Error Analysis on Closed-Form Solutions for Kinematic Calibration

Sheng-Wen Shih[†], Yi-Ping Hung[†], and Wei-Song Lin[‡]

[†]Institute of Information Science, Academia Sinica, Nankang, Taipei, Taiwan.
[‡]Institute of Electrical Engineering, National Taiwan University, Taipei, Taiwan.
Email: hung@iis.sinica.edu.tw

Abstract

Many closed-form solutions have been developed for calibrating robot kinematic parameters. The existing closed-form solutions for kinematic calibration can be classfied into two categories according to the information they used. Methods estimating kinematic parameters by using pose measurements are referred to as the pose methods. While there is only one closed-form solution referred to as the point method using 3D point measurements for calibrating robot kinematic parameters. Relatively less work has been devoted to the error analysis on the calibration methods. Error analysis results are very useful to serve as a guideline for selecting calibration techniques, for determining the calibration condition and even for designing a robot head or a robot arm when considering the calibration task. In this paper, we derived the expressions of variances of the kinematic parameters estimated by using the point method or the pose method, respectively. The derived error variances for the point method are functions of the calibration range, number of measurements, amount of measurement noise and amount of joint value noise. Furthermore, if the joint under calibration is revolute, then the error variances are also functions of the distance between the calibration point and the revolute joint axis and length of the link corresponding to the joint under calibration. The derived error variances for a pose method are functions of the calibration range, number of measurements, amount of measurement noise, amount of the joint value noise and length of the link corresponding to the joint under calibration.

Keywords

Error Analysis, Kinematic Calibration, Rotation Axis Estimation

I. Introduction

Kinematic models are important for controlling robot manipulators and binocular robot heads. In general, a robot is manufactured to have high repeatability and because of manufacturing inaccuracy, aging and some non-geometric factors, robot positioning accuracy is usually worse than its repeatability. Therefore, for applying robots to more applications requiring high positioning accuracy, many techniques were developed for estimating accurate kinematic parameters of a robot, e.g., [1], [2], [3], [5], [6], [7], [9], [10], [11], [14], [13], [17], [16], [19], [20], [22], [21] and [23]. In contrast, relatively less work has been devoted to the theoretical analysis on the estimation error of kinematic parameters. Theoretical error analysis on the estimation error is important for revealing factors that dominat calibration accuracy and for reducing the calibration error. Furthermore, based on the error analysis results, the most appropriate calibration technique can be determined for obtaining accurate calibration results.

A. Closed-form Solutions and Nonlinear Optimization Techniques

In general, kinematic equations are nonlinear in parameters, therefore, most of the calibration methods are based on nonlinear optimization techniques, e.g., [1], [2], [5], [6], [7], [9], [10], [11], [19], [22], and [23]. This kind of calibration techniques are very accurate, providing that an accurate initial estimate of the kinematic parameters is available. Notice that although kinematic equations are nonlinear in parameters, the calibration problem can be made linear by estimating parameters of one joint axis at a time, which is exactly the way all the closed-form solutions doing. As a result, initial estimate of kinematic parameters for nonlinear optimization can be obtained from closed-form solutions. Closed-form solutions are attractive because they are more reliable than a nonlinear iterative solution. Furthermore, accuracy of a closed-form solution can be carefully improved to be comparable with that of a nonlinear method. However, it should be noticed that if some factors are not properly controlled, then closed-form solutions do not promise accurate results. Again, finding factors that affect accuracy of a closed-form solution should rely on the theoretically error analysis.

Existing closed-form solutions for joint axis estimation can be classified into two categories according to the information they used. Methods from the first category use the pose measurements and will be referred to as the *pose methods*, e.g., Lenz and Tsai [10] and Young *et al.* [20], Zhuang [21] and Shih *et al.* [13], where a pose measurement contains both position and orientation estimates of a calibration object attached to the joint axis being calibrated. Notice that although different pose methods were proposed according to different problem formulations, [10], [13], [20] and [21], we found that these methods for calibrating a revolute joint axis are all equivalent, except for the Zhuang method [21]. The main difference of the Zhuang method comparing to others is that the translation parameters of all joints are estimated simultaneously to prevent the error propagation [21]. Nevertheless, when calibrating a robot of a single revolute joint, all the pose methods are equivalent.

The second category contains only one closed-form solution, i.e., Shih *et. al.*[14] (refer to Chen [3], Stone [17] and Sklar [16] for the corresponding nonlinear iterative solutions), which uses position measurements of a calibration point attached to the joint axis being calibrated. This kind of method will be referred to as the *point method*. Notice that when the joint under calibration is moving, the trajectory of the calibration point will form either a 3-D circle or a 3-D line according to its joint type (revolute or prismatic). Hence, the point method is essentially to solve 3-D circle and 3-D line fitting problems for revolute and prismatic joints, respectively. It is well known that a 3-D circle fitting problem is nonlinear when only 3-D measurements of circle edges are available [8] [12]. Whereas, we showed that if 3-D measurements of circle edges and their corresponding rotation angles are available, then we can derived a closed-form solution to the 3-D circle fitting problem (refer to [14]). Details of the above-mentioned closed-form solutions will be described in the next section.

Accurate closed-form solutions of kinematic parameters are very important for either subsequent nonlinear optimization or direct applications; hence, we will focus on the error analysis of closed-form solutions for kinematic calibration problems. However, since error analysis results of an overall kinematic model are usually dedicated to a specific type of robots, it is hard to generalize the analysis results to others. On the other hand, because most robots are composed of several prismatic and revolute joints, error analysis results on calibration of a single joint will be generally useful. Therefore, we will further focus on the error analysis of closed-form solutions when they are applied to the calibration problem of a single revolute or prismatic joint.

B. Review of Related Work

So far as we know, less work has been devoted to the investigation of error analysis on kinematic calibration; especially, on analyzing the parameter estimation error of closed-form solutions for kinematic calibration. Stone [17] analyzed the estimation error of his calibration method using point measurements; however, in his analysis, influence of some factors were obtained from fitting empirical data to parameterized functions. Lenz and Tsai showed some analysis results on estimation error of a rotation axis [10]; however, their error analysis results are mainly for hand-eye calibration.

In this paper, we will derive error variances of the estimated kinematic parameters of a single joint for the pose method and point method, respectively. Notice that during kinematic calibration, calibration data are measured with respect to different joint values. For convenience, we will refer to the distribution range of joint values corresponding to the calibration data as the *calibration range* which is a subspace of the joint space. The derived error variances for the point method are functions of the calibration range, number of measurements, amount of measurement noise and amount of joint value noise. Furthermore, for the point method, if the joint under calibration is revolute, then the error variances are also functions of the distance between the calibration point and the revolute joint axis and length of the link corresponding to the joint under calibration. The derived error variances for a pose method are functions of the calibration range, number of measurements, amount of measurement noise and length of the link corresponding to the joint under calibration.

This paper is organized as follows. In section II, the pose methods and the point method for kinematic parameters are described; general equations for the pose methods will be derived to show that all the pose methods are equivalent. In section III, error analysis on the point method for kinematic calibration of a single joint is described. In section IV, error analysis on the pose method for kinematic calibration of a single joint is described. In section V, the derived error variances of the estimated kinematic parameters are verified by computer simulations. Conclusions are given in section VI.

II. Closed-form Solutions for Kinematic Calibration

Assume that the robot under calibration contains no closed-loop kinematic chain and has n joints. The kinematic equation of the n-joint robot is

$$ {}^{w}T_n = {}^{w}T_0 {}^{0}T_1 {}^{1}T_2 \ldots {}^{n-1}T_n, \tag{1} $$

where ${}^{i}T_j$ is the transformation matrix from frame $\{i\}$ to frame $\{j\}$. The transformation matrix between two consecutive joint frames can be described by different kinematic models, e.g., the D-H model [4], S-model [17] and the CPC (*Complete and Parametrically Continuous*) model [22]. According to the robotics conventions, the z-axis of a joint frame is defined by its joint axis. Hence, the general form of a kinematic model between two consecutive joint frames can be represented as follows:

$$ {}^{i-1}T_i = {}^{i-1}T \, Q\left(q_i\right) \, T_i, \tag{2} $$

where q_i is the joint value of the ith joint, ${}^{i-1}T$ and T_i are the kinematic-model-dependent constant transformation matrices (e.g., for a CPC kinematic model, ${}^{i-1}T$ is an identity matrix and T_i is the shape matrix [22]), and

$$ Q\left(q_i\right) = \begin{cases} Rot_Z\left(q_i\right), & \text{if joint } i \text{ is revolute,} \\ Trans_Z\left(q_i\right), & \text{if joint } i \text{ is prismatic.} \end{cases} \tag{3} $$

Notice that by substituting equation (2) into equation (1), we have

$$ {}^{w}T_n = {}^{w}T_0 \, Q(q_1) \, V_1 \, Q(q_2) \, V_2 \, \ldots \, Q(q_n) \, V_n, \tag{4} $$

where

$$ V_i = T_i \, {}^{i}T. \tag{5} $$

Equation (4) is exactly in the form of the CPC kinematic model [22]. Therefore, no matter what kind of kinematic model is used, during the kinematic calibration, it can always be transformed into the CPC kinematic model [22] (this is owing to the *completeness* property of the CPC model). Henceforth, we will use the CPC kinematic model in deriving the closed-form solutions and in the error analysis. The shape matrix, V_i, for the CPC kinematic model is

$$V_i = R_i \, Rot_z(\beta_i) \, Trans([l_{ix} \, l_{iy} \, l_{iz}]^t), \tag{6}$$

and

$$R_i = \begin{bmatrix} 1 - \frac{b_{ix}^2}{1+b_{iz}} & \frac{-b_{ix}b_{iy}}{1+b_{iz}} & b_{ix} & 0 \\ \frac{-b_{ix}b_{iy}}{1+b_{iz}} & 1 - \frac{b_{iy}^2}{1+b_{iz}} & b_{iy} & 0 \\ -b_{ix} & -b_{iy} & b_{iz} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{7}$$

Notice that the parameters, $\{\beta_i, l_{ix}, l_{iy}, l_{iz}\}$ and $\{\beta_i, l_{iz}\}$, are redundant for a prismatic and a revolute joint (refer to [22]), respectively.

Kinematic calibration of an n-joint serial robot is the process of estimating unknown kinematic parameters contained in equation (1). Kinematic calibration problems are nonlinear because unknown parameters of the n joints are all multiplied together. However, if we sequentially estimate kinematic parameters of one joint at a time, then closed-form solutions can be derived.

Both the point method and the pose method discussed in this paper can be applied to calibrate a serial robot sequentially from the end-effector to the base and vice versa. When calibrating a robot from its end-effector toward its base, the ith shape matrix, V_i, is calibrated based on the net motion of joint i. Whereas, when calibrating a robot from its base toward its end-effector, the ith shape matrix, V_i, is calibrated based on the net motion of joint $(i+1)$, instead. The pose method and the point method will be discussed in the following subsections for both prismatic and revolute joints, respectively. Also, the objective functions for deriving the closed-form solutions are described in the following subsections, which will also be used in the theoretical analysis.

A. Fundamental Calibration Equation of the Pose Methods

The goal of this subsection is to derive the fundamental calibration equation of the pose methods. Assume that the kinematic calibration is proceeded in the forward order from the base toward the end-effector. Without loss of generality, we assume that the kinematic parameters of joints 1–$(i-1)$ are all calibrated. To calibrate the shape matrix of joint i, all the un-calibrated joints except for joint $(i+1)$, i.e., joints $(i+2)$–n, should be kept still.

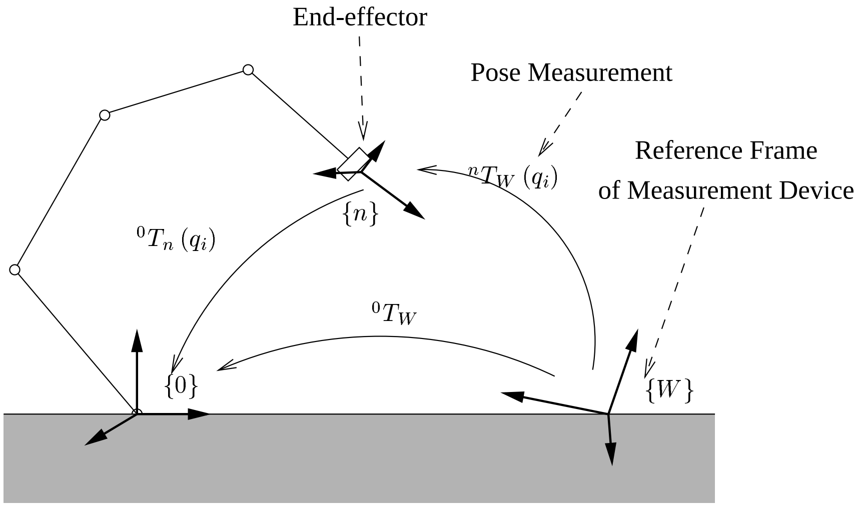

Fig. 1. A schematic diagram of the pose method.

