

$$\begin{aligned}
&\left[\mathbf{v}_{N+1}^*,\ldots,\mathbf{v}_{N+M}^*,\mathbf{b}_{NM}^*\right] \\
&= \underset{\mathbf{v}_{N+1},\ldots,\mathbf{v}_{N+M},\mathbf{b}_{NM}}{\arg\min}\left(\sum_{k=1}^{N}\left(\mathbf{v}_k^T\boldsymbol{\Sigma}_{\mathbf{v}}^{-1}\mathbf{v}_k+\mathbf{e}_k^T\boldsymbol{\Sigma}_{\mathbf{e}}^{-1}\mathbf{e}_k\right)+\sum_{j=N+1}^{N+M}\left(\mathbf{v}_j^T\boldsymbol{\Sigma}_{\mathbf{v}}^{-1}\mathbf{v}_j+\mathbf{e}_j^T\boldsymbol{\Sigma}_{\mathbf{e}}^{-1}\mathbf{e}_j\right)+\mathbf{b}_{NM}^T\boldsymbol{\Sigma}_{\mathbf{b}}^{-1}\mathbf{b}_{NM}\right), \quad (33)\\
&= \underset{\mathbf{v}_{N+1},\ldots,\mathbf{v}_{N+M},\mathbf{b}_{NM}}{\arg\min}\left(\sum_{k=1}^{N}\left(\mathbf{e}_k^T\boldsymbol{\Sigma}_{\mathbf{e}}^{-1}\mathbf{e}_k\right)+\sum_{j=N+1}^{N+M}\left(\mathbf{v}_j^T\boldsymbol{\Sigma}_{\mathbf{v}}^{-1}\mathbf{v}_j+\mathbf{e}_j^T\boldsymbol{\Sigma}_{\mathbf{e}}^{-1}\mathbf{e}_j\right)+\mathbf{b}_{NM}^T\boldsymbol{\Sigma}_{\mathbf{b}}^{-1}\mathbf{b}_{NM}\right)
\end{aligned}$$

where $\mathbf{v}_{N+1}$, $\mathbf{v}_{N+2}$, …, $\mathbf{v}_{N+M}$ are the joint positioning errors for the new M measurements and $\mathbf{b}_{NM}$ is the error parameter vector to be identified with $N+M$ measurements. Comparing to (31), two changes are made. First, since $\mathbf{v}_1^*,\ldots,\mathbf{v}_N^*$ have been identified in the model with N measurements, they don't have to be identified again in the new model construction.

Second, while $\mathbf{b}_N$ is initially set as a vector with zeros in (31), $\mathbf{b}_{NM}$ can be set as $\mathbf{b}_N^*$ in (33), which means that the error parameters will be kept being updated based on the previous modeling results. Thus, comparing with the traditional method of (31), less parameters are identified and error parameters are set much closer to the optimal solution.

4.2. POSITION MEASUREMENT ACCURACY

A laser tracker uses the azimuth and elevation angles, as well as the distance between the laser tracker and the measured point, to determine the Cartesian position of the measured point in the laser tracker base frame. From Figure 1, the conversion process is

$$\begin{bmatrix} x_{s0}^{LT0} \\ y_{s0}^{LT0} \\ z_{s0}^{LT0} \end{bmatrix} = \begin{bmatrix} d\cos(\alpha)\sin(\beta) \\ d\sin(\alpha)\sin(\beta) \\ d\cos(\beta) \end{bmatrix}, \tag{34}$$

where d is the distance between the laser tracker and the measured point. For the laser tracker, it is known that the measured accuracy of d will decrease as d increases. The position measurement accuracy is thus distance-dependent. An example of the measurement uncertainty distribution is shown in Figure 6. The uncertain errors $\Delta\alpha$, $\Delta\beta$ and Δd on α, β and d will result with a 3D ellipsoid measurement uncertainty. The recorded measurement by the laser tracker will thus be located at any position inside the green uncertainty ellipsoid. As Cartesian position is used in (29), it is necessary to express and characterize the elliptical uncertainty from α, β and d in the forms of uncertainties along the x, y and z axis of the laser tracker base frame. In other words, the accuracies of α, β and d will be converted to describe the accuracies of x_{s0}^{LT0}, y_{s0}^{LT0} and z_{s0}^{LT0}.

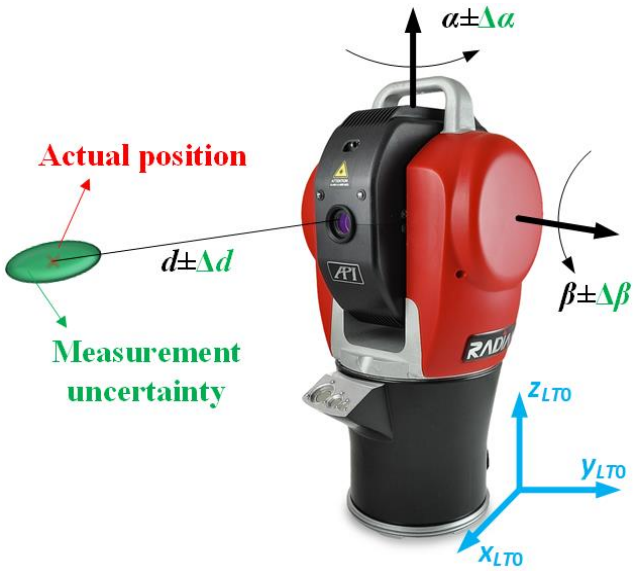

Figure 6. Laser tracker position measurement accuracy.

As x_{s0}^{LT0}, y_{s0}^{LT0} and z_{s0}^{LT0} have common variables, i.e., α, β and d as shown in (34), which all have uncertainties, the covariance matrix of the position measurement accuracy in Cartesian space is thus a non-diagonal matrix. Assuming the measurement errors of α, β and d follow normal distributions with standard deviations σ_α, σ_β and σ_d, respectively, which are given in the laser tracker specification, $\mathbf{\Sigma}_{\mathbf{e},p}$ can be written as

$$\mathbf{\Sigma}_{\mathbf{e},p} = \begin{bmatrix} \sigma_x^2 & \rho_{xy}\sigma_x\sigma_y & \rho_{xz}\sigma_x\sigma_z \\ \rho_{xy}\sigma_x\sigma_y & \sigma_y^2 & \rho_{yz}\sigma_y\sigma_z \\ \rho_{xz}\sigma_x\sigma_z & \rho_{yz}\sigma_y\sigma_z & \sigma_z^2 \end{bmatrix}, \tag{35}$$

where ρ_{ij} is the correlation coefficient between the position measurement error distributions of i and j axis, σ_x, σ_y and σ_z are standard deviations of the accuracies of position measurement in the x, y and z axis of laser tracker base frame. Since it is super complicated to directly derive the accuracies of x_{s0}^{LT0}, y_{s0}^{LT0} and z_{s0}^{LT0} with the given accuracies of α, β and d due to the nonlinear relationship in (34), a linearization is used. In this work, for a

given position, α_0, β_0 and d_0, Taylor series expansion [21] of the corresponding Cartesian positions, $x_{s0}^{LT0}(\alpha_0,\beta_0,d_0)$, $y_{s0}^{LT0}(\alpha_0,\beta_0,d_0)$ and $z_{s0}^{LT0}(\alpha_0,\beta_0,d_0)$, are estimated in terms of α_0, β_0 and d_0. Then the method of determining covariance between normal distributions [22] are used to estimate the elements in (35). The expressions for σ_x, σ_y, σ_z and ρ_{ij}, the details of which are given in the Appendix, are

$$\begin{aligned}
\sigma_x^2 &= \left(d_0 s_{\alpha_0} s_{\beta_0}\right)^2 \sigma_\alpha^2 + \left(d_0 c_{\alpha_0} c_{\beta_0}\right)^2 \sigma_\beta^2 + \left(c_{\alpha_0} s_{\beta_0}\right)^2 \sigma_d^2 \\
\sigma_y^2 &= \left(d_0 c_{\alpha_0} s_{\beta_0}\right)^2 \sigma_\alpha^2 + \left(d_0 s_{\alpha_0} c_{\beta_0}\right)^2 \sigma_\beta^2 + \left(s_{\alpha_0} s_{\beta_0}\right)^2 \sigma_d^2 \\
\sigma_z^2 &= \left(d_0 s_{\beta_0}\right)^2 \sigma_\beta^2 + \left(c_{\beta_0}\right)^2 \sigma_d^2 \\
\rho_{xy}\sigma_x\sigma_y &= \left(d_0 s_{\beta_0}\right)^2 s_{\alpha_0} c_{\alpha_0} \sigma_\alpha^2 + \left(d_0 c_{\beta_0}\right)^2 s_{\alpha_0} c_{\alpha_0} \sigma_\beta^2 + \left(s_{\beta_0}\right)^2 s_{\alpha_0} c_{\alpha_0} \sigma_d^2 \\
\rho_{xz}\sigma_x\sigma_z &= \left(d_0\right)^2 s_{\beta_0} c_{\alpha_0} c_{\beta_0} \sigma_\beta^2 + c_{\alpha_0} s_{\beta_0} c_{\beta_0} \sigma_d^2 \\
\rho_{yz}\sigma_y\sigma_z &= \left(d_0\right)^2 s_{\alpha_0} c_{\beta_0} s_{\beta_0} \sigma_\beta^2 + s_{\alpha_0} s_{\beta_0} c_{\beta_0} \sigma_d^2
\end{aligned}, \tag{36}$$

where

$$\begin{aligned}
s_{\alpha_0} &= \sin(\alpha_0),\ c_{\alpha_0} = \cos(\alpha_0) \\
s_{\beta_0} &= \sin(\beta_0),\ c_{\beta_0} = \cos(\beta_0)
\end{aligned}. \tag{37}$$

4.3. ORIENTATION MEASUREMENT ACCURACY

Treating the measured angles from the encoders and level sensor as having errors that follow normal distributions, the variances of the distributions are used to describe the orientation measurement accuracy. In this work, *K* STS configurations are measured with *M* STS measurements at each configuration. The angles of the encoders and level senor are determined using (20) and (21).