Furthermore, we assume that the pose of the end-effector can be measured with respect to a fixed reference frame referred to as the WCS (*World Coordinate System*). Figure 1 shows the schematic diagram of the pose method for kinematic calibration. Obviously, the pose measurement of the end-effector, denoted as ${}^{n}T_w$, is a function of an n-dimensional vector of joint values denoted as $\mathbf{q}$. When joints 1–$(i+1)$ are moved to two different positions in the joint space, e.g., $\mathbf{q}(j)$ and $\mathbf{q}(k)$, we have

$$\begin{aligned} & F(\mathbf{q}(j))\, V_i\, Q(q_{i+1}(j))\, V_{(i+1)}\, {}^{i+1}T_n \\ &= F(\mathbf{q}(k))\, V_i\, Q(q_{i+1}(k))\, V_{(i+1)}\, {}^{i+1}T_n, \end{aligned} \tag{8}$$

where

$$F(\mathbf{q}(\cdot)) = {}^{n}T_w(\mathbf{q}(\cdot))\, {}^{w}T_{i-1}(q_i(\cdot))\, Q(q_i(\cdot)), \tag{9}$$

and both sides of equation (8) are all equal to identity matrix.

Since joints $(i+2)$–n were held still, $V_{(i+1)}\, {}^{i+1}T_n$ on both sides of the above equation can be eliminated, and equation (8) can be rewritten as follows:

$$V_i\, Q(\Delta q_{jk}) = \Delta T^{f}_{jk}\, V_i, \tag{10}$$

where $\Delta q_{jk} = q_{i+1}(j) - q_{i+1}(k)$ and

$$\Delta T^{f}_{jk} = F^{-1}(\mathbf{q}(j))\, F(\mathbf{q}(k)). \tag{11}$$

Notice that in the above equation, all the unknowns are contained in the shape matrix, V_i, whereas the other matrices can be computed from the calibrated kinematic model of joints 1–$(i-1)$ and the pose measurements.

Similarly, when the kinematic calibration is proceeded in the reverse order from the end-effector toward the base, the fundamental equation for the kinematic calibration can be derived as follows (refer to [13]):

$$Q(q_i(j) - q_i(k))\, V_i = V_i\, \Delta T^r_{jk}, \tag{12}$$

where

$$\Delta T^r_{jk} = {}^iT_n(\mathbf{q}(j))\, {}^nT_w(\mathbf{q}(j))\ \left[{}^iT_n(\mathbf{q}(k))\, {}^nT_w(\mathbf{q}(k))\right]^{-1}. \tag{13}$$

Notice that equation (12) can be transformed to be exactly in the same form of equation (10) by computing its inverse matrix. Furthermore, calibrating the kinematic parameters in the order from the base toward end-effectors is very suitable for robots having multiple end-effectors. On the contrary, if we calibrate a multiple end-effector robot from the end-effectors to the base then at the link having two branching kinematic chains, we will have to estimate an additional transformation matrix for unifying the coordinate systems from different end-effectors (as in [15]). Therefore, we will choose to analyze the estimation error of equation (10), which can be used to calibrate a robot in the forward order.

B. Pose Method: Calibration of a Prismatic Joint

For a prismatic joint, the unknown parameters to be estimated are the orientation of the prismatic joint axis, whereas the translation parameters are all redundant (refer to [22] and [13]). Assume that we have M pose measurements. Based on equation (10), an objective function of the unit vector of the translation joint axis can be defined as follows (refer to [13]):

$$\epsilon\,(u_i) = \sum_{j=1}^{M}\sum_{k=1}^{M} \left\| u_i\, \Delta q_{jk} - t_{\Delta T_{jk}} \right\|^2, \tag{14}$$

where u_i denotes the unit vector of joint axis satisfying the constraint, $\|u_i\|^2 = 1$, and $t_{\Delta T_{jk}}$ is the translation vector of the transformation matrix ΔT_{jk}.

The closed-form solution of u_i minimizing the objective function (14) can be derived as follows (refer to [13]):

$$u_i = \frac{\sum_{j=1}^{M} \sum_{k=1}^{M} \left(t_{\Delta T_{jk}} \Delta q_{jk} \right)}{\left\| \sum_{j=1}^{M} \sum_{k=1}^{M} \left(t_{\Delta T_{jk}} \Delta q_{jk} \right) \right\|} \tag{15}$$

C. Pose Method: Calibration of a Revolute Joint

For a revolute joint, the unknown parameters to be estimated are two orientation parameters and two translation parameters. From the rotation matrices of equation (10), we can define an objective function of the unit orientation vector as follows (refer to [13]):

$$\epsilon\left(u_i\right) = u_i^t \, E \, u_i, \tag{16}$$

where

$$u_i = R_i \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \tag{17}$$

R_i is the rotation matrix of the shape matrix, V_i, and

$$E = \sum_{j=1}^{M} \sum_{k=1}^{M} \left[R_{\Delta T_{jk}} - I \right]^t \left[R_{\Delta T_{jk}} - I \right], \tag{18}$$

where $R_{\Delta T_{jk}}$ is the rotation matrix of the transformation matrix, ΔT_{jk}. Notice that minimizing the objective function (16) is equivalent to computing the common rotation axis of a set of rotation matrices. Actually, all the existing pose method for the kinematic calibration of a revolute joint are doing in this way. The optimal solution to the objective function (16) is the eigenvector of the matrix, E, corresponding to the smallest eigenvalue. From the estimated orientation vector of the joint axis, u_i, into equation (7), we can construct the orientation matrix, R_i (refer to [22] or [13]).

By substituting the estimated orientation matrix, R_i, into equation (10), we have a equation of the unknown translation vector, t_i, where

$$\left[I - Rot_Z\left(\Delta q_{jk}\right) \right] \, l_i = R_i^t \, t_{\Delta T_{jk}}. \tag{19}$$

The objective function of the translation parameters can be define as follows:

$$\epsilon\left(t_i\right) = \sum_{j=1}^{M} \sum_{k=1}^{M} \left\| \begin{bmatrix} 1 - \cos\left(\Delta q_{jk}\right) & \sin\left(\Delta q_{jk}\right) \\ -\sin\left(\Delta q_{jk}\right) & 1 - \cos\left(\Delta q_{jk}\right) \end{bmatrix} l_i' - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} R_i^t \, t_{\Delta T_{jk}} \right\|^2, \tag{20}$$

where

$$l'_i = \begin{bmatrix} l_{ix} \\ l_{iy} \end{bmatrix}, \tag{21}$$

and the relation between the kinematic parameters, l_i, and the translation vector, t_i, of the shape matrix, V_i, is $t_i = R_i\, l_i$. The closed-form solution of the unknown translation parameter, t_i, can be computed by using linear least-square method based on equation (20).

D. Fundamental Equation of the Point Method

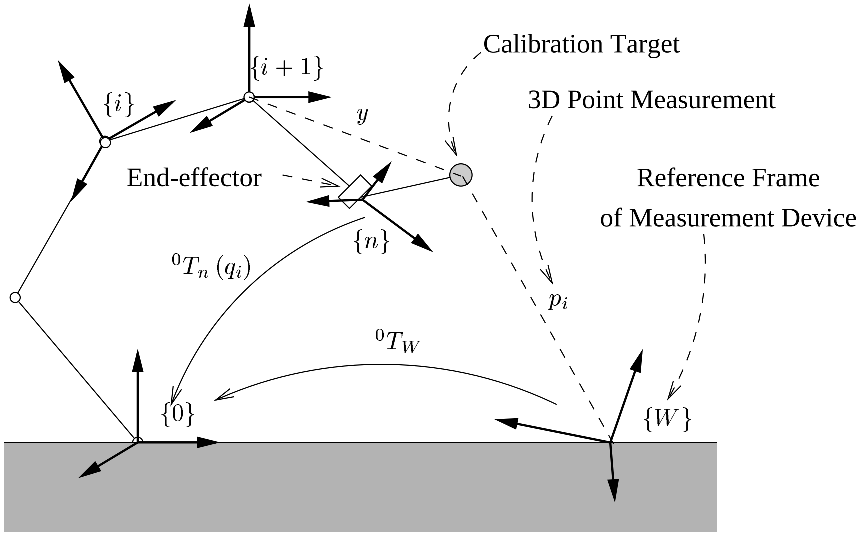

Fig. 2. A schematic diagram of the point method.

In this subsection, we will derive the fundamental equation for the point method. Suppose that the kinematic calibration is proceeded from the base toward the end-effector. Without loss of generality, we assume that the kinematic parameters of joints 1–$(i-1)$ are all calibrated, and that of joint i is being calibrated. Figure 2 shows the schematic diagram of the point method for robot kinematic calibration in the forward direction (from the base toward the end-effector). To provide position measurements for calibrating joint i, a calibration target is attached to any link between joint i and (including) the end-effector (notice that when calibrating different joints, the calibration target can be mounted at different places). For deriving a closed-form solution, when calibrating joint i, the un-calibrated joints which will affect the position of the calibration target should be kept still. For example, if the calibration target is mounted on the end-effector, then all the un-calibrated joints, i.e., joints $(i+1)$–n should be held still.

Let the unknown 3-D coordinates and the measured 3-D coordinates of the calibration target with respect to the $(i+1)$st joint frame and the WCS be denoted as y and ${}^{W}p$, respectively. By transforming the 3-D coordinates of the calibration target to the ith joint frame, we have

$$ {}^{0}T_i^{-1}\,{}^{0}T_W\,{}^{W}p = V_i\,Q(q_{i+1})\,V_{i+1}\,y, \tag{22} $$

where ${}^{0}T_i$ and ${}^{0}T_W$ can be obtained from the calibrated kinematic parameters, and V_i, V_{i+1} and y are unknown parameters. By combining V_{i+1} and y into one unknown 3-D position and substituting it into equation (22), we have

$$ {}^{i}p = V_i\,Q(q_{i+1})\,x, \tag{23} $$

where ${}^{i}p = {}^{0}T_i^{-1}\,{}^{0}T_W\,{}^{W}p$ and $x = V_{i+1}\,y$. Also, when the robot kinematic calibration is proceeded in the reverse order, i.e., calibrating from the end-effector toward the base, an equation similar to equation (23) can be easily derived. However, due to the drawback of calibrating a robot having multiple end-effectors in the reverse order described in section II-A, we will focus on the calibration method in the forward calibration order. From equation (23), it is obvious that ${}^{i}p$ is a function of q_{i+1}. Moreover, if joint $(i+1)$ is revolute, then the trajectory of ${}^{i}p$ subject to change of q_{i+1} will form a 3-D circle; otherwise, it will form a 3-D line when joint $(i+1)$ is prismatic.

E. Point Method: Estimation of a Prismatic Joint Axis

If joint $(i+1)$ is prismatic, then when it is moving, the trajectory of the calibration target will form a 3-D straight line. The 3-D line equation for the trajectory of the calibration target can be derived from the fundamental equation (23) as follows (refer to [14]):

$$ {}^{i}p = p_0 + q_{i+1}\,u_i, \tag{24} $$

where p_0 and u_i are the position and orientation of the 3-D line, respectively. It can be shown that, from u_i, the shape matrix, V_i, can be computed and then from V_i and p_0 the unknown vector, x, can be computed [14]. Hence, in order to estimate the shape matrix, the objective function is defined as follows:

$$\epsilon\left(p_0, u_i\right) = \sum_{j=1}^{M} \left\| {}^{i}p(j) - p_0 - q_{i+1}(j)\, u_i \right\|^2 , \tag{25}$$

where u_i is a unit vector and M is the number of measurements. The closed-form solution of p_0 and u_i minimizing the objective function (24) can be derived as follows (refer to [14]):

$$u_i = \frac{\sum_{j=1}^{M} \left(\underline{q}_{i+1}(j)\, {}^{i}\underline{p}(j) \right)}{\left\| \sum_{j=1}^{M} \left(\underline{q}_{i+1}(j)\, {}^{i}\underline{p}(j) \right) \right\|}, \tag{26}$$

and

$$p_0 = {}^{i}\bar{p} - \bar{q}_{i+1}\, u_i, \tag{27}$$

where

$${}^{i}\bar{p} = \frac{1}{M} \sum_{j=1}^{M} {}^{i}p(j), \tag{28}$$

$$\bar{q}_{i+1} = \frac{1}{M} \sum_{j=1}^{M} q_{i+1}(j), \tag{29}$$

$${}^{i}\underline{p}(j) = {}^{i}p(j) - {}^{i}\bar{p}, \tag{30}$$

and

$$\underline{q}_{i+1}(j) = q_{i+1}(j) - \bar{q}_{i+1}. \tag{31}$$

F. Point Method: Estimation of a Revolute Joint Axis

If joint $(i+1)$ is revolute, then when it is rotating, the trajectory of the calibration target will form a 3-D circle. Equation for describing the 3-D circle can be derived from equation (23). The 3-D circle equation derived from equation (23) contains seven unknowns including two position and two orientation parameters of the revolute joint axis and the unknown 3-D coordinates of the calibration target, i.e., x, in equation (23). Directly estimating the seven parameters is a nonlinear problem (refer to [17] and [16]). Nevertheless, we showed that by decomposing the unknown 3-D coordinates of the calibration target as follows (refer to [14]):

$$x = Rot_Z(\alpha_x) \begin{bmatrix} \rho \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ t_z \end{bmatrix}, \tag{32}$$

the original problem can be transformed to the optimization problem of the following objective function:

$$e\left(R, t, \rho\right) = \sum_{i=1}^{M} \left\| R_x\, {}^{i}p(j) + t - \rho r(j) \right\|^2 , \tag{33}$$

where ρ is the radius of the 3-D circle, R_x is a rotation matrix containing the orientation of the joint axis, t is the center of the 3-D circle and $r(j) = [\cos(q_{i+1}(j))\ \ \sin(q_{i+1}(j))\ 0]^t$. Closed-form solution of equation (33) can be computed by using a method, refer to [14], which is similar to the Umeyama method [18].

III. Theoretical Error Analysis on the Point Method

In this section, we will derive the covariance matrices of the estimated kinematic parameters for the point method. While the error analysis on the pose method is more complicated and will be discussed in the next section. Notice that when using the point method for kinematic calibration, each measurement will contribute one 3-D vector equation (see equation (23)). Furthermore, when calibrating a joint with the point method, it is essentially to fit a 3-D line or a 3-D circle according to the joint type. Therefore, the calibration data should be acquired uniformly from the trajectory of either a 3-D line or of a 3-D circle to have accurate 3-D line or 3-D circle fitting results. To derive the covariance matrix of the estimated orientation of an revolute joint, we need the following lemma.

Lemma 1: A rotation matrix, δR, constructed by using three small XYZ Euler angles, $\delta\phi_x$, $\delta\phi_y$ and $\delta\phi_z$, can be approximated as follows:

$$\delta R = I + Skew\left[\delta\Phi\right], \tag{34}$$

where

$$\delta\Phi = \begin{bmatrix} \delta\phi_x \\ \delta\phi_y \\ \delta\phi_z \end{bmatrix}, \tag{35}$$

and

$$Skew\left[\delta\Phi\right] = \begin{bmatrix} 0 & \delta\phi_z & -\delta\phi_y \\ -\delta\phi_z & 0 & \delta\phi_x \\ \delta\phi_y & -\delta\phi_x & 0 \end{bmatrix}. \tag{36}$$

Lemma 2: If x_i, $i = 1, 2, ..., M$ are uniformly distributed within the region, $\left[\frac{-X}{2}, \frac{+X}{2}\right]$, then

$$\sum_{i=1}^{M} f(x_i) \approx \frac{M}{X} \int_{\frac{-X}{2}}^{\frac{X}{2}} f(x)\, dx,$$

where $f(x)$ is a smooth function of x.

A. Error Analysis on the Point Method for a Prismatic Joint

The derived covariance matrix of the orientation error for a prismatic joint is based on the following assumptions:

1. The 3-D measurement noise is white Gaussian with zero mean and diagonal covariance matrix, $\sigma^2 I$.
2. The error in joint value (*i.e., the encoder error*) of the prismatic joint is negligible comparing to the 3-D measurement noise.

Define the estimation error of the kinematic parameters of a prismatic joint as follows:

$$\delta u_i = \hat{u}_i - u_i, \tag{37}$$

where $\hat{u}_i$ and u_i are respectively the estimated and true kinematic parameters for a prismatic joint i. From equation (26), we have

$$u_i = \frac{\sum_{j=1}^{M} \left(\underline{q}_{i+1}(j) \left({}^{i}\underline{p}(j) + \delta\underline{p}(j)\right)\right)}{\left\|\sum_{j=1}^{M} \left(\underline{q}_{i+1}(j) \left({}^{i}\underline{p}(j) + \delta\underline{p}(j)\right)\right)\right\|}, \tag{38}$$

where $\delta\underline{p}(j) = \delta p(j) - \delta\bar{p}$.

For convenience, define the *composite vector* and the *composite noise vector* as follows:

$$P = \sum_{j=1}^{M} \left[\underline{q}_{i+1}(j)\, {}^{i}\underline{p}(j)\right] \tag{39}$$

and

$$\delta P = \sum_{j=1}^{M} \left[\underline{q}_{i+1}(j)\, \delta\underline{p}(j)\right] \tag{40}$$

Substituting equations (38) and (39) into (40), we have

$$\hat{u}_i = \frac{P + \delta P}{||P + \delta P||}. \tag{41}$$

It can be shown that if the amount of composite noise vector, δP, is relatively small comparing to the composite vector, P, the denominator of equation (41) can be represented as follows:

$$\frac{1}{\|P+\delta P\|} = \frac{1}{\|P\|}\left[1 - \frac{\delta P^t\, P}{\|P\|^2}\right] + O(2) \tag{42}$$

Substituting equation (42) into (41), we have

$$\hat{u}_i = u_i + \delta\beta - (u_i^t\, \delta\beta)\, u_i + O(2), \tag{43}$$

where $\delta\beta = \frac{\delta P}{\|P\|}$, is the effective noise vector, and $u_i = \frac{P}{\|P\|}$ is the true kinematic parameters since P is noise free.

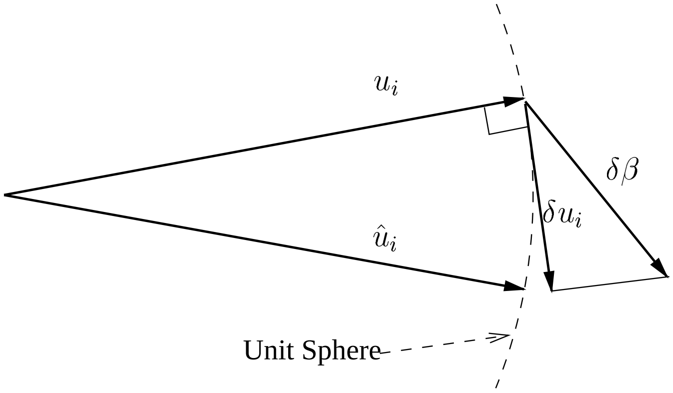

Fig. 3. Orientation parameter estimation error of a prismatic joint.

From equations (37) and (43), we have

$$\delta u_i = \delta\beta - (u_i^t\, \delta\beta)\, u_i + O(2). \tag{44}$$

Notice that in equation (44), δu_i is approximately equal to the component of $\delta\beta$ perpendicular to u_i as shown in Figure 3. Therefore, the size of δu_i is proportional to and bounded by the size of $\delta\beta$, and minimizing the amount of $\delta\beta$ is equivalent to minimizing the amount of parameter estimation error. Hence, we would derive the covariance matrix of the random vector $\delta\beta$ to find factors affecting the calibration accuracy.

Suppose that in the calibration process, the joint values corresponding to the M measurements were uniformly distributed within the region, $\{Q_1,\ Q_1 + \Delta Q\}$, i.e.,

$$q_{i+1}(j) = Q_1 + (j-1)\frac{\Delta Q}{M-1}. \tag{45}$$

The covariance matrix of $\delta\beta$ can be derived as follows (refer to AppendixVII-A).

$$Cov\,[\delta\beta] = \sigma^2_{\delta\beta}\, I, \tag{46}$$

where

$$\sigma^2_{\delta\beta} = \frac{\sigma^2_{\delta P}}{\|P\|^2} = \frac{12}{M}\frac{\sigma^2}{\Delta Q^2}. \tag{47}$$

According to equation (47), it is obvious that the orientation estimation error of a prismatic joint can be reduced by making ΔQ and M larger or by making the measurement noise smaller. Also, from equation (47), for minimizing the estimation error, increasing size of the calibration range, ΔQ, is more efficient than increasing the number of measurements, M.

B. Error Analysis on the Point Method for a Revolute Joint

In this subsection, we will derive the covariance matrix of the estimated parameters for a revolute joint. Notice that in this case, the encoder error in no longer negligible, because as the distance between the calibration target and the revolute joint axis becoming further, the encoder error will influence the calibration results more seriously.

Assume that the optimal parameters minimizing objective function (33) are as follows:

$$\begin{aligned} \hat{R}_x &= [I + Skew\,(\delta\Phi)]\; R_x, & (48)\\ \hat{\rho} &= \rho + \delta\rho, & (49)\\ \hat{t} &= t + \delta t, & (50) \end{aligned}$$

where $\delta\Phi$, $\delta\rho$ and δt are the parameter estimation error.

Define the error vector corresponding to the jth measurement, ${}^i p(j)$, as follows:

$$\epsilon_j = \hat{R}_x\, {}^i\hat{p}(j) + \hat{t} - \hat{\rho}\,\hat{r}(j) \approx 0 \tag{51}$$

By computing the Tyler series expansion of equation (51) and neglecting high order terms, we have the equation relating the parameter estimation error and the measurement noise as follows:

$$\epsilon_j = A_j\,\delta\Theta + R_x\,\delta p(j) + B_j\,\delta q(j) + O(2), \tag{52}$$

where

$$A_j = \frac{\partial \epsilon_j}{\partial \delta \Theta} \tag{53}$$

$$B_j = \frac{\partial \epsilon_j}{\partial \delta q(j)} \tag{54}$$

$\delta p(j)$ is the 3-D measurement error, $\delta q(j)$ is the encoder error and the error vector of the estimated parameters, and $\delta\Theta$ is defined as follows

$$\delta\Theta = \left[\delta\Phi^t \ \delta t^t \ \delta\rho\right]. \tag{55}$$

By substituting equation (52) into the objective function (33), the estimation error of the unknown parameters minimizing the objective function can be derived as follows:

$$\delta\Theta = -\left[\sum_{j=1}^{M} A_j^t \, A_j\right]^{-1} \left[\sum_{j=1}^{M} A_j^t \, R_x \, \delta p(j) + \sum_{j=1}^{M} A_j^t \, B_j \, \delta q(j)\right]. \tag{56}$$

Assume that the noisy point measurements in equation (33) is

$${}^i\hat{p}(j) = {}^ip(j) + \delta p(j), \tag{57}$$

where ${}^ip(j)$ is the true value and $\delta p(j)$ is a white Gaussian noise vector with zero mean and diagonal covariance matrix, $\sigma_p^2 I$, and that the encoder error, $\delta q(j)$, and the 3-D measurement error, $\delta p(j)$, are independent, where the encoder error is a zero mean Gaussian noise having variance, σ_q^2. From equation (56), the covariance matrix of the unknown parameters of $\delta\Theta$ can be derived as follows:

$$Var\left[\delta\Theta\right] = \sigma_p^2 \, \mathcal{A}^{-1} + \sigma_q^2 \, \mathcal{A}^{-1} \, \mathcal{B} \, \mathcal{A}^{-1}, \tag{58}$$

where

$$\mathcal{A} = \left[\sum_{j=1}^{M} A_j^t \, A_j\right], \tag{59}$$

and

$$\mathcal{B} = \left[\sum_{j=1}^{M} A_j^t \, B_j \, B_j^t \, A_j\right]. \tag{60}$$

Notice that A_j and B_j are matrix function of measurements, ${}^ip(j)$, the joint values, $q_{i+1}(j)$, for $j = 1, 2, ..., M$, and the parameters, ρ, R_x and t. Notice that the true 3-D measurement, ${}^ip(j)$, can be derived from equation (33) as follows:

$${}^ip(j) = R_x^{-1} \, \left(\rho \, r(j) - t\right). \tag{61}$$

The 3-D measurements, ${}^{i}p(j)$, in matrices A_j and B_j can be eliminated by substituting the above equation into A_j and B_j. Therefore, matrices A_j and B_j are now functions of the joint value, $q_{i+1}(j)$, and the parameters, ρ, R_x and t.

Since the origin of the joint value can be arbitrarily assigned in the CPC kinematic model [22], hence, without lost of generality, we can assume that the joint values, $q_{i+1}(j)$, for $j = 1, 2, ..., M$, are uniformly distributed within the region, $\left[\frac{-\Delta Q}{2}, \frac{+\Delta Q}{2}\right]$. Also, since A_j and B_j are smooth matrix functions of joint values, based on Lemma 2, we have

$$\mathcal{A} \approx \frac{M}{\Delta Q} \int_{\frac{-\Delta Q}{2}}^{\frac{\Delta Q}{2}} A^t(q)\, A(q)\, dq, \tag{62}$$

and

$$\mathcal{B} \approx \frac{M}{\Delta Q} \int_{\frac{-\Delta Q}{2}}^{\frac{\Delta Q}{2}} A^t(q)\, B(q)\, B^t(q)\, A(q)\, dq. \tag{63}$$

By using Mathematica, both of the approximation matrices of $\mathcal{A}$ and $\mathcal{B}$ in equations (62) and (63) can be easily computed in analytic form. The inverse of the matrix, $\mathcal{A}$, can also be computed using Mathematica in analytic form, and then we have the covariance matrix of the estimated parameters by substituting the approximated matrices, $\mathcal{A}^{-1}$ and $\mathcal{B}$, into equation (58).