For the i^{th} robot configuration, *M* STS measurements are taken and *M* groups of encoders and level sensor angles are calculated. Let the j^{th} computed angles for the i^{th}

configuration be denoted $\psi_{i,j}$, $\theta_{i,j}$ and $\varphi_{i,j}$. The average angular measurements for the i^{th} configuration are

$$\begin{aligned} \bar{\psi}_i &= \frac{1}{M}\sum_{j=1}^{M}\psi_{i,j} \\ \bar{\theta}_i &= \frac{1}{M}\sum_{j=1}^{M}\theta_{i,j} \\ \bar{\varphi}_i &= \frac{1}{M}\sum_{j=1}^{M}\varphi_{i,j} \end{aligned} . \tag{38}$$

The errors between the measurements and averages are

$$\begin{aligned} \Delta\psi_{i,j} &= \psi_{i,j} - \bar{\psi}_i \\ \Delta\theta_{i,j} &= \theta_{i,j} - \bar{\theta}_i \\ \Delta\varphi_{i,j} &= \varphi_{i,j} - \bar{\varphi}_i \end{aligned} . \tag{39}$$

Normal distributions are then fitted using all of the errors for the i^{th} configuration. The variances, $\sigma^2_{\psi,i}$, $\sigma^2_{\theta,i}$ and $\sigma^2_{\varphi,i}$ of the fitted normal distributions are the variances at the i^{th} STS configuration. Since the two encoder angles and the level sensor angle are independent to each other, the orientation measurement covariance matrix for the i^{th} configuration is diagonal, which is

$$\mathbf{\Sigma}_{\mathbf{e},\psi,\theta,\varphi,i} = \begin{bmatrix} \sigma^2_{\psi,i} & 0 & 0 \\ 0 & \sigma^2_{\theta,i} & 0 \\ 0 & 0 & \sigma^2_{\varphi,i} \end{bmatrix}. \tag{40}$$

4.4. ROBOT REPEATABILITY

The ISO 9283 standard provides a technical procedure to determine a robot's position repeatability [23]. However, a standard for orientation repeatability does not exist. Since STS is able to measure robot's orientation information, similar to the ISO procedure

for position repeatability, the procedure for orientation repeatability is proposed and measured in this work.

Five robot poses are measured and twenty STS measurements are collected at each pose. Let the measured $\mathbf{z}_{s0}$ axis of the STS base frame of the j^{th} measurement at the i^{th} pose be denoted $\mathbf{z}_{s0,i,j}$, the mean of the vectors is

$$\mathbf{z}_{s0,i} = \frac{1}{20}\sum_{j=1}^{20}\mathbf{z}_{s0,i,j} \ . \tag{41}$$

The angle between the j^{th} measurement of the i^{th} pose and $\mathbf{z}_{s0,i,j}$ is

$$\tau_{i,j} = \frac{\operatorname{acos}\left(\mathbf{z}_{s0,i} \bullet \mathbf{z}_{s0,i,j}\right)}{\left\|\mathbf{z}_{s0,i}\right\|_2 \left\|\mathbf{z}_{s0,i,j}\right\|_2}, \tag{42}$$

and the mean angle at the i^{th} pose is

$$\overline{\tau}_i = \frac{1}{20}\sum_{j=1}^{20}\tau_{i,j} \ . \tag{43}$$

The standard deviation of the differences between the measured and mean angles is

$$\sigma_{\tau,i} = \sqrt{\frac{\sum_{j=1}^{20}\left(\tau_{i,j} - \overline{\tau}_i\right)^2}{20-1}} \ . \tag{44}$$

Thus, the local orientation repeatability of the i^{th} pose is defined as

$$R_{o,i} = \overline{\tau}_i + 3\sigma_{\tau,i} \ . \tag{45}$$

5. EXPERIMENTAL RESULTS

5.1. EXPERIMENTAL SETUP

The experimental studies are conducted on a FANUC LR Mate 200*i* robot. Figure 7 shows a photograph of the robot. Figure 8 shows a schematic of its kinematic structure and Table 1 lists the joint DH parameters.

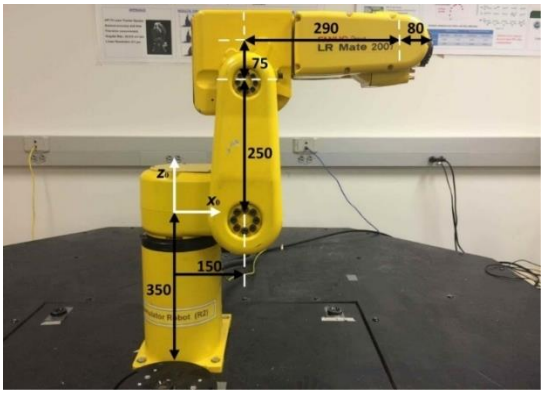

Figure 7. Photograph of FANUC LR Mate 200*i* robot side view with links lengths (mm).

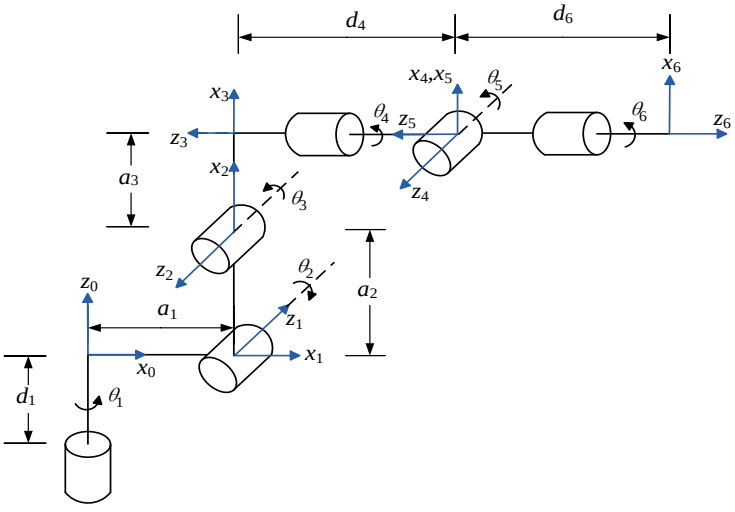

Figure 8. Schematic of FANUC LR Mate 200*i* kinematic structure with joint frames and rotation directions.

Table 1. FANUC LR Mate 200*i* DH parameters.

Frame	θ_i **(deg)**	d_i **(mm)**	a_i **(mm)**	α_i **(deg)**
1	q_1	0	150	-90
2	$q_2 - 90$	0	250	180
3	$q_2 + q_3$	0	75	-90
4	q_4	-290	0	90
5	q_5	0	0	-90
6	q_6	-80	0	180

5.2. REPEATABILITY

Both position and orientation repeatability are measured following the ISO procedure in [23] and the procedure from (41) to (45) for FANUC LR Mate 200*i*. The maximum position and orientation repeatability are determined to be ±0.03 mm and $\pm 1.22\times10^{-4}$ rad, respectively.

5.3. ESTIMATE OF MEASUREMENT COVARIANCE MATRIX

The encoders and level sensor accuracies are estimated following the procedure outlined in Section 4.3. Five hundred measurements are collected at each of five STS configurations. Figure 9 shows the five STS configurations used in this study.