Since the diagonal terms of the covariance matrix themselves are sufficient for describing the estimation error, hence we only list the variance terms of the seven parameters as follows (covariance terms are listed in Appendix-VII-B:

$$\sigma^2_{\delta\phi_x} = \frac{\sigma_p^2\, \Phi_x(\Delta Q)}{\rho^2\, M}, \tag{64}$$

$$\sigma^2_{\delta\phi_y} = \frac{\sigma_p^2\, \Phi_y(\Delta Q)}{\rho^2\, M}, \tag{65}$$

$$\sigma^2_{\delta\phi_z} = \frac{\sigma_p^2\, \Phi_z(\Delta Q)}{\rho^2\, M} + \frac{\sigma_q^2\, \Psi_1(\Delta Q)}{M}, \tag{66}$$

$$\sigma^2_{\delta t_x} = \frac{\sigma_p^2\, \Phi_z(\Delta Q)\, \left(\rho^2 + t_y^2\right)}{\rho^2\, M} + \frac{\sigma_p^2\, \Phi_y(\Delta Q)\, t_z^2}{\rho^2\, M} + \frac{\sigma_q^2\, t_y^2\, \Psi_1(\Delta Q)}{M} + \frac{\sigma_q^2\, \rho^2\, \Psi_2(\Delta Q)}{M}, \tag{67}$$

$$\begin{aligned}\sigma^2_{\delta t_y} = {} & \frac{\sigma_p^2\, \Phi_z(\Delta Q)\, \left(\rho^2 - 2\,\mathrm{sinc}\left(\frac{\Delta Q}{2}\right)\, t_x\, \rho + t_x^2\right)}{\rho^2\, M} + \frac{\sigma_p^2\, \Phi_x(\Delta Q)\, t_z^2}{\rho^2\, M} \\ & + \frac{\sigma_q^2\, t_x^2\, \Psi_1(\Delta Q)}{M} + \frac{\sigma_q^2\, \rho^2\, \Psi_3(\Delta Q)}{M} - \frac{\sigma_q^2\, t_x\, \rho\, \Psi_4(\Delta Q)}{M},\end{aligned} \tag{68}$$

$$\sigma^2_{\delta t_z} = \frac{\sigma_p^2\, \Phi_y(\Delta Q)\, \left(\frac{\Delta Q + \sin(\Delta Q)}{2\, \Delta Q} \rho^2 - 2\, \mathrm{sinc}\left(\frac{\Delta Q}{2}\right)\, t_x\, \rho + t_x^2\right)}{\rho^2\, M} + \frac{\sigma_p^2\, \Phi_x(\Delta Q)\, ty^2}{\rho^2\, M}, \tag{69}$$

$$\sigma^2_{\delta \rho} = \frac{\sigma_p^2\, \Phi_z(\Delta Q)}{M} + \frac{\sigma_q^2\, \rho^2\, \Psi_5(\Delta Q)}{M}, \tag{70}$$

where

$$\mathrm{sinc}\left(\frac{\Delta Q}{2}\right) = \frac{\sin\left(\frac{\Delta Q}{2}\right)}{\frac{\Delta Q}{2}}, \tag{71}$$

$$\Phi_x\left(\Delta Q\right) = \frac{2\, \Delta Q}{\Delta Q - \sin\left(\Delta Q\right)}, \tag{72}$$

$$\Phi_y\left(\Delta Q\right) = \frac{2\, \Delta Q^2}{-4 + \Delta Q^2 + 4\, \cos\left(\Delta Q\right) + \Delta Q\, \sin\left(\Delta Q\right)}, \tag{73}$$

$$\Phi_z\left(\Delta Q\right) = \frac{\Delta Q^2}{-2 + \Delta Q^2 + 2\, \cos\left(\Delta Q\right)}, \tag{74}$$

$$\Psi_1\left(\Delta Q\right) = \frac{-6\, \Delta Q^2 + 2\, \Delta Q^4 + 6\, \Delta Q^2\, \cos(\Delta Q) + 2\, \Delta Q\, \sin(\Delta Q) - \Delta Q\, \sin(2\Delta Q)}{2(-2 + \Delta Q^2 + 2\, \cos(\Delta Q))^2}, \tag{75}$$

$$\Psi_2\left(\Delta Q\right) = \frac{\Delta Q^4 - \Delta Q^3\, \sin(\Delta Q)}{2\left(-2 + \Delta Q^2 + 2\, \cos(\Delta Q)\right)^2}, \tag{76}$$

$$\Psi_3\left(\Delta Q\right) = \frac{-4\, \Delta Q^2 + \Delta Q^4 + 4\, \Delta Q^2\, \cos(\Delta Q) + \Delta Q^3\, \sin(\Delta Q)}{2(-2 + \Delta Q^2 + 2\, \cos(\Delta Q))^2}, \tag{77}$$

$$\begin{aligned}\Psi_4\left(\Delta Q\right) = &-\frac{-2\, \Delta Q^2\, \cos\left(\frac{\Delta Q}{2}\right) + 2\, \Delta Q^2\, \cos\left(\frac{3\, \Delta Q}{2}\right) + 24\, \Delta Q\, \sin\left(\frac{\Delta Q}{2}\right)}{2\left(-2 + \Delta Q^2 + 2\, \cos(\Delta Q)\right)^2} \\ &- \frac{-4\, \Delta Q^3\, \sin\left(\frac{\Delta Q}{2}\right) - 8\, \Delta Q\, \sin\left(\frac{3\, \Delta Q}{2}\right)}{2\left(-2 + \Delta Q^2 + 2\, \cos(\Delta Q)\right)^2},\end{aligned} \tag{78}$$

and

$$\Psi_5\left(\Delta Q\right) = \frac{2\, \Delta Q\, \sin^2\left(\frac{\Delta Q}{2}\right)\left(\Delta Q - \sin(\Delta Q)\right)}{\left(-2 + \Delta Q^2 + 2\, \cos(\Delta Q)\right)^2}. \tag{79}$$

The values of the three functions, $\Phi_x(\Delta Q)$, $\Phi_y(\Delta Q)$ and $\Phi_z(\Delta Q)$, are plotted in Figure 4. Whereas the values of the functions, $\Psi_1(\Delta Q)$, $\Psi_2(\Delta Q)$, $\Psi_3(\Delta Q)$, $\Psi_4(\Delta Q)$, and $\Psi_5(\Delta Q)$, are plotted in Figure 5. Notice that when ΔQ is small, $\Phi_y(\Delta Q)$ is much larger than the other two, which means that the orientation error about the y-axis will be worse than the other two axes. This is because that when the calibration range, ΔQ, is small, the trajectory of the calibration target will be close to a 3-D straight line segment parallel to the y-axis, which can be an arc of any 3-D circle having rotation axis perpendicular to the y-axis (see Figure 6).

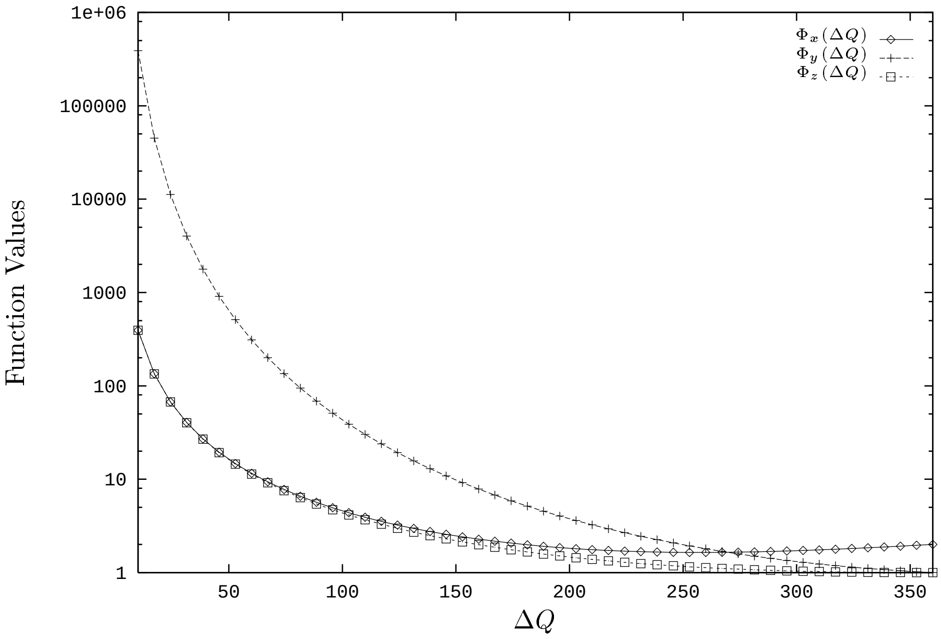

Fig. 4. Function values of $\Phi_x(\Delta Q)$, $\Phi_y(\Delta Q)$ and $\Phi_z(\Delta Q)$.

IV. Theoretical Error Analysis on the Pose Method

In this section, we will derive the covariance matrices of the estimated kinematic parameters for the pose method. Notice that when using the pose method, pose measurements have to be matched pairwise to be used in the calibration equation (10). In section II, matching pairwise of the pose measurements is preformed without considering the redundancy problem, which makes M^2 pairs from M measurements. Nevertheless, there is still one problem to be solved, i.e., how we select a set of configurations for the robot such that the corresponding pose measurements will yield a more accurate calibration result. Here we consider two kinds of data collection processes for the kinematic calibration. The first kind of data collection processes is to sample $\frac{M}{2}$ pose measurements at one configuration and $\frac{M}{2}$ pose measurements at another configuration, where M is the total number of calibration data. For example, suppose θ_0 and θ_1 are all belong to the working space of a revolute joint under calibration, then the first kind of data collection processes may acquire $\frac{M}{2}$ pose measurements at joint position θ_0 and acquire another $\frac{M}{2}$ pose measurements at joint position θ_1, which will make $\left(\frac{M}{2}\right)^2$ effective pairs. This kind of processes

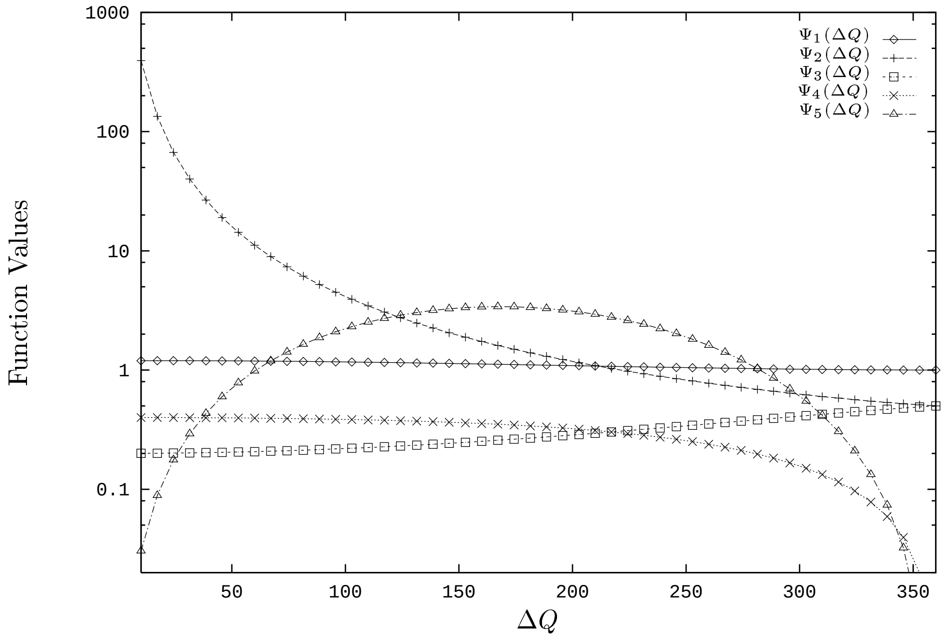

Fig. 5. Function values of $\Psi_1(\Delta Q)$, $\Psi_2(\Delta Q)$, $\Psi_3(\Delta Q)$, $\Psi_4(\Delta Q)$, and $\Psi_5(\Delta Q)$.

is intended to to maximize the net motion between the pairwise pose measurements such that the estimation accuracy can be improved (refer to [13]), therefore, these two calibration configurations in the joint space, i.e., θ_0 and θ_1, should be made apart as far as possible. While the second kind of data collection processes is to acquire uniformly distributed calibration data, i.e., the pose measurements are acquired according to a set of joint values uniformly distributed within a certain region. For convenience, we will refer to the first kind of collection processes as the two-configuration approach, and the second kind of collection processes as the uniform-configuration approach, respectively.

For deriving the covariance matrices of the estimated kinematic parameters, we need the following lemmas.

Lemma 3: If R is a rotation matrix, then

$$Skew(R^t\,\delta\Phi) = R^t\,Skew(\delta\Phi)\,R, \tag{80}$$

where $\delta\Phi$ is a small orientation error vector.

Lemma 4: Let δR_1 and δR_2 be two rotation matrices constructed by using small XYZ

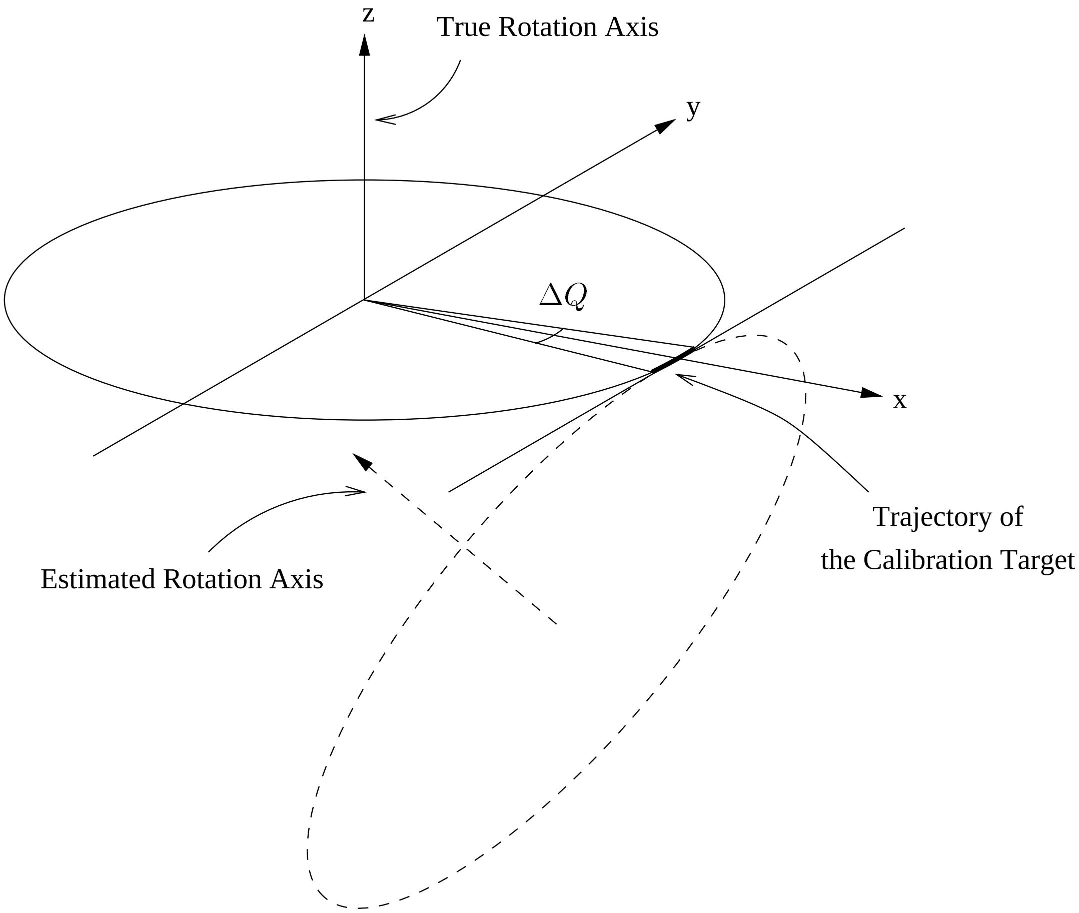

Fig. 6. Trajectory of the calibration target corresponding to small calibration range is close to a straight line.

Euler angles, $\delta\Phi_1$ and $\delta\Phi_2$, respectively. Then, their product can be represented as follows:

$$\delta R_1 \, \delta R_2 = I + Skew \left[\delta\Phi_1 + \delta\Phi_2\right]. \tag{81}$$

Lemma 5: Let $\hat{R}$ be an estimate of a rotation matrix, R, and

$$\hat{R} = \delta R_L \, R = R \, \delta R_R,$$

where δR_L and δR_R are error rotation matrices constructed by using small XYZ Euler angles, $\delta\Phi_L$ and $\delta\Phi_R$, respectively. Then, the orientation error, $\delta\Phi_L$ and $\delta\Phi_R$, can be related by the following equation:

$$\delta\Phi_R = R^t \delta\Phi_L \tag{82}$$

For convenience, we will refer to $\delta\Phi_L$ and $\delta\Phi_R$ as the *left orientation error* and *right orientation error*, respectively.

Lemma 6: If $\hat{R}_1 = R_1\,\delta R_1$ and $\hat{R}_2 = R_2\,\delta R_2$, then the right orientation error, $\delta\Phi_{12}$, of the matrix $\hat{R}_1^t\,\hat{R}_2$, can be computed as follows:

$$\delta\Phi_{12} = \left(\delta\Phi_2 - R_2^t\,R_1\,\delta\Phi_1\right), \tag{83}$$

Lemma 7: The rotation matrix of ΔT_{jk} defined following equation (11), i.e., $R_{\Delta T_{jk}}$, is equivalent to $R_i\;Rot_Z(\Delta q_{jk})\;R_i^t$.

<Proof>:

Recall that this rotation matrix is derived from equations (8)–(11). Notice that the both sides of equation (8) are equal to identity, hence,

$$F(\mathbf{q}(j)) = \left[V_i\;Q(q_{i+1}(j))\;V_{(i+1)}\;{}^{i+1}T_n\right]^{-1}, \tag{84}$$

and

$$F(\mathbf{q}(k)) = \left[V_i\;Q(q_{i+1}(k))\;V_{(i+1)}\;{}^{i+1}T_n\right]^{-1}. \tag{85}$$

By substituting the above two equations into equation (11), we have

$$\Delta T_{jk} = V_i\;Q(\Delta q_{jk})\;V_i^{-1}. \tag{86}$$

Lemma 8: If the right orientation error of the rotation matrices of $F(\mathbf{q}(j))$ and $F(\mathbf{q}(k))$ are $\delta\Phi_j$ and $\delta\Phi_k$, respectively, then the right orientation error, $\delta\Phi_{jk}$, of an estimate of $R_{\Delta T_{jk}}$ defined in Lemma 7 is as follows:

$$\delta\Phi_{jk} = \delta\Phi_j - R_i\;Rot_Z(-\Delta q_{jk})\;R_i^t\;\delta\Phi_k. \tag{87}$$

where $\delta\Phi'_j = R_i^t\,\delta\Phi_j$ and $\delta\Phi'_k = R_i^t\,\delta\Phi_k$

<Proof>:

From Lemma 6, we have the right orientation error:

$$\delta\Phi_{jk} = \delta\Phi_k - R_{\Delta T_{jk}}^t\delta\Phi_j. \tag{88}$$

By using Lemma 7, we have

$$\delta\Phi_{jk} = \delta\Phi_k - R_i\;Rot_Z(-\Delta q_{jk})\;R_i^t\;\delta\Phi_j \tag{89}$$

Lemma 9: An estimate of $R_{\Delta T_{jk}}$ can be formulated as follows:

$$\hat{R}_{\Delta T_{jk}} = R_i\;Rot_Z(\Delta q_{jk})\;\left[I + Skew(R_i^t\,\delta\Phi_{jk})\right]\;R_i^t. \tag{90}$$

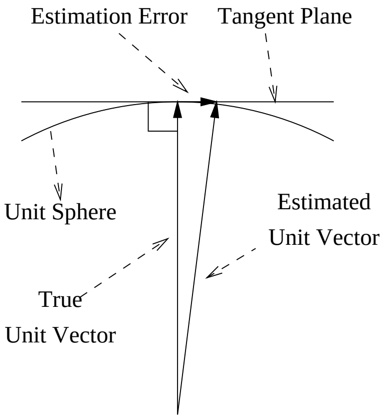

Fig. 7. Small estimation error of a unit vector.

Notice that the the above lemma can be easily derive based on Lemmas 3, 6 and 7.

Lemma 10: Small estimation error of a unit vector will fall in the tangent plane of the unit sphere perpendicular to the true unit vector (see Figure 7. Hence, when using the pose method, the orientation estimation error of a revolute joint can be described as follows:

$$\hat{u_i} = R_i \; [\delta u_x \; \delta u_y \; 1]^t \,, \tag{91}$$

where δu_x and δu_y are the error vector on the tangent plane perpendicular to u_i.

Lemma 11: The relation between the true value, estimation error and an estimate of the translation vector of the matrix,

$$\begin{bmatrix} \hat{R} & \hat{t} \\ 0 & 1 \end{bmatrix}^{-1},$$

is as follows,

$$-\hat{R}^t \, \hat{t} = -R^t \, t + Skew(\delta\Phi) \, R^t \, t - R^t \, \delta t, \tag{92}$$

where $\delta\Phi$ is the right orientation error of $\hat{R}$, and R and t are the true values of $\hat{R}$ and $\hat{t}$, respectively.

Lemma 12: Consider the following transformation matrix.

$${}^1T_2 = \begin{bmatrix} \hat{R}_1 & \hat{t}_1 \\ 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \hat{R}_2 & \hat{t}_2 \\ 0 & 1 \end{bmatrix},$$

The true and translation vector of the matrix, ${}^1T_2$, is

$$t_{12} = -R_1^t \, (t_1 - t_2) \tag{93}$$

and the error translation vector of ${}^{1}T_2$ is

$$\delta t_{12} = Skew(\delta\Phi_1)\, R_1^t\, (t_1 - t_2) - R_1^t\, (\delta t_1 - \delta t_2), \tag{94}$$

where $\delta\Phi_1$ is the right orientation error of $\hat{R}_1$ and R and t are the true values of R and t, respectively.

Lemma 13: For any $\theta \in \Re$, $z \in \Re$ and $x \in \Re^3$, the following equality will hold:

$$Rot_Z(\theta)\; Skew[x] \begin{bmatrix} 0 \\ 0 \\ z \end{bmatrix} = Skew\left[Rot_Z(\theta)\, x\right] \begin{bmatrix} 0 \\ 0 \\ z \end{bmatrix}. \tag{95}$$

Lemma 14: Let

$$q_j = \begin{cases} q_1, & 1 \le j \le \frac{M}{2}, \\ q_1 + \Delta Q, & \frac{M}{2} < j \le M. \end{cases}$$

Suppose that $K(\Delta q_{jk})$ is a matrix function of $\Delta q_{jk} = q_j - q_k$, and that $K(0) = 0$. The following equations will hold:

$$\sum_{j=1}^{M}\sum_{k=1}^{M} K(q_j - q_k) = \left(\frac{M}{2}\right)^2 \left[K(\Delta Q) + K(-\Delta Q)\right], \tag{96}$$

$$\begin{aligned} &\sum_{j=1}^{M}\left[\left(\sum_{k=1}^{M} K(q_j - q_k)\right)\left(\sum_{k=1}^{M} K^t(q_j - q_k)\right)\right] \\ &= \left(\frac{M}{2}\right)^3 \left[K(\Delta Q)\, K^t(\Delta Q) + K(-\Delta Q)\, K^t(-\Delta Q)\right] \end{aligned} \tag{97}$$

A. Error Analysis on the Pose Method for a Prismatic Joint

The process for deriving the orientation estimation error of a prismatic joint using the pose method is similar to that described in section III-A. We will first derive the composite vector, P, and the effective error vector, δP, as what we did in section III-A. Similarly, we assume that the encoder error is negligible comparing to the position measurement error. However, when using pose measurements, the orientation error contained in the pose measurements will also influence the calibration accuracy. In this case, the effective noise vector (derived from equation (10) and Lemma 12) will be

$$\delta P = \sum_{j=1}^{M}\sum_{k=1}^{M}\left[\Delta q_{jk}^2\; Skew(u_i)\, \delta\Phi_j - 2\, \Delta q_{jk}\, R_j^t\, \delta t_j\right]. \tag{98}$$

Assume that the orientation error, $\delta\Phi_j$, and the translation error, δt_j, are independent. The covariance matrix of the effective error vector will be

$$Cov\,(\delta P) = \sum_{j=1}^{M} \left\{ \sigma_{\delta\Phi}^2 \, \left[\sum_{k=1}^{M} \Delta q_{jk}^2 \; Skew(u_i) \right] \left[\sum_{k=1}^{M} \Delta q_{jk}^2 \; Skew(u_i) \right]^t + 4\,\sigma_{\delta t}^2 \, \left[\sum_{k=1}^{M} \Delta q_{jk} \right]^2 \; I \right\}. \tag{99}$$

Also, in this case, the composite vector of the position measurements, $t_{\Delta T_{jk}}$, will be

$$P = \sum_{j=1}^{M} \sum_{k=1}^{M} \Delta q_{jk} \; t_{\Delta T_{jk}}. \tag{100}$$

Notice that in the above equation, Δq_{jk} and $t_{\Delta T_{jk}}$ are all true values. Hence, for each j and k, $t_{\Delta T_{jk}}$ is in the direction, u_i, and its length is Δq_{jk}.