Normal distributions are fitted to the angular error data for each configuration, and the standard deviations are computed. Figure 10 shows the standard deviations of the three angular errors as functions of the level sensor angle. As shown in Figure 10, the two

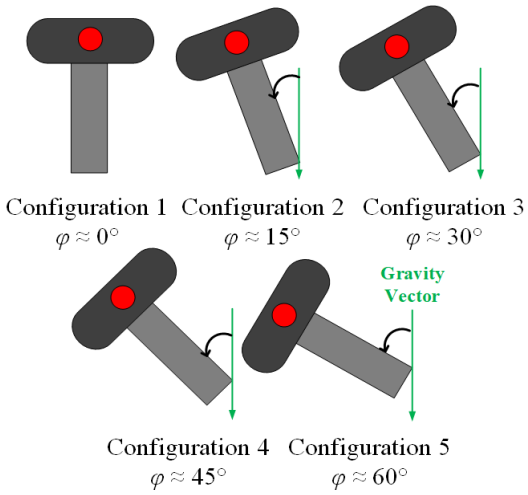

Figure 9. Five STS configurations.

encoders' accuracies are approximately equal and constant regardless of the inclination angle of the level sensor. On the contrary, the level sensor accuracy is dependent on how the level sensor is tilted with respect to the gravity vector. This is because the sensitivity and accuracy of an accelerometer will decrease as φ increases [18]. The encoders accuracies are modeled by constants and a piecewise function with two lines is fitted to the level sensor accuracy. The fitted functions are

$$\begin{aligned} \hat{\sigma}_{\psi} &= 0.0005^{\circ} \\ \hat{\sigma}_{\theta} &= 0.0005^{\circ} \\ \hat{\sigma}_{\varphi} &= \begin{cases} 0.0025^{\circ}, & if\ |\varphi| \le 30^{\circ} \\ 0.00017^{\circ}\,|\varphi| - 0.0026^{\circ}, & if\ |\varphi| > 30^{\circ} \end{cases} \end{aligned} \quad . \tag{46}$$

5.4. MEASUREMENT COLLECTION

An Automated Precision, Inc. model R-20 Radian laser tracker and an STS are used for data collection. The accuracies for azimuth, elevation and distance from the manufacturer are,

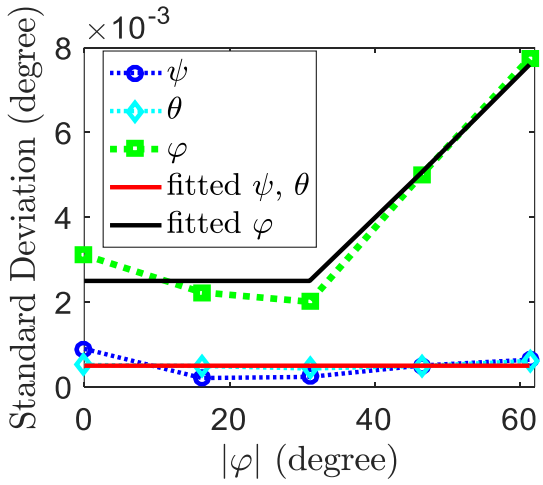

Figure 10. Standard deviations of angular errors at each configuration.

$$\begin{aligned} \sigma_\alpha &= 3.5\times10^{-6}\ rad \\ \sigma_\beta &= 3.5\times10^{-6}\ rad \\ \sigma_d &= \begin{cases} 5\ \mu m & if\ d \le 2m \\ 2.5d\ \mu m & if\ d > 2m \end{cases} \end{aligned} \quad . \tag{47}$$

Table 2 lists the minimum and maximum limits of the measured joint space. Over this space, robot commands are generated with a quasi-random distribution [24], which has the advantage of having low discrepancy such that smaller gaps and less clustering of measurements are generated. The configurations of the generated robot commands are where the STS is visible to the laser tracker and the level sensor is within its working range.

Table 2. Measured range of each joint.

	Joint 1	**Joint 2**	**Joint 3**	**Joint 4**	**Joint 5**	**Joint 6**
Minimum	-100º	-30º	-30º	-180º	-90º	-180º
Maximum	100º	100º	100º	180º	90º	180º

5.5. ONLINE ADAPTIVE MODEL IDENTIFICATION

Over the space in Table 2, STS measurements are collected. The first 300 measurements are collected as the validation set. The 50 measurements after that are used to build the first model using (31). Third order Chebyshev polynomials are selected to describe the robot kinematic error terms. Then each time a new group of 25 measurements are collected. An adaptive model is constructed with (33) based on the previous model. This process is repeated until the validation set model performance is converged.

For the purpose of model analysis, the Euclidean distances between the nominal and actual positions are defined as nominal positional residuals and the Euclidean distances between the modeled and actual positions are defined as modeled positional residuals. The angles between the nominal and measured $\mathbf{z}_{s0}$ axes are defined as the nominal angular residuals and the angles between the modeled and measured $\mathbf{z}_{s0}$ axes are defined as the modeled angular residuals. To find the potential maximum residual that the 300 validation points may not contain, a Gamma distribution is fitted to the modeled residuals. The residual at 99% of the Gamma distribution is are defined as the Gamma 99% residual [15].

Non-adaptive models with each number of identification points are also constructed using (31) for comparison. The comparison is to check if the adaptive model is able to have the same model accuracy as the non-adaptive model with the same number of identification measurements. Figure 11 shows the mean and Gamma 99% positional residuals on the validation set for the adaptive models and non-adaptive models at different number of identification measurements. Figure 12 shows the mean and Gamma 99% angular residuals on the validation set for the adaptive models and non-adaptive models at different number of identification measurements.

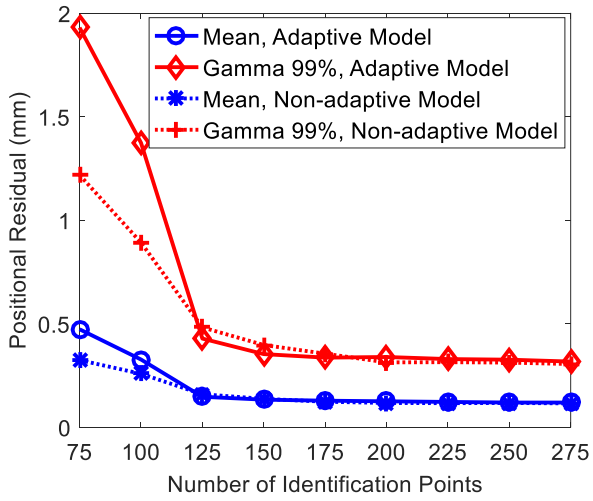

Figure 11. Mean and Gamma 99% positional residuals on validation set for adaptive and non-adaptive models at different number of identification points.

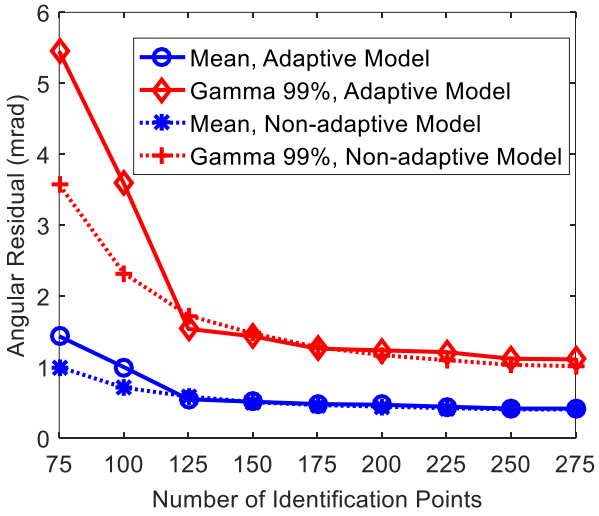

Figure 12. Mean and Gamma 99% angular residuals on validation set for adaptive and non-adaptive models at different number of identification points.

As shown in Figures 11 and 12, starting from models with 75 identification points, both positional and angular residuals decrease as more identification points are added. After 150 measurements, the decrease of validation set residuals slows down as the number

of identification points is getting sufficient. The reduction on validation set residuals is getting smaller. Considering the balance between saving measurement time and improving model accuracy, 250 is determined to be the smallest number of identification points. It takes STS about 1.5 hours to finish the collection of both identification and validation data. For the modeling step, the construction of adaptive models takes 70 minutes while the construction of non-adaptive models takes 5 hours. Using the identified model, the joint-dependent error terms are calculated and shown in Figure 13.

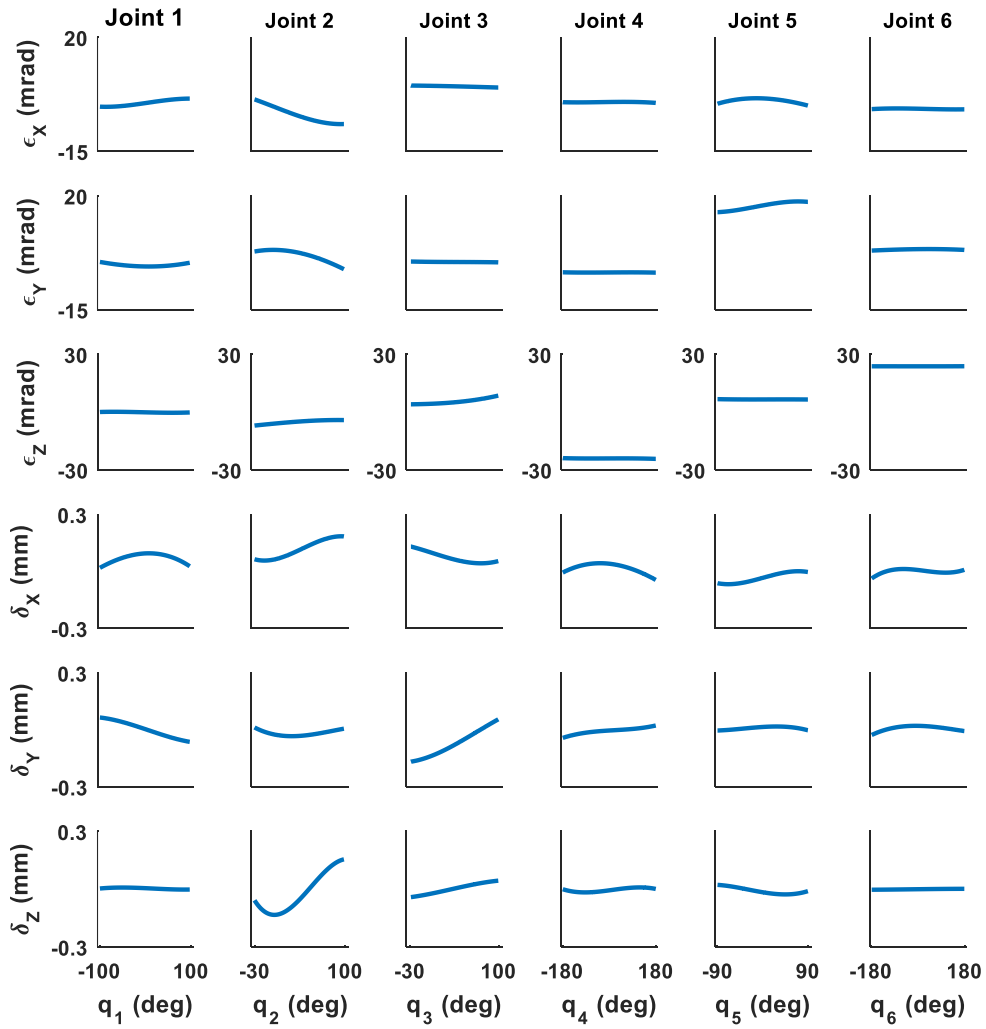

Figure 13. Rotational and translational kinematic error model terms for each joint.

In Figure 13, both rotational and translational errors have joint-dependent terms. The variations of the rotational errors especially Joint 2 indicates a complicated kinematic errors from pitch errors, strain wave gear errors and out of the rotating plane errors. The joint-dependency on translational errors is more obvious, illustrating a wobbling motion of the rotating centers for all joints. Although some errors do not show joint-dependency, the constant offsets from zero indicate that the robot also suffers rotating center offset, mastering, link length and assembly errors. Tables 3 and 4, Figures 14 to 17 show the performance of the constructed model.

Table 3. Mean, maximum and Gamma 99% positional residuals for identification and validation sets.

	Mean		**Maximum**		**Gamma 99%**	
Residual	Value (mm)	Percent Decrease	Value (mm)	Percent Decrease	Value (mm)	Percent Decrease
Identification Set Nominal	9.218	N/A	12.087	N/A	12.797	N/A
Identification Set Modeled	0.114	98.76%	0.278	97.70%	0.288	97.75%
Validation Set Nominal	9.301	N/A	12.627	N/A	12.959	N/A
Validation Set Modeled	0.118	98.76%	0.384	96.96%	0.311	97.60%

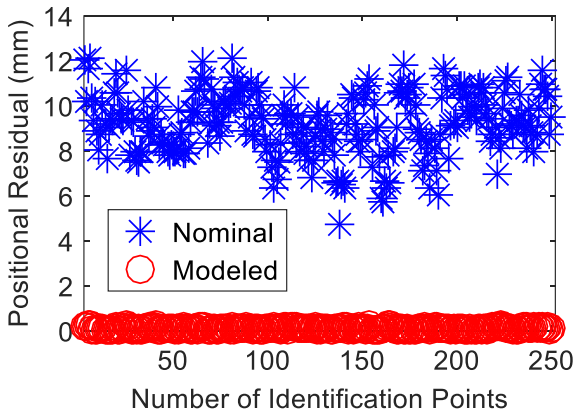

Figure 14. Nominal and modeled positional residuals for identification data set.

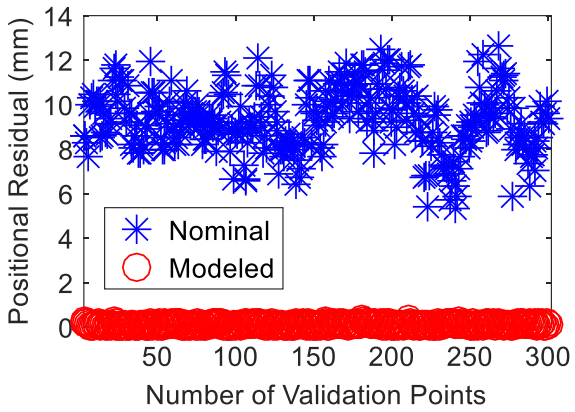

Figure 15. Nominal and modeled positional residuals for validation data set.

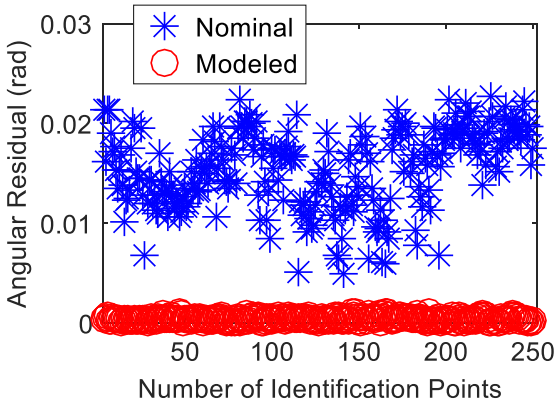

Figure 16. Nominal and modeled angular residuals for identification data set.

Table 4. Mean, maximum and Gamma 99% angular residuals for identification and validation sets.

	Mean		**Maximum**		**Gamma 99%**	
Residual	Value (rad)	Percent Decrease	Value (rad)	Percent Decrease	Value (rad)	Percent Decrease
Identification Set Nominal	0.0155	N/A	0.0227	N/A	0.0271	N/A
Identification Set Modeled	0.00045	97.10%	0.0011	95.15%	0.0012	95.57%
Validation Set Nominal	0.0157	N/A	0.0230	N/A	0.0271	N/A
Validation Set Modeled	0.0004	97.45%	0.0012	94.78%	0.0012	95.57%

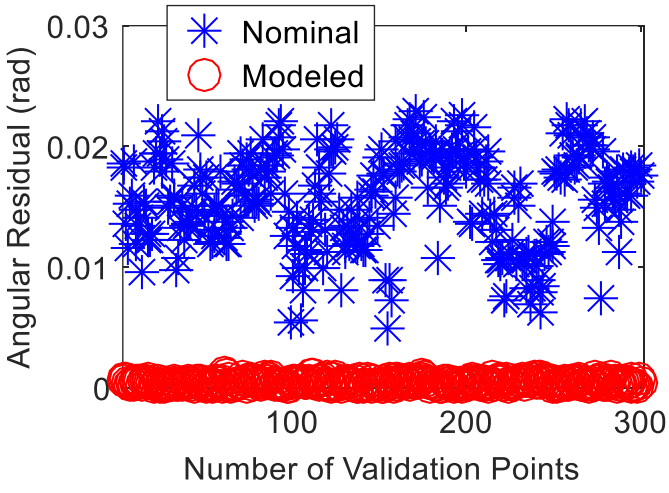

Figure 17. Nominal and modeled angular residuals for validation data set.

In Table 3, the constructed high-order joint-dependent model reduces the mean positional residual to 0.114 mm on the identification set, which is 3.8 times of the measured position repeatability. For angular residuals in Table 4, the mean residual on the identification set is reduced to 0.00045 rad, which is about 3.7 times of the measured orientation repeatability. Both positional and angular residuals are reduced to the same level comparing with the measured position and orientation repeatability. The performance of the validation set is similar as the identification set, indicating an appropriate model order and fitting of the measurements.

6. COMPARISON AND ANALYSIS

6.1. STS AND AT MODEL COMPARISON

In order to evaluate the performance of the new device comparing to traditional devices, the method in [15] with using Active Target (AT) is implemented. Figure 18 shows the three setups of AT on the robot at the same pose. At each setup, the same 250 identification points are collected. The third setup of AT is used for measuring the same 300 validation points. The whole measurement event takes about 4 hours. Third order Chebyshev polynomials are also used. Table 5 lists the mean, max, and Gamma 99% positional residuals for the AT and STS model.

The mean and Gamma 99% modeled positional residuals of the AT model are 0.018 mm and 0.027 mm larger than the mean and Gamma 99% modeled positional residuals of the STS model, respectively. The STS model has a slightly better modeled positional residuals than the AT model.

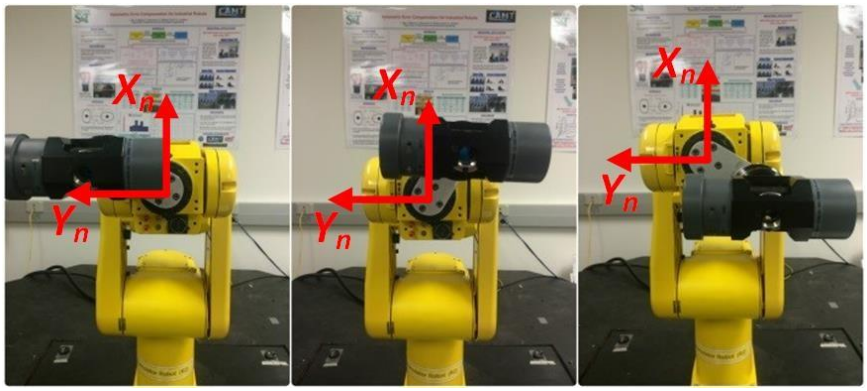

Figure 18. Three setups of AT at same robot pose.