B. Covariance Matrix of the Calibration Error of a Prismatic Joint Using Two-Configuration Approach

In this subsection, we will derive the covariance matrix of the effective noise vector, $\delta\beta$. Suppose that there are totally M measurements. When using the two-configuration approach for calibrating a prismatic joint, half of the measurements will correspond to a joint value, q_1, and the others will correspond to another joint value, $(q_1 + \Delta Q)$. The covariance matrix of the error vector, δP, can be computed from equation (99) and Lemma 14 as follows:

$$Cov(\delta P) = -\frac{M^3}{4} \, \sigma_{\delta\Phi}^2 \, \Delta Q^4 \; Skew^2(u_i) + M^3 \, \sigma_{\delta t}^2 \, \Delta Q^2 \; I. \tag{101}$$

Also, based on Lemma 14 and from equation (100), we have that the length of P will be

$$\|P\| = \frac{M^2}{4} \, \Delta Q^2. \tag{102}$$

Hence, the covariance matrix of the effective noise vector, $\delta\beta$, is

$$Cov(\delta\beta) = \frac{Cov(\delta P)}{\|P\|^2} = -\frac{4\,\sigma_{\delta\Phi}^2}{M} \; Skew^2(u_i) + \frac{\sigma_{\delta t}^2}{M\,\Delta Q^2} \; I. \tag{103}$$

C. Covariance Matrix of the Calibration Error of a Prismatic Joint Using Uniform-Configuration Approach

In this subsection, we will derive the covariance matrix of the effective noise vector, $\delta\beta$. Suppose that there are totally M measurements. When using the uniform-configuration approach for calibrating a revolute joint, the joint values corresponding to the M measurements are uniformly distributed within a region, e.g., $[q_1, q_1 + \Delta Q]$. Based on Lemma 2, the covariance matrix of the error vector, δP, can be computed from equation (99), by using Mathematica, as follows:

$$Cov(\delta P) = -\frac{M^3}{30}\,\sigma^2_{\delta\Phi}\,\Delta Q^4\,Skew^2(u_i) + \frac{M^3}{3}\,\sigma^2_{\delta t}\,\Delta Q^2\,I. \tag{104}$$

Similarly, the length of the composite vector, P, is

$$\|P\| = \frac{\Delta Q^2\,M^2}{6}. \tag{105}$$

Therefore, the covariance matrix of the effective noise vector, $\delta\beta$, is

$$Cov(\delta\beta) = \frac{Cov(\delta P)}{\|P\|^2} = -\frac{6\,\sigma^2_{\delta\Phi}}{5\,M}\,Skew^2(u_i) + \frac{12\,\sigma^2_{\delta t}}{M\,\Delta Q^2}\,I. \tag{106}$$

D. Error Analysis on the Pose Method for a Revolute Joint

In this subsection, we will derive some general equations about the orientation and position estimation error to be used in the next two subsections. By using Lemmas 1–10, the objective function (16) can be simplified as follows:

$$\epsilon\,(u_i) = \sum_{j=1}^{M}\sum_{k=1}^{M}\left\|\left\{Rot_Z(\Delta q_{jk})\left[I + Skew(R_i^t\,\delta\Phi_{jk})\right] - I\right\}\,(e_Z + \delta u)\right\|^2, \tag{107}$$

where $e_Z = [0\ 0\ 1]^t$ and $\delta u = [\delta u_x\ \delta u_y\ 0]^t$. Notice that e_Z is invariant with respect to the transformation, $Rot_Z(\cdot)$. Also, high order terms of the estimation error in equation (107) is usually very small comparing to other terms, hence, they can be neglected. Therefore, the above equation can be further simplified as follows:

$$\epsilon\,(u_i) = \sum_{j=1}^{M}\sum_{k=1}^{M}\left\|\left[Rot_Z(\Delta q_{jk}) - I\right]\,\delta u + Rot_Z(\Delta q_{jk})\,Skew(\delta\Phi'_{jk})\,e_Z\right\|^2, \tag{108}$$

where $\delta\Phi'_{jk} = R_i^t\,\delta\Phi_{jk}$. The optimal solution, δu, minimizing equation (108) will satisfying the following normal equation:

$$\left\{\sum_{j=1}^{M}\sum_{k=1}^{M}\left[Rot_Z(\Delta q_{jk}) - I\right]^t\left[Rot_Z(\Delta q_{jk}) - I\right]\right\}\,\delta u$$

$$= -\sum_{j=1}^{M}\sum_{k=1}^{M}\left[Rot_Z(\Delta q_{jk}) - I\right]^t\,Rot_Z(\Delta q_{jk})\,Skew(\delta\Phi'_{jk})\,e_Z \tag{109}$$

By simplifying the above equation, we have

$$\begin{aligned}
&\sum_{j=1}^{M}\sum_{k=1}^{M} 2(1-\cos(\Delta q_{jk}) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \delta u \\
&= \sum_{j=1}^{M}\sum_{k=1}^{M} \left[Rot_Z(\Delta q_{jk}) - I\right] \, Skew(\delta\Phi'_j) \, e_Z \\
&+ \sum_{j=1}^{M}\sum_{k=1}^{M} \left[Rot^t_Z(\Delta q_{jk}) - I\right] \, Skew(\delta\Phi'_k) \, e_Z,
\end{aligned} \tag{110}$$

where $\delta\Phi'_j = R^t_i \, \delta\Phi_j$ and $\delta\Phi'_k = R^t_i \, \delta\Phi_k$. Notice that by definition, $\Delta q_{jk} = -\Delta q_{kj}$, hence, these two terms on the right hand side of equation (110) are actually the same (it can be easily proved by swap the indices j and k). As a result, equation (110) can be further simplified as follows:

$$\begin{aligned}
&\sum_{j=1}^{M}\sum_{k=1}^{M} 2(1-\cos(\Delta q_{jk}) \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \delta u \\
&= 2\sum_{j=1}^{M}\sum_{k=1}^{M} \left[Rot_Z(\Delta q_{jk}) - I\right] \, Skew(\delta\Phi'_k) \, e_Z
\end{aligned} \tag{111}$$

Suppose that the orientation error of the pose measurements are white Gaussian noise with zero mean, variance, σ^2, and independently identical distributions. We have that the orientation estimation error of the revolute joint axis will be

$$\sigma^2_{\delta u} = \frac{\sigma^2 \sum_{j=1}^{M} \left[\left(\sum_{k=1}^{M} \cos(\Delta q_{jk}) - 1 \right)^2 + \left(\sum_{k=1}^{M} \sin(\Delta q_{jk}) \right)^2 \right]}{\left[\sum_{j=1}^{M} \sum_{k=1}^{M} (1 - \cos(\Delta q_{jk})) \right]^2} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{112}$$

Notice that the orientation error, δu, of the ith joint axis will induce a right orientation error, $\delta\Phi_{R_i}$, of the orientation matrix, R_i, of the shape matrix as follows (refer to AppendixVII-C for the proof):

$$\delta\Phi_{R_i} = \begin{bmatrix} \delta u_x \\ \delta u_y \\ \frac{\delta u_y \, b_{ix} - \delta u_x \, b_{iy}}{1 + b_{iz}} \end{bmatrix}. \tag{113}$$

We will now derive the translation estimation error for a revolute joint when using the pose method. When considering the measurement noise and encoder noise, equation (19) can be written as follows:

$$\left[I - Rot_Z\left(\Delta\hat{q}_{jk}\right)\right]\ \hat{l}_i = \hat{R}_i^t\ \hat{t}_{\Delta T_{jk}}. \tag{114}$$

By substituting the following equations,

$$\begin{aligned}
\hat{l}_i &= l_i + \delta l_i, & (115)\\
\Delta\hat{q}_{jk} &= \Delta\hat{q}_{jk} + \delta q_{jk}, & (116)\\
\hat{R}_i &= R_i\ \left[I + \delta\Phi_{R_i}\right], & (117)\\
\hat{t}_{\Delta T_{jk}} &= t_{\Delta T_{jk}} + \delta t_{\Delta T_{jk}}, & (118)
\end{aligned}$$

into equation (114) and neglect the high order term, we have

$$A_{jk}\delta l_i = B_{jk}\delta q + C_{jk}\ \delta\Phi_{R_i} + R_i^t\ \delta t_{\Delta T_{jk}}, \tag{119}$$

where $\delta q = \left[0\ 0\ \delta q_{jk}\right]^t$ is the encoder error, and

$$\begin{aligned}
A_{jk} &= \left[I - Rot_Z\left(\Delta q_{jk}\right)\right], & (120)\\
B_{jk} &= -Skew\left[Rot_Z\left(\Delta q_{jk}\right)\ l_i\right], & (121)\\
C_{jk} &= +Skew\left[\left[I - Rot_Z\left(\Delta q_{jk}\right)\right]\ l_i\right]. & (122)
\end{aligned}$$

From Lemma 12 and equation (119), we have

$$A_{jk}\delta l_i = B_{jk}\delta q + C_{jk}\ \left(\delta\Phi_{R_i} - \delta\Phi_j\right) + D_{ij}\ \left(\delta t_j - \delta t_k\right), \tag{123}$$

where $D_{ij} = -R_i^t R_j^t$. In general, in the above equation, the amount of the orientation estimation error of the joint axis, $\delta\Phi_{R_i}$, is usually much smaller than that of the measurement noise, i.e., $\delta\Phi_j$. Therefore, $\delta\Phi_{R_i}$ is omitted in the following derivation, and the objective function (20) can be reformulated as follows:

$$\epsilon\left(\delta l_i\right) = \sum_{j=1}^{M}\sum_{k=1}^{M}\left\|A_{jk}\delta l_i - B_{jk}\delta q + C_{jk}\ \delta\Phi_j - D_{ij}\ \left(\delta t_j - \delta t_k\right),\right\|^2. \tag{124}$$

The solution of δl_i minimizing equation 124, can be derived as follows:

$$\delta l_i = \left[\sum_{j=1}^{M}\sum_{k=1}^{M} A_{jk}^t\ A_{jk}\right]^{-1}\left[\sum_{j=1}^{M}\sum_{k=1}^{M} A_{jk}^t\ B_{jk}\delta q - A_{jk}^t\ C_{jk}\ \delta\Phi_j + A_{jk}^t\ D_{ij}\ \left(\delta t_j - \delta t_k\right)\right]. \tag{125}$$

Notice that in the above equation, $\delta q = [0 \; 0 \; \delta q_j - \delta q_k]^t$, hence

$$\sum_{j=1}^{M}\sum_{k=1}^{M} A_{jk}^t B_{jk} \delta q = -\sum_{j=1}^{M}\sum_{k=1}^{M} \left[I - Rot_Z\left(\Delta q_{jk}\right)\right]^t Skew\left[Rot_Z\left(\Delta q_{jk}\right)\; l_i\right] \begin{bmatrix} 0 \\ 0 \\ \delta q_j - \delta q_k \end{bmatrix}. \tag{126}$$

By using Lemma 13 and noting that $\Delta q_{jk} = -\Delta q_{kj}$, equation (126) can be simplified to be as follows:

$$\sum_{j=1}^{M}\sum_{k=1}^{M} A_{jk}^t \, B_{jk} \, \delta q = \sum_{j=1}^{M}\sum_{k=1}^{M} [Rot_Z(\Delta q_{jk}) - Rot_Z(-\Delta q_{jk})] \;\; Skew[l_i] \begin{bmatrix} 0 \\ 0 \\ \delta q_j \end{bmatrix}. \tag{127}$$

Furthermore, the last term in the right hand side of equation (125) can be simplified as follows:

$$\left[\sum_{j=1}^{M}\sum_{k=1}^{M} A_{jk}^t \, D_{ij} \;\; (\delta t_j - \delta t_k)\right] = 2 \left[\sum_{j=1}^{M}\sum_{k=1}^{M} A_{jk}^t \, D_{ij} \, \delta t_j\right]. \tag{128}$$

By substituting equations (127) and (128) into equation (125), we have

$$\left[\sum_{j=1}^{M}\sum_{k=1}^{M} A_{jk}^t \, A_{jk}\right] \;\; \delta l_i = \sum_{j=1}^{M}\sum_{k=1}^{M} \left[E_{jk} \begin{bmatrix} 0 \\ 0 \\ \delta q_j \end{bmatrix} - A_{jk}^t \, C_{jk} \, \delta \Phi_j + 2 \, A_{jk}^t \, D_{ij} \, \delta t_j \right], \tag{129}$$

where $E_{jk} = [Rot_Z(\Delta q_{jk}) - Rot_Z(-\Delta q_{jk})] \;\; Skew[l_i]$. Assume that the encoder error, δq_j, the orientation measurement error, $\delta \Phi_j$, and the position measurement error, δt_j, are independent. From equation (129), we have that the covariance matrix of δl_i will satisfy the following equation:

$$\mathcal{A} \, Cov(\delta l_i) \, \mathcal{A} = \left[\sigma_{\delta q}^2 \, \mathcal{B} + \sigma_{\delta \Phi}^2 \, \mathcal{C} + \sigma_{\delta t}^2 \, \mathcal{D}\right], \tag{130}$$

where

$$\mathcal{A} \;\; = \;\; \sum_{j=1}^{M}\sum_{k=1}^{M} A_{jk}^t \, A_{jk}, \tag{131}$$

$$\mathcal{B} \;\; = \;\; \sum_{j=1}^{M} \left[\left(\sum_{k=1}^{M} E_{jk}\right) \left(\sum_{k=1}^{M} E_{jk}^t\right)\right], \tag{132}$$

$$\mathcal{C} \;\; = \;\; \sum_{j=1}^{M} \left[\left(\sum_{k=1}^{M} A_{jk}^t \, C_{jk}\right) \left(\sum_{k=1}^{M} C_{jk}^t \, A_{jk}\right)\right] \tag{133}$$

$$\mathcal{D} \;\; = \;\; 4\sum_{j=1}^{M} \left[\left(\sum_{k=1}^{M} A_{jk}^t \, D_{ij}\right) \left(\sum_{k=1}^{M} D_{ij}^t \, A_{jk}\right)\right] \tag{134}$$