Table 5. Mean, maximum and Gamma 99% nominal, AT and STS positional residual for validation set.

	Mean Residual (mm)	**Maximum Residual (mm)**	**Gamma 99% (mm)**
Nominal	10.191	12.923	13.856
AT	0.136	0.396	0.338
STS	0.118	0.384	0.311

The angular residuals of the AT model for the validation set are not available since only the third setup of AT is used for validation data collection. However, the identification set can be used for the analysis of angular residuals. Since there are three groups of AT measurements at each robot pose, a triangle and thus the vector that is perpendicular to this triangle plane can be determined. This vector is thus the measured z axis of the robot's last frame, which is used to represent the robot orientation. Using the calculated robot orientation, nominal robot kinematics and modeled robot kinematics, the angular residuals

are computed for the identification set. Table 6 gives the nominal, AT and STS modeled mean, maximum and Gamma 99% angular residuals for the identification set.

Table 6. Mean, maximum and Gamma 99% nominal, AT and STS angular residual for identification set.

	Mean Residual (rad)	**Maximum Residual (rad)**	**Gamma 99% (rad)**
Nominal	0.0155	0.0227	13.856
AT	0.0007	0.0015	0.0016
STS	0.00045	0.0011	0.0012

As shown in Table 6, the STS model has a better performance than the AT model regarding the angular residuals. The mean and Gamma 99% modeled angular residuals of the AT model are 0.00025 rad and 0.0004 rad larger than the mean and Gamma 99% modeled angular residuals of the STS model, which are about 2 and 3.3 times of the measured robot orientation repeatability, respectively. More improvement is seen on the angular residuals than the positional residuals.

6.2. ANALYSIS

The previous comparison shows that the STS model has a similar performance in reducing the positional residuals while has a much better performance in reducing the angular residuals than the AT model. The modeled positional residuals are similar between the STS and the AT model is because STS is actually designed based on AT. They have

the same position measurement accuracy and they use the same kinematic error model. Although AT has three setups at each robot pose, the description for the position measurement accuracy is the same for each setup.

For the orientation measurement accuracy, the case will be different. The reason of using three AT setups at each robot pose is to determine the robot orientation. As mentioned, the three AT measurements at each robot pose form a triangle, which also determines a circle. If there is no joint positioning errors and measurement errors, the triangle and the radius of the circle will be fixed. However, due to the existence of joint positioning error and measurement error, the three side lengths of the triangle vary and thus the radius of the circle also varies at different robot poses. Using the three groups of AT measurements with 250 identification points at each AT setup, triangles and circles can be fitted. Then, the radii of those circles can be determined. The variation of the radii is thus the uncertainty from the joint positioning error and measurement error. Assuming all circles are placed at a same plane sharing the same center point and the three groups of measurements have the same position measurement uncertainty, Figure 19 illustrates the relationship between the two circles with the maximum and minimum radius and the uncertainty at each AT setup.

Using the AT identification set, the maximum radius is determined as 66.527 mm and the minimum radius is 66.425 mm. Based on Figure 19, the radius of the position measurement uncertainty is thus 0.051 mm. In volumetric space, this uncertainty can be briefly treated as a ball. Thus, there will be three balls representing uncertainties at each robot pose. Since the actual position measurement may be located at any place inside the ball, at each robot pose, the actual triangle formed by three actual measurements will vary within the limitations of the three balls. In other words, the uncertainty of the position

measurement will cause an uncertainty of the triangle formed by the three position measurements, resulting with a wobbling of triangles within the three balls. Figure 20 gives an example of three possible triangles caused by the position measurement uncertainties.

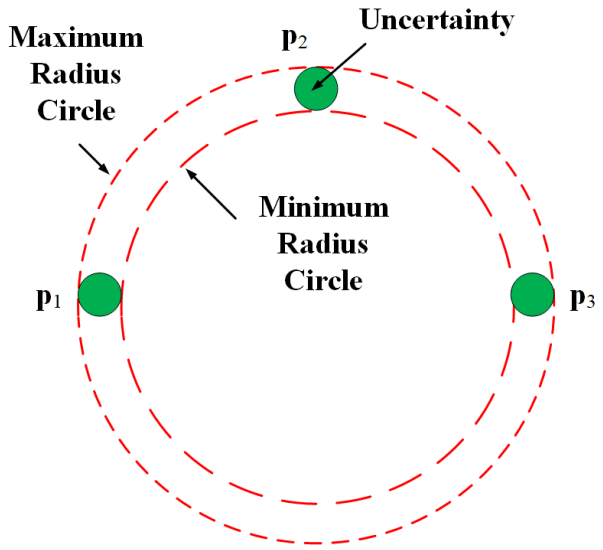

Figure 19. Relationship between maximum, minimum circle and position measurement uncertainty at one robot pose.

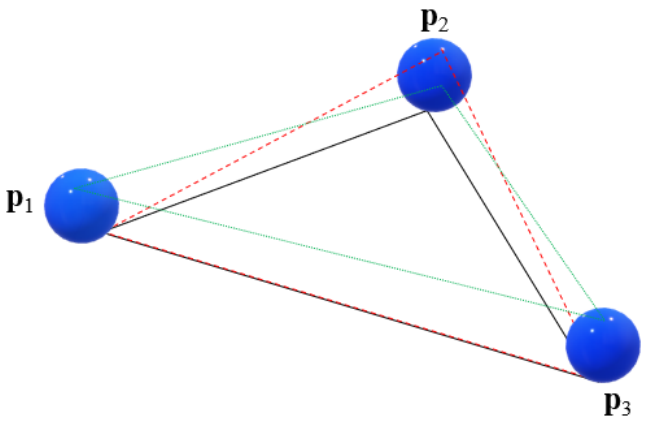

Figure 20. Three possible triangles formed by the three position measurements.

In Figure 20, the balls represent the position measurement uncertainties. For each triangle, a vector that is perpendicular to the triangle can be determined. The maximum variation of all vectors is determined which is thus the orientation measurement uncertainty. Within the wobbling of the triangles, when two measurements are fixed, the maximum change of the third measurement is the diameter of the uncertainty ball. In this case, the angle between the two vectors that are perpendicular to the two triangles, respectively, is

$$\gamma = \frac{2r_e}{R_t}, \tag{48}$$

where r_e is the radius of the position measurement uncertainty (i.e., 0.051 mm), R_t is the radius of the circle fitted with three measurements. Using the distances between each two of the three measurements, the mean radius of all circles is 66.43 mm and the maximum angle is computed as γ = 0.0015 rad. Thus, the maximum angular variation of the vector that is perpendicular to the triangle plane, which represents the orientation measurement uncertainty of the AT data, is ±0.0015 rad.

For the orientation measurement uncertainty of the STS data, from Figure 10, the maximum standard deviation of φ is 0.0076° while the standard deviation of θ and φ are 0.0005°, which are small enough to be neglected comparing with the influence to orientation accuracy of φ. The maximum uncertainty of the level sensor, i.e., φ, is thus the maximum uncertainty of $\mathbf{z}_{s2}$ and thus the maximum uncertainty of $\mathbf{z}_{s0}$. So, the orientation measurement uncertainty, which is the 3 sigma (99.7%) value of the maximum level sensor accuracy, is ±0.0004 rad (±0.0228°). So, regarding the orientation measurement uncertainty, the maximum uncertainty of the AT data (±0.0015 rad) is 3.75 times of the

maximum uncertainty of the STS data (±0.0004 rad). This explains why the STS model has smaller modeled angular residuals than the AT model.

7. SUMMARY AND CONCLUSIONS

SmartTRACK Sensor (STS), which can measure robot position and orientation information simultaneously, is introduced in this paper. By equipping a traditional position retroreflector with two rotary encoders and one level sensor, and using the angles measured by the encoders and level sensor, STS is able to determine a coordinate frame that has a fixed transformation with respect to the robot's last frame. Using a high-order joint-dependent kinematic model, a measurement model that includes robot error model, laser tracker and STS nominal kinematics is proposed to describe the STS position measurement, two encoders and the level sensor angles. The stochastic errors in the measurement system, including robot repeatability error, laser tracker position measurement error and STS orientation measurement error are also considered and characterized. By using and modifying a maximum likelihood estimator, an online adaptive model identification method is proposed to determine the least number of identification points which uses less measurement and modeling time while guarantees model accuracy.

A series of experiments were conducted on a FANUC LR mate 200*i* robot. First, the STS was attached to the robot and an API Radian laser tracker was used for data collection. The first 300 STS measurements were used as the validation set. A model was constructed with the next 50 STS measurements. Adaptive models were then constructed each time a new group of 25 measurements was collected. The online adaptive method was

shown to be able to construct models with the same performance as the non-adaptive models. The performance of a model with 250 identification measurements was shown. The model was able to reduce the mean positional errors by 98.76% and 97.10% for the identification and validation sets, respectively. The mean angular errors were also reduced by 98.73% and 97.45%, respectively.

Another experiment comparing the model performance between an STS model and an AT model was implemented. Three AT setups at each robot pose were used to acquire robot orientation information. Although the AT model was able to reduce the positional errors to the same level of magnitude as the STS model, its mean modeled angular residual was larger than the mean modeled angular residual of the STS model by a magnitude of 2 times of the robot orientation repeatability. An analysis was given on the position and orientation measurement uncertainties of STS and AT. While the AT measurements have the same position measurement uncertainty as the STS measurements, the maximum orientation measurement uncertainty of the AT data is about 3.8 times of the maximum orientation measurement uncertainty of the STS data. STS is able to reduce the measurement time as well as to further improve model accuracy.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support for this work from the National Science Foundation (Award No. CMMI 1335340) and the Center for Aerospace Manufacturing Technologies at the Missouri University of Science and Technology.

APPENDIX

For a given point, α_0, β_0, d_0, using Taylor series, x_{s0}^{LT0}, y_{s0}^{LT0} and z_{s0}^{LT0} can be approximated as

$$\begin{aligned} x_{s0}^{LT0} &\cong d_0\cos(\alpha_0)\sin(\beta_0)+ \\ &\left(x_{s0}^{LT0}\right)'(\alpha_0)(\alpha-\alpha_0)+\left(x_{s0}^{LT0}\right)'(\beta_0)(\beta-\beta_0)+\left(x_{s0}^{LT0}\right)'(d_0)(d-d_0) \\ &= d_0\cos(\alpha_0)\sin(\beta_0)+ \\ &\left.\frac{\partial x_{s0}^{LT0}}{\partial\alpha}\right|_{\alpha_0,\beta_0,d_0}(\alpha-\alpha_0)+\left.\frac{\partial x_{s0}^{LT0}}{\partial\beta}\right|_{\alpha_0,\beta_0,d_0}(\beta-\beta_0)+\left.\frac{\partial x_{s0}^{LT0}}{\partial d}\right|_{\alpha_0,\beta_0,d_0}(d-d_0) \\ &= d_0\cos(\alpha_0)\sin(\beta_0)-d_0\sin(\alpha_0)\sin(\beta_0)(\alpha-\alpha_0) \\ &\quad +d_0\cos(\alpha_0)\cos(\beta_0)(\beta-\beta_0)+\cos(\alpha_0)\sin(\beta_0)(d-d_0) \end{aligned}, \tag{49}$$

$$\begin{aligned} y_{s0}^{LT0} &\cong d_0\sin(\alpha_0)\sin(\beta_0)+ \\ &\left(y_{s0}^{LT0}\right)'(\alpha_0)(\alpha-\alpha_0)+\left(y_{s0}^{LT0}\right)'(\beta_0)(\beta-\beta_0)+\left(y_{s0}^{LT0}\right)'(d_0)(d-d_0) \\ &= d_0\sin(\alpha_0)\sin(\beta_0)+d_0\cos(\alpha_0)\sin(\beta_0)(\alpha-\alpha_0) \\ &\quad +d_0\sin(\alpha_0)\cos(\beta_0)(\beta-\beta_0)+\sin(\alpha_0)\sin(\beta_0)(d-d_0) \end{aligned}, \tag{50}$$

$$\begin{aligned} z_{s0}^{LT0} &\cong d_0\cos(\beta_0)+\left(z_{s0}^{LT0}\right)'(\beta_0)(\beta-\beta_0)+\left(z_{s0}^{LT0}\right)'(d_0)(d-d_0) \\ &= d_0\cos(\beta_0)-d_0\sin(\beta_0)(\beta-\beta_0)+\cos(\beta_0)(d-d_0) \end{aligned}. \tag{51}$$

Similarly, for a measured point, $\bar{\alpha}_0$, $\bar{\beta}_0$ and $\bar{d}_0$ which have position measurement errors, the measured Cartesian space positions $\bar{x}_{s0}^{LT0}$, $\bar{y}_{s0}^{LT0}$ and $\bar{z}_{s0}^{LT0}$ have the same representations as (49), (50) and (51) in terms of $\bar{\alpha}_0$, $\bar{\beta}_0$ and $\bar{d}_0$. The difference between measured and nominal Cartesian space positions are thus,

$$
\begin{aligned}
&\bar{x}_{s0}^{LT0} - x_{s0}^{LT0} \\
&= d_0 \cos(\alpha_0)\sin(\beta_0) - d_0 \sin(\alpha_0)\sin(\beta_0)(\bar{\alpha}_0 - \alpha_0) \\
&\quad + d_0 \cos(\alpha_0)\cos(\beta_0)(\bar{\beta}_0 - \beta_0) \\
&\quad + \cos(\alpha_0)\sin(\beta_0)(\bar{d}_0 - d_0) - d_0 \cos(\alpha_0)\sin(\beta_0) \ , \\
&= -d_0 \sin(\alpha_0)\sin(\beta_0)(\bar{\alpha}_0 - \alpha_0) \\
&\quad + d_0 \cos(\alpha_0)\cos(\beta_0)(\bar{\beta}_0 - \beta_0) \\
&\quad + \cos(\alpha_0)\sin(\beta_0)(\bar{d}_0 - d_0)
\end{aligned}
\tag{52}
$$

$$
\begin{aligned}
&\bar{y}_{s0}^{LT0} - y_{s0}^{LT0} \\
&= d_0 \sin(\alpha_0)\sin(\beta_0) + d_0 \cos(\alpha_0)\sin(\beta_0)(\bar{\alpha}_0 - \alpha_0) \\
&\quad + d_0 \sin(\alpha_0)\cos(\beta_0)(\bar{\beta}_0 - \beta_0) \\
&\quad + \sin(\alpha_0)\sin(\beta_0)(\bar{d}_0 - d_0) - d_0 \sin(\alpha_0)\sin(\beta_0) \ , \\
&= d_0 \cos(\alpha_0)\sin(\beta_0)(\bar{\alpha}_0 - \alpha_0) \\
&\quad + d_0 \sin(\alpha_0)\cos(\beta_0)(\bar{\beta}_0 - \beta_0) \\
&\quad + \sin(\alpha_0)\sin(\beta_0)(\bar{d}_0 - d_0)
\end{aligned}
\tag{53}
$$

$$
\begin{aligned}
&\bar{z}_{s0}^{LT0} - z_{s0}^{LT0} \\
&= d_0 \cos(\beta_0) - d_0 \sin(\beta_0)(\bar{\beta}_0 - \beta_0) + \cos(\beta_0)(\bar{d}_0 - d_0) - d_0 \cos(\beta_0). \\
&= -d_0 \sin(\beta_0)(\bar{\beta}_0 - \beta_0) + \cos(\beta_0)(\bar{d}_0 - d_0)
\end{aligned}
\tag{54}
$$

It is known that

$$
\begin{aligned}
&\bar{\alpha}_0 - \alpha_0 \sim N\left(0, \sigma_\alpha^2\right) \\
&\bar{\beta}_0 - \beta_0 \sim N\left(0, \sigma_\beta^2\right). \\
&\bar{d}_0 - d_0 \sim N\left(0, \sigma_d^2\right)
\end{aligned}
\tag{55}
$$

Let

$$
\begin{aligned}
&\bar{x}_{s0}^{LT0} - x_{s0}^{LT0} \sim N\left(0, \sigma_x^2\right) \\
&\bar{y}_{s0}^{LT0} - y_{s0}^{LT0} \sim N\left(0, \sigma_y^2\right). \\
&\bar{z}_{s0}^{LT0} - z_{s0}^{LT0} \sim N\left(0, \sigma_z^2\right)
\end{aligned}
\tag{56}
$$

Thus,

$$\begin{aligned}\sigma_x^2 &= \left(d_0 s_{\alpha_0} s_{\beta_0}\right)^2 \sigma_\alpha^2 + \left(d_0 c_{\alpha_0} c_{\beta_0}\right)^2 \sigma_\beta^2 + \left(c_{\alpha_0} s_{\beta_0}\right)^2 \sigma_d^2 \\ \sigma_y^2 &= \left(d_0 c_{\alpha_0} s_{\beta_0}\right)^2 \sigma_\alpha^2 + \left(d_0 s_{\alpha_0} c_{\beta_0}\right)^2 \sigma_\beta^2 + \left(s_{\alpha_0} s_{\beta_0}\right)^2 \sigma_d^2 , \\ \sigma_z^2 &= \left(d_0 s_{\beta_0}\right)^2 \sigma_\beta^2 + \left(c_{\beta_0}\right)^2 \sigma_d^2\end{aligned} \tag{57}$$

where

$$\begin{aligned}s_{\alpha_0} &= \sin\left(\alpha_0\right), c_{\alpha_0} = \cos\left(\alpha_0\right) \\ s_{\beta_0} &= \sin\left(\beta_0\right), c_{\beta_0} = \cos\left(\beta_0\right)\end{aligned} . \tag{58}$$