E. Covariance Matrices of the Calibration Error of a Revolute Joint Using Two-Configuration Approach

In this subsection, we will derive the covariance matrices of the estimation error of a revolute joint when using pose method. Suppose that there are totally M measurements. When using the two-configuration approach for calibrating a revolute joint, half of the measurements will correspond to a joint value, q_1, and the others will correspond to another joint value, $(q_1 + \Delta Q)$. The covariance matrix of the orientation estimation error can be computed from equation (112) and Lemma 14 as follows:

$$\sigma_{\delta u}^2 = \frac{2\,\sigma^2}{M\;(1-\cos(\Delta Q))} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{135}$$

To derive the covariance matrix of the translation estimation error, we first compute those matrices in equation (130) by using Lemma 14, and the results are

$$\mathcal{A} = \left(\frac{M}{2}\right)^2 \begin{bmatrix} 8\sin^2(\frac{\Delta Q}{2}) & 0 & 0 \\ 0 & 8\sin^2(\frac{\Delta Q}{2}) & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{136}$$

$$\mathcal{B} = \left(\frac{M}{2}\right)^3 \begin{bmatrix} 8\,l_{ix}^2\sin^2(\Delta Q) & 8\,l_{ix}\,l_{iy}\,\sin^2(\Delta Q) & 0 \\ 8\,l_{ix}\,l_{iy}\,\sin^2(\Delta Q) & 8\,l_{ix}^2\sin^2(\Delta Q) & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{137}$$

$$\mathcal{C} = \left(\frac{M}{2}\right)^3 \begin{bmatrix} 32\,l_{iy}^2\,\sin^4(\frac{\Delta Q}{2}) & -32\,l_{ix}\,l_{iy}\,\sin^4(\Delta Q) & 0 \\ -32\,l_{ix}\,l_{iy}\,\sin^4(\Delta Q) & 32\,l_{iy}^2\,\sin^4(\frac{\Delta Q}{2}) & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{138}$$

$$\mathcal{D} = 4\left(\frac{M}{2}\right)^3 \begin{bmatrix} 8\sin^2(\frac{\Delta Q}{2}) & 0 & 0 \\ 0 & 8\sin^2(\frac{\Delta Q}{2}) & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{139}$$

From the above equations, the covariance matrix of the translation estimation error can be computed as follows:

$$Cov[\delta l_i] = \frac{\sigma_{\delta q}^2}{M\ \tan^2(\frac{\Delta Q}{2})} \begin{bmatrix} l_{ix}^2 & l_{ix}\, l_{iy} & 0 \\ l_{ix}\, l_{iy} & l_{iy}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$+\frac{\sigma_{\delta\Phi}^2}{M} \begin{bmatrix} l_{iy}^2 & -l_{ix}\, l_{iy} & 0 \\ -l_{ix}\, l_{iy} & l_{ix}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$+\frac{\sigma_{\delta t}^2}{M\ \sin^2(\frac{\Delta Q}{2})} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{140}$$

F. Covariance Matrices of the Calibration Error of a Revolute Joint Using Uniform-Configuration Approach

In this subsection, we will derive the covariance matrices of the estimation error of a revolute joint when using pose method. Suppose that there are totally M measurements. When using the uniform-configuration approach for calibrating a revolute joint, the joint values corresponding to the M measurements are uniformly distributed within a region, e.g., $[q_1, q_1 + \Delta Q]$. Based on Lemma 2, the covariance matrix of the orientation estimation error can be computed from equation (112), by using Mathematica, as follows:

$$\sigma_{\delta u}^2 = \frac{\sigma^2\ \Phi_z(\Delta Q)}{M} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{141}$$

Also, based on Lemma 2, the covariance matrix of the translation estimation error can be computed from equation (130) as follows:

$$Cov[\delta l_i] = \frac{\sigma_{\delta q}^2\ \Psi_5(\Delta Q)}{M} \begin{bmatrix} l_{ix}^2 & l_{ix}\, l_{iy} & 0 \\ l_{ix}\, l_{iy} & l_{iy}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$+\frac{\sigma_{\delta\Phi}^2\ \Psi_1(\Delta Q)}{M} \begin{bmatrix} l_{iy}^2 & -l_{ix}\, l_{iy} & 0 \\ -l_{ix}\, l_{iy} & l_{ix}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$+\frac{\sigma_{\delta t}^2\,\Phi_z(\Delta Q)}{M}\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\end{bmatrix}. \tag{142}$$

V. Simulation Results

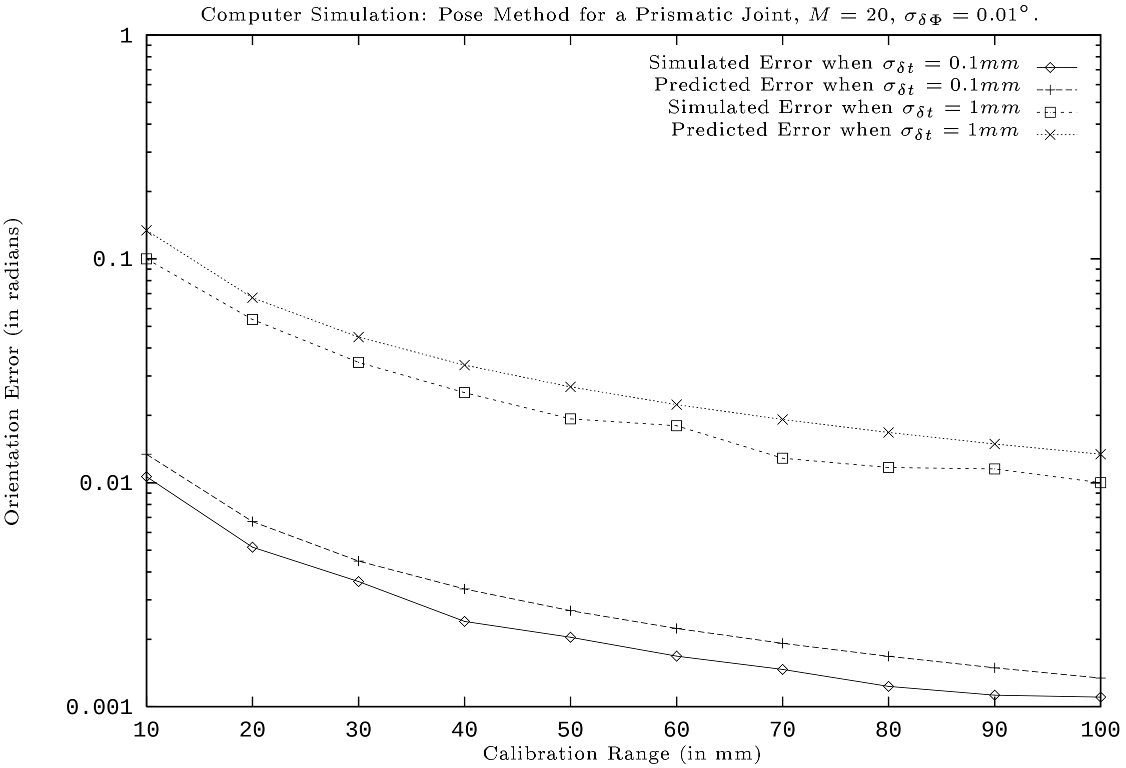

Fig. 8. The orientation parameter estimation error of a prismatic joint using the pose methods versus the value of the calibration range.

In this section, we will show some simulation results compared with the predicted estimation error derived in the previous two sections. In the computer simulations, independent Gaussian noise was added to the joint value and the position and orientation measurements of the calibration object. Then, RMSE (Root Mean Square Error) of the estimated parameters were computed from 100 random trials. For convenience, the RMSE of the parameter estimation error obtained in computer simulation will be referred to as the *simulated estimation error*, and the estimation error predicted by using the theoretical analysis results will be referred to as the *predicted estimation error*. For clarity of data

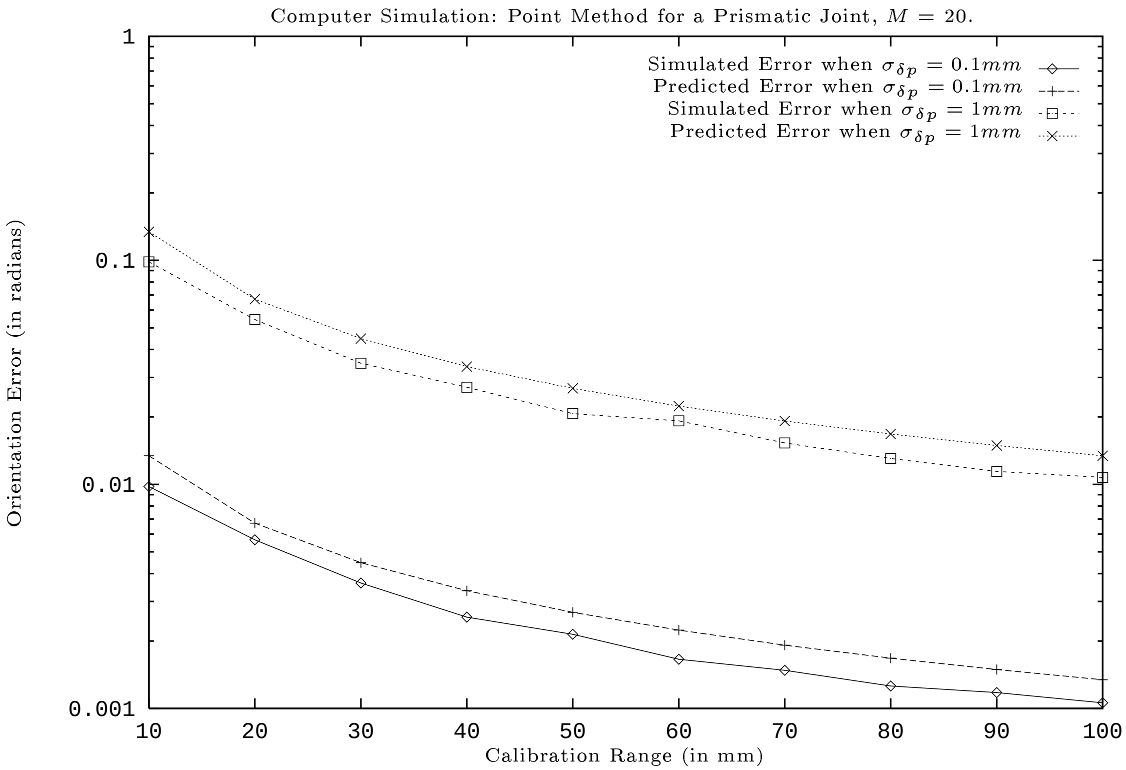

Fig. 9. The orientation parameter estimation error of a prismatic joint using the point method versus the value of the calibration range.

representing, instead of showing every x-, y- and z- components of the translation and orientation estimation errors, we will show the total 3-D RMSE translation error and the total oritntation error in the next. For example, if the x-, y- and z- components of the simulated (predicted) translation estimation error are $e_{\delta t_x}$, $e_{\delta t_y}$ and $e_{\delta t_z}$ ($\sigma_{\delta t_x}$, $\sigma_{\delta t_y}$ and $\sigma_{\delta t_z}$), then the total 3-D simulated (predicted) translation error will be $(e^2_{\delta t_x} + e^2_{\delta t_y} + e^2_{\delta t_z})^{\frac{1}{2}}$ $(\sigma^2_{\delta t_x} + \sigma^2_{\delta t_y} + \sigma^2_{\delta t_z})^{\frac{1}{2}})$. Similarly, if the x-, y- and z- components of the simulated (predicted) orientation estimation error are $e_{\delta\phi_x}$, $e_{\delta\phi_y}$ and $e_{\delta\phi_z}$ ($\sigma_{\delta\phi_x}$, $\sigma_{\delta\phi_y}$ and $\sigma_{\delta\phi_z}$), then the total 3-D simulated (predicted) orientation error will be $(e^2_{\delta\phi_x} + e^2_{\delta\phi_y} + e^2_{\delta\phi_z})^{\frac{1}{2}}$ $(\sigma^2_{\delta\phi_x} + \sigma^2_{\delta\phi_y} + \sigma^2_{\delta\phi_z})^{\frac{1}{2}})$.