Also, using the method of determining the covariance between normal distributions, the non-diagonal elements of (35) are computed with the coefficients in (52), (53) and (54),

$$\begin{aligned}\rho_{xy}\sigma_x\sigma_y &= d_0 s_{\alpha_0} s_{\beta_0} d_0 c_{\alpha_0} s_{\beta_0} \sigma_\alpha^2 + d_0 c_{\alpha_0} c_{\beta_0} d_0 s_{\alpha_0} c_{\beta_0} \sigma_\beta^2 + c_{\alpha_0} s_{\beta_0} s_{\alpha_0} s_{\beta_0} \sigma_d^2 \\ &= \left(d_0 s_{\beta_0}\right)^2 s_{\alpha_0} c_{\alpha_0} \sigma_\alpha^2 + \left(d_0 c_{\beta_0}\right)^2 s_{\alpha_0} c_{\alpha_0} \sigma_\beta^2 + \left(s_{\beta_0}\right)^2 s_{\alpha_0} c_{\alpha_0} \sigma_d^2\end{aligned} , \tag{59}$$

$$\begin{aligned}\rho_{xz}\sigma_x\sigma_z &= d_0 c_{\alpha_0} c_{\beta_0} d_0 s_{\beta_0} \sigma_\beta^2 + c_{\alpha_0} s_{\beta_0} c_{\beta_0} \sigma_d^2 \\ &= \left(d_0\right)^2 s_{\beta_0} c_{\alpha_0} c_{\beta_0} \sigma_\beta^2 + c_{\alpha_0} s_{\beta_0} c_{\beta_0} \sigma_d^2\end{aligned} , \tag{60}$$

$$\begin{aligned}\rho_{yz}\sigma_y\sigma_z &= d_0 s_{\alpha_0} c_{\beta_0} d_0 s_{\beta_0} \sigma_\beta^2 + s_{\alpha_0} s_{\beta_0} c_{\beta_0} \sigma_d^2 \\ &= \left(d_0\right)^2 s_{\alpha_0} c_{\beta_0} s_{\beta_0} \sigma_\beta^2 + s_{\alpha_0} s_{\beta_0} c_{\beta_0} \sigma_d^2\end{aligned} . \tag{61}$$

REFERENCES

[1] A. Grau, M. Indri, L.L. Bello, T. Sauter, Industrial robotics in factory automation: from the early stage to the internet of things. 43rd Annual Conference of the IEEE Industrial Electronics Society, 2017, 6159-6164.

[2] Z. Roth, B. Mooring, B. Ravani, An overview of robot calibration. IEEE Journal on Robotics and Automation, 1987, 5(3):377-385.

[3] D. Cong, D. Yu, J. Han, Kinematic calibration of parallel robots using CMM. 6th World Congress on Intelligent Control and Automation, 2006, 2:8514-8518.

[4] A. Nubiola, M. Slamani, A. Joubair, I.A. Bonev, Comparison of two calibration methods for a small industrial robot based on an optical CMM and a laser tracker. Robotica, 2014, 32(3):447-466.

[5] A. Nubiola, I.A. Bonev, Absolute robot calibration with a single telescoping ballbar. Precision Engineering, 2014, 38(3):472-480.

[6] C. Ye, H. Zhao, Y. Lin, H. Ding, Kinematic calibration and compensation for industrial robots using a telescoping ballbar. IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems, 2017, 574-579.

[7] J.M.S. Motta, G.C. de Carvalho, R.S. McMaster, Robot calibration using a 3D vision-based measurement system with a single camera. Robotics and Computer-Integrated Manufacturing, 2001, 17(6):487-497.

[8] H. Zhuang, Z.S. Roth, K. Wang, Robot calibration by mobile camera systems. Journal of Robotic Systems, 1994, 11(3):155-167.

[9] T. Sun, Y. Zhai, Y. Song, J. Zhang, Kinematic calibration of a 3-DoF rotational parallel manipulator using laser tracker. Robotics and Computer-Integrated Manufacturing, 2016, 41:78-91.

[10] J. Santolaria, J. Conte, M. Ginés, Laser tracker-based kinematic parameter calibration of industrial robots by improved CPA method and active retroreflector. The International Journal of Advanced Manufacturing Technology, 2013, 1-20.

[11] G. Alici, B. Shirinzadeh, A systematic technique to estimate positioning errors for robot accuracy improvement using laser interferometry based sensing. Mechanism and Machine Theory, 2005, 40(8):879-906.

[12] Y. Bai, H. Zhuang, S. Roth, Experiment study of PUMA robot calibration using a laser tracking system. Proceedings of the IEEE International Workshop on Soft Computing in Industrial Applications SMCia/03, 2003, 139-144.

[13] S. Aguado, D. Samper, J. Santolaria, J.J. Aguilar, Identification strategy of error parameter in volumetric error compensation of machine tool based on laser tracker measurements. International Journal of Machine Tools and Manufacture, 2012, 53(1):160-169.

[14] A. Nubiola, I.A. Bonev, Absolute calibration of an ABB IRB 1600 robot using a laser tracker. Robotics and Computer-Integrated Manufacturing, 2013, 29(1):236-245.

[15] L. Ma, P. Bazzoli, P.M. Sammons, R.G. Landers, D.A. Bristow, Modeling and calibration of high-order joint-dependent kinematic errors for industrial robots. Robotics and Computer-Integrated Manufacturing, 2018, 50:153-167.

[16] M.W. Spong, S. Hutchinson, M. Vidyasagar, Robot Modeling and Control, Wiley, New York, 2006.

[17] P. Freeman, A novel means of software compensation for robots and machine tools. 2006 SAE Technical Paper, 2006-01-3167.

[18] S. Luczak, W. Oleksiuk, M. Bodnicki, Sensing tilt with MEMS accelerometers. IEEE Sensors Journal, 2006, 6(6):1669-1675.

[19] C.W. Wampler, J.M. Hollerbach, T. Arai, An implicit loop method for kinematic calibration and its application to closed-chain mechanisms. Robotics and Automation, 1995, 11(5):710-724.

[20] J. Creamer, P.M. Sammons, D.A. Bristow, R.G. Landers, P.L. Freeman, S.J. Easley, Table-based volumetric error compensation of large five-axis machine tools. Journal of Manufacturing Science and Engineering, 2017, 139(2):021011.

[21] S. Lang, A First Course in Calculus, Springer Science & Business Media, 2012.

[22] L.J. Bain, M. Engelhardt, Introduction to Probability and Mathematical Statistics, Brooks/Cole, 1987.

[23] ISO 9283, Manipulating industrial robots-Performance criterion and related test methods. International Standardization Organization, 1998.

[24] H. Niederreiter, Low-discrepancy and low-dispersion sequences. Journal of Number Theory, 1988, 30(1):51-70.

SECTION

2. CONCLUSIONS

Indirect measurement compensation methods for machine tools always suffer a curve-fitting challenge. Measurement space is often limited and error model tends to lose its accuracy around and out of the measured space. To achieve a uniform and accurate model, a method that can optimally interpolate and extrapolate machine tool error model over the entire workspace is proposed in this work. Using axis perturbation model, an analytical relationship between tool tip volumetric error slopes and single geometric error slopes is constructed. By introducing a nonlinear constraint on the tool tip error slopes during model identification process, all single geometric error slopes are controlled and model parameters are optimized throughout the entire workspace. A method of determining the constraint value is developed. A comparison between constrained and unconstrained models is given, which shows that the proposed method is able to keep the interpolated model performance as well as to improve the extrapolated model accuracy.

The previous volumetric error compensation methods for industrial robots also have limitations and challenges. Simple robot kinematic errors are focused and complicated errors are often neglected. Thus, the final compensation accuracy is also limited. Paper II gives a summary of robot kinematic error sources. The model description of each one is also given. Then, a general high-order joint-dependent kinematic error model, describing both simple error (e.g., link length and alignment errors) and complicated error (e.g., strain wave gearing error) is proposed. Methods for measurement, error identification

and compensation are given. Experimental results demonstrate that about 20% more kinematic errors are captured by the proposed joint-dependent comparing to the traditional joint-independent error model. In addition to the previous limitation in robot error modeling step, the measurement step also suffers a difficulty in acquiring robot's orientation information. Measurement devices have to be set up multiple times to measure robot's orientation which costs more time and introduces more uncertainties. To address this issue, a new device that can measure robot position and orientation information simultaneously is introduced in Paper III. A measurement model, which describes all kinematics in the measurement system including a laser tracker, robot and SmartTRACK Sensor, is proposed. Robot static kinematic errors, repeatability errors and measurement errors are all considered and described in the proposed model. An online adaptive model identification method is also proposed which improves the modeling efficiency to determine the most appropriate number of identification measurements. Experimental results demonstrate an improvement in both accuracy and efficiency with the new device and the proposed adaptive modeling method.

BIBLIOGRAPHY

[1] Okafor, A.C., and Ertekin, Y. M., 2000, "Derivation of Machine Tool Error Models and Error Compensation Procedure for Three Axes Vertical Machining Center using Rigid Body Kinematics," International Journal of Machine Tools and Manufacture, 40(8), pp.1199-1213.

[2] Bringmann, B., Besuchet, J. P., and Rohr, L., 2008, "Systematic Evaluation of Calibration Methods," CIRP Annals–Manufacturing Technology, 57(1), pp. 529–532.

[3] Xiang, S., and Altintas, Y., 2016, "Modeling and Compensation of Volumetric Errors for Five-Axis Machine Tools," International Journal of Machine Tools and Manufacture, 101, pp.65-78.

[4] Schwenke, H., Knapp, W., Haitjema, H., Weckenmann, A., Schmitt, R., and Delbressine, F., 2008, "Geometric Error Measurement and Compensation of Machines-An Update," CIRP Annals-Manufacturing Technology, 57(2), pp. 660-675.

[5] Ibaraki, S., and Knapp, W., 2012, "Indirect Measurement of Volumetric Accuracy for Three Axis and Five-Axis Machine Tools: A Review," International Journal of Automation Technology, 6(2), pp.110-124.

[6] ISO, 2012, "Test Code for Machine Tools Part I: Geometric Accuracy of Machine Tools Operating Under No-Load or Quasi-Static Conditions," International Organization for Standardization, Geneva, Switzerland, Standard No. ISO 230-1.

[7] Yang, J., and Altintas, Y., 2013, "Generalized Kinematics of Five-Axis Serial Machines with Non-Singular Tool Path Generation," International Journal of Machine Tools and Manufacture, 75, pp.119-132.

[8] Fu, G., Fu, J., Xu, Y., and Chen, Z., 2014, "Product of Exponential Model for Geometric Error Integration of Multi-Axis Machine Tools," The International Journal of Advanced Manufacturing Technology, 71(9-12), pp.1653-1667.

[9] Bringmann, B., Besuchet, J. P., and Rohr, L., 2003, "Identification and Compensation of Systematic Deviations Particular to 5-Axis Machining Centers", International Journal of Machine Tools and Manufacture, 43(8), pp. 771–780.

[10] Lei, W. T., Sung, M. P., Liu, W. L., and Chuang, Y. C., 2007, "Double Ballbar Test for the Rotary Axes of Five-Axis CNC Machine Tools", International Journal of Machine Tools and Manufacture, 47(2), pp. 273–285.

[11] Ibaraki, S., Oyama, C., and Otsubo, H., 2011, “Construction of an Error Map of Rotary Axes on a Five-Axis Machining Center by Static R-Test”, International Journal of Machine Tools and Manufacture, 51(3), pp. 190–200.

[12] Bringmann, B., and Knapp, W., 2006, “Model-Based ‘Chase-the-Ball’ Calibration of a 5-Axes Machining Center”, CIRP Annals-Manufacturing Technology, 55(1), pp. 531–534.

[13] Erkan, T., and Mayer, J. R. R., 2010, “A Cluster Analysis Applied to Volumetric Errors of 5-Axis Machine Tools Obtained by Probing an Uncalibrated Artifact”, CIRP Annals-Manufacturing Technology, 59(1), pp. 539–542.

[14] Ibaraki, S., Iritani, T., and Matsushita, T., 2012, “Calibration of Location Errors of Rotary Axes on 5-Axis Machine Tools by On-the-Machine Measurement Using a Touch-Trigger Probe”, International Journal of Machine Tools and Manufacture, 58(1), pp. 44–53.

[15] Aguado, S., Samper, D., Santolaria, J., and Aguilar, J. J., 2012, “Identification Strategy of Error Parameter in Volumetric Error Compensation of Machine Tool Based on Laser Tracker Measurements”, International Journal of Machine Tools and Manufacture, 53(1), pp. 160–169.

[16] Wang, J., Guo, J., Zhang, G., Guo, B. A., and Wang, H., 2012, “The Technical Method of Geometric Error Measurement for Multi-Axis NC Machine Tool by Laser Tracker”, Measurement science and technology, 23(4), p. 045003.

[17] Hong, C., Ibaraki, S., and Matsubara, A., 2001, “Influence of Position Dependent Error of Rotary Axes on a Machining Test of Cone Frustram by 5-Axis Machine Tools”, Precision Engineering, 35(1), pp. 1–11.

[18] Creamer, J., Sammons, P. M., Bristow, D. A., Landers, R. G., Freeman, P. L., and Easley, S. J., 2017, “Table-Based Volumetric Error Compensation of Large Five-Axis Machine Tools,” ASME Journal of Manufacturing Science and Engineering, 139(2), p. 021011.

[19] Choi, J. P., Min, B. K., and Lee, S. J., 2004, “Reduction of Machining Errors of a Three-Axis Machine Tool by On-Machine Measurement and Error Compensation System”, Journal of Materials Processing Technology, 155, pp. 2056–2064.

[20] Xiang, S., and Altintas, Y., 2016, “Modeling and Compensation of Volumetric Errors for Five-Axis Machine Tools”, International Journal of Machine Tools and Manufacture, 101, pp. 65–78.

[21] Runge, C., 1901, ““Uber empirische Funktionen und die Interpolation zwischen ¨aquidistanten Ordinaten”, Zeitschrift f ¨ur Mathematik und Physik, 46, pp. 224–243.

[22] Summers, M., 2005, "Robot Capability Test and Development of Industrial Robot Positioning System for the Aerospace Industry," SAE Transactions, 114, pp. 1108-1118.

[23] Mooring, B. W., Roth, Z. S., and Driels, M. R., 1991, "Fundamentals of Manipulator Calibration," New York, Wiley.

[24] Hollerbach, J. M., and Wampler, C. W., 1996, "The Calibration Index and Taxonomy for Robot Kinematic Calibration Methods," International Journal of Robotics Research, 15(6), pp. 573-591.

[25] Elatta, A. Y., Gen, L. P., Zhi, F. L., Daoyuan, Y., and Fei, L., 2004, "An Overview of Robot Calibration," Information Technology Journal, 3(1), pp. 74-78.

[26] Spong, M. W., Hutchinson, S., and Vidyasagar, M., 2006, "Robot Modeling and Control," New York, Wiley.

[27] Santolaria, J., Conte, J., Pueo, M., and Javierre, C., 2014, "Rotation Error Modeling and Identification for Robot Kinematic Calibration by Circle Point Method," Metrology and Measurement System, 21(1), pp. 85-98.

[28] Stone, H. W., 1986, "Kinematic Modeling, Identification and Control of Robotic Manipulator," Ph.D. Thesis, Robotic Institute, Carnegie Mellon University.

[29] Okamura, K., and Park, F. C., 1996, "Kinematic Calibration Using the Product of Exponentials Formula," Robotica, 14(4), pp. 415-421.

[30] Cong, D., Yu, D., and Han, J., 2006, "Kinematic Calibration of Parallel Robots Using CMM," 6th World Congress on Intelligent Control and Automation, 2, pp. 8514-8518, IEEE.

[31] Nubiola, A., and Bonev, I. A., 2014, "Absolute Robot Calibration with a Single Telescoping Ballbar," Precision Engineering, 38(3), pp. 472-480.

[32] Motta, J. M. S., de Carvalho, G. C., and McMaster, R. S., 2001, "Robot Calibration Using a 3D Vision-Based Measurement System with a Single Camera," Robotics and Computer-Integrated Manufacturing, 17(6), pp. 487-497.

[33] Zhuang, H., Roth, Z. S., and Wang, K., 1994, "Robot Calibration by Mobile Camera Systems," Journal of Robotic Systems, 11(3), pp.155-167.

[34] Nubiola, A., and Bonev, I. A., 2013, "Absolute Calibration of an ABB IRB 1600 Robot Using a Laser Tracker," Robotics and Computer-Integrated Manufacturing, 29(1), pp.236-245.

[35] Bai, Y., Zhuang, H., and Roth, Z. S., 2003, "Experiment Study of PUMA Robot Calibration Using a Laser Tracking System," In Proceedings of the 2003 IEEE International Workshop on Soft Computing in Industrial Applications, SMCia/03, pp. 139-144. IEEE.

[36] Bennet, D. J., and Hollerbach, J. M., 1991, "Autonomous Calibration, of Single-Loop Closed Kinematic Chains, Formed by Manipulators with Passive Endpoint Constraints," IEEE Transaction on Robotics and Automation, 7, pp. 597-606.

[37] Renders, J. M., Rossignal, E., Becquet, M., and Hanus, R., 1991, "Kinematic Calibration and Geometrical Parameter Identification for Robots," IEEE Transaction on Robotics and Automation, 7, pp. 721-732.

[38] Karan, B., and Vukobratović, M., 1994, "Calibration and Accuracy of Manipulation Robot Models—An Overview," Mechanism and Machine Theory, 29(3), pp. 479-500.

[39] Santolaria, J., Conte, J., Pueo, M. and Javierre, C., 2014, "Rotation Error Modeling and Identification for Robot Kinematic Calibration by Circle Point Method," Metrology and Measurement System, 21(1), pp. 85-98.

[40] Santolaria, J., Conte, J., and Ginés, M., 2013, "Laser Tracker-Based Kinematic Parameter Calibration of Industrial Robots by Improved CPA Method and Active Retroreflector," International Journal of Advanced Manufacturing Technology, 66, pp. 2087-2106.

[41] Ma, L., Bazzoli, P., Sammons, P. M., Landers, R. G., and Bristow, D. A, 2018, "Modeling and Calibration of High-Order Joint-Dependent Kinematic Errors for Industrial Robots," Robotics and Computer-Integrated Manufacturing, 50, pp. 153-167.

VITA

Le Ma was born in Huludao, China. He earned his Bachelor of Science in Mechanical Engineering in June 2011 from Dalian University of Technology, China. He started his direct Ph.D. study in the Department of Mechanical and Aerospace Engineering at Missouri University of Science and Technology in August 2013. In July 2019, he received his Ph.D. in Mechanical Engineering from Missouri University of Science and Technology.