In the first experiment, the derived orientation error upper bound for calibrating a prismatic joint was tested. The number of measurements was assumed to be 20 and standard deviation of the orientation measurement error was set to 0.01°. The simulated and predicted estimation errors were computed with respect to 10 different calibration ranges and two position measurement noise levels, i.e., ΔQ = 10, 20, ..., 100 mm and $\sigma_{\delta t} = \sigma_{\delta p} = 0.1$ mm and $\sigma_{\delta t} = \sigma_{\delta p} = 1$ mm, respectively. The simulated errors for the pose method and the point method were shown in Figure 8 and Figure 9, respectively. Notice that all the computed 3-D RMSE were bounded by the derived upper bounds. As expected, the orientation estimation error for a prismatic joint is inversely proportional to the calibration range, no matter which kind of method is used.

In the second experiment, the relation between the parameter estimation error of a revolute joint and the calibration range was evaluated. In this simulation, number of measurements, the joint value noise, $\sigma_{\delta q}$, and the orientation measurement noise were set to 20, 0.01° and 0.2°, respectively. The simulated and predicted estimation errors were computed with respect to 20 different calibration ranges and two position measurement noise levels, i.e., ΔQ = 10, 20, ..., 200 degrees and $\sigma_{\delta t} = \sigma_{\delta p} = 0.1$ mm and $\sigma_{\delta t} = \sigma_{\delta p} = 1$ mm, respectively. The simulation results for the pose method and the point method were shown in Figures 10–11 and Figures 12–14, respectively. Notice that the predicted estimation errors are very close to the computed ones, which means that the theoretical analysis results are very accurate.

In the last experiment, the relation between the parameter estimation errors of the point method and the length of the trajectory radius, ρ, was evaluated. In this simulation, the number of measurements, the standard deviation of joint values noise and the standard deviation of the position measurement noise were set to 20, 0.01° and 0.1 mm, respectively. The simulated and predicted estimation errors were computed with respect to 10 radius lengthes and two calibration ranges, i.e., $\rho = 50, 100, ..., 500$mm and $\Delta Q = 40, 180$ degrees. The calibration results were shown in Figures 15–17. From the computer simulation results, we concluded that the theoretical analysis results are correct and, thus, can be used as a guideline for selecting calibration methods and for controlling calibration conditions to reduce the calibration error.

VI. Conclusions

Robot kinematic calibration is important for many applications requiring high positioning accuracy; hence, many techniques have been developed to calibrate the kinematic parameters to reduce the positioning error. Most of the existing techniques are based on

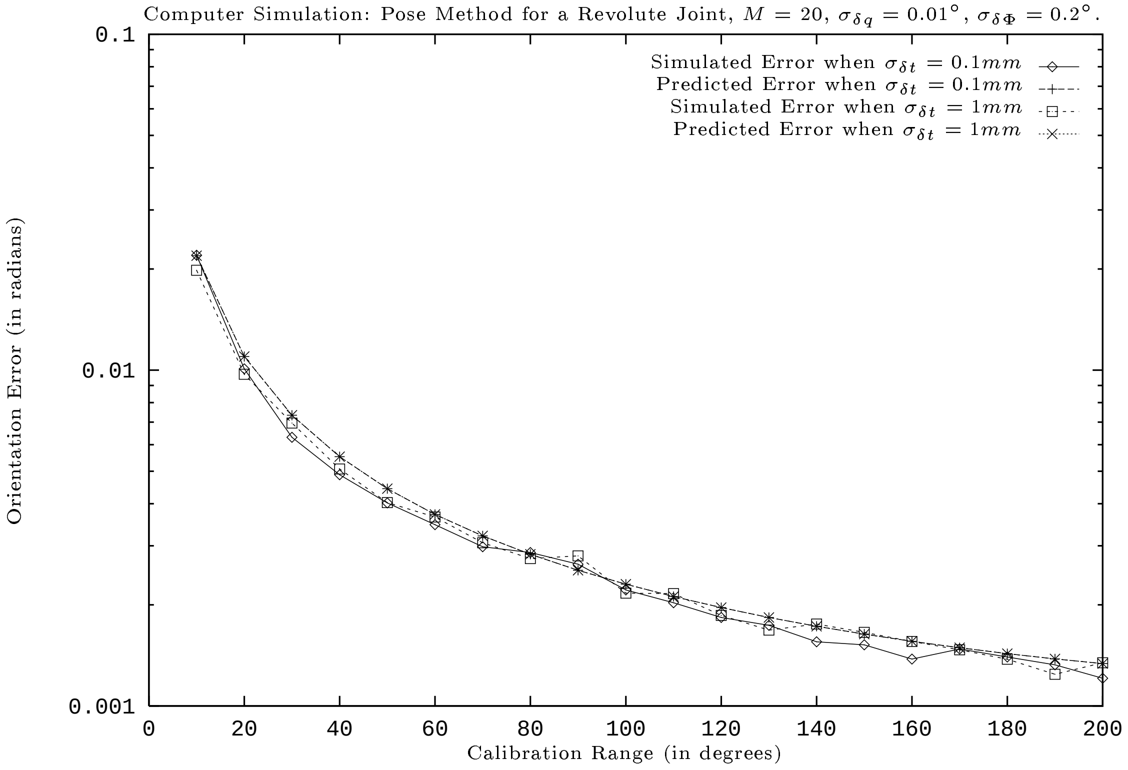

Fig. 10. The orientation parameter estimation error of a revolute using the pose methods versus the calibration range.

nonlinear optimization techniques which require accurate initial estimate of kinematic parameters. Several methods have been developed for providing closed-form solutions to the kinematic parameters which can be used as an initial estimate for nonlinear optimization techniques or for direct applications. Existing closed-form solutions to kinematic parameter calibration problems can be classified into two categories, namely the pose methods and the point methods, according to the information they used. Pose methods estimate the kinematic parameters by using pose measurements of the end-effector. Whereas, point methods estimate the kinematic parameters by using only 3-D point measurements of a calibration target attached to the end-effector.

In this paper, we have successfully derived expressions for covariance matrices of kinematic parameters estimated by using the pose method and the point method, respectively. The derived error variances for a point method are functions of the calibration range,

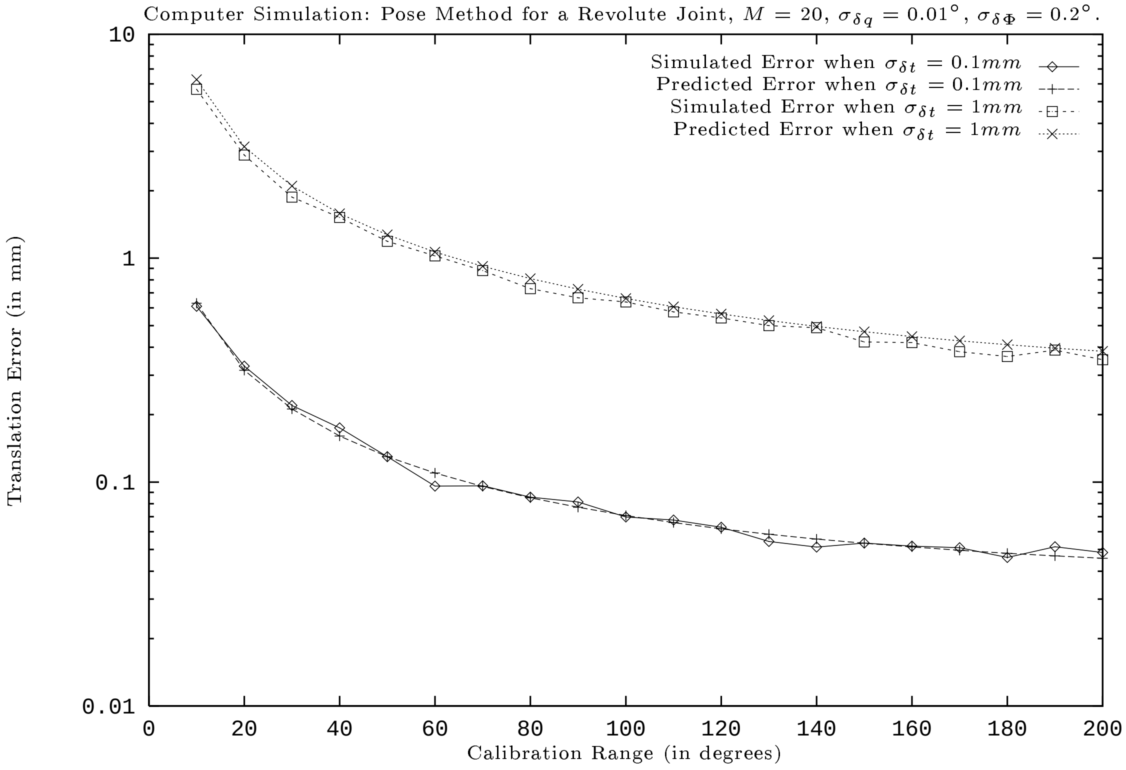

Fig. 11. The translation parameter estimation error of a revolute joint using the pose methods versus the calibration range.

number of measurements, amount of measurement noise and amount of joint value noise. Furthermore, if the joint under calibration is revolute, then the error variances are also functions of the distance between the calibration point and the revolute joint axis and length of the link corresponding to the joint under calibration. The derived error variances for a pose method are functions of the calibration range, number of measurements, amount of measurement noise, amount of the joint value noise and length of the link corresponding to the joint under calibration. To verify the theoretical analysis results, extensive computer simulations were conducted which showed that the derived variance equations for the estimated kinematic parameters are very accurate.

Based on our error analysis results, factors affecting calibration accuracy are revealed. We found that when the calibration range is small, both pose methods and point methods will become very sensitive to noise, especially when calibrating a revolute joint. Notice

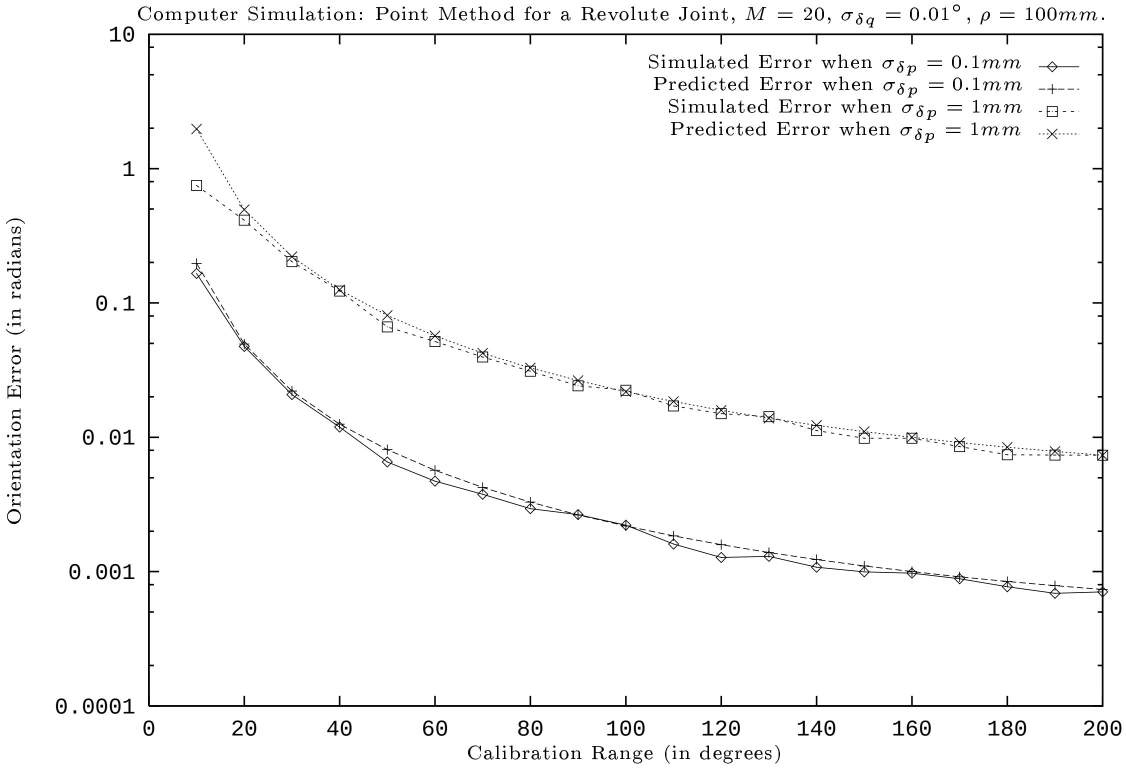

Fig. 12. The orientation parameter estimation error of a revolute joint using the point method versus the calibration range.

that although accuracy of the point method will degrade more rapidly than a pose method when the calibration range is becoming smaller, accuracy of point method can be improved by increasing either the number of measurements or the distance between the calibration target and the rotation joint axis. In general, amount of measurement noise can hardly be reduced; hence, the calibration error should be reduced by controlling other factors. According to our analysis results, increasing the calibration range is the most efficient way, for both point method and pose method, to reduce the calibration error. Nevertheless, when the calibration range can not be increased, than increasing the distance between the calibration target and the rotation axis is much more efficient than increasing the number of measurements for point method. On the other hand, when we want to reduce the estimation error for a pose method, the only way is to increase the number of calibration points.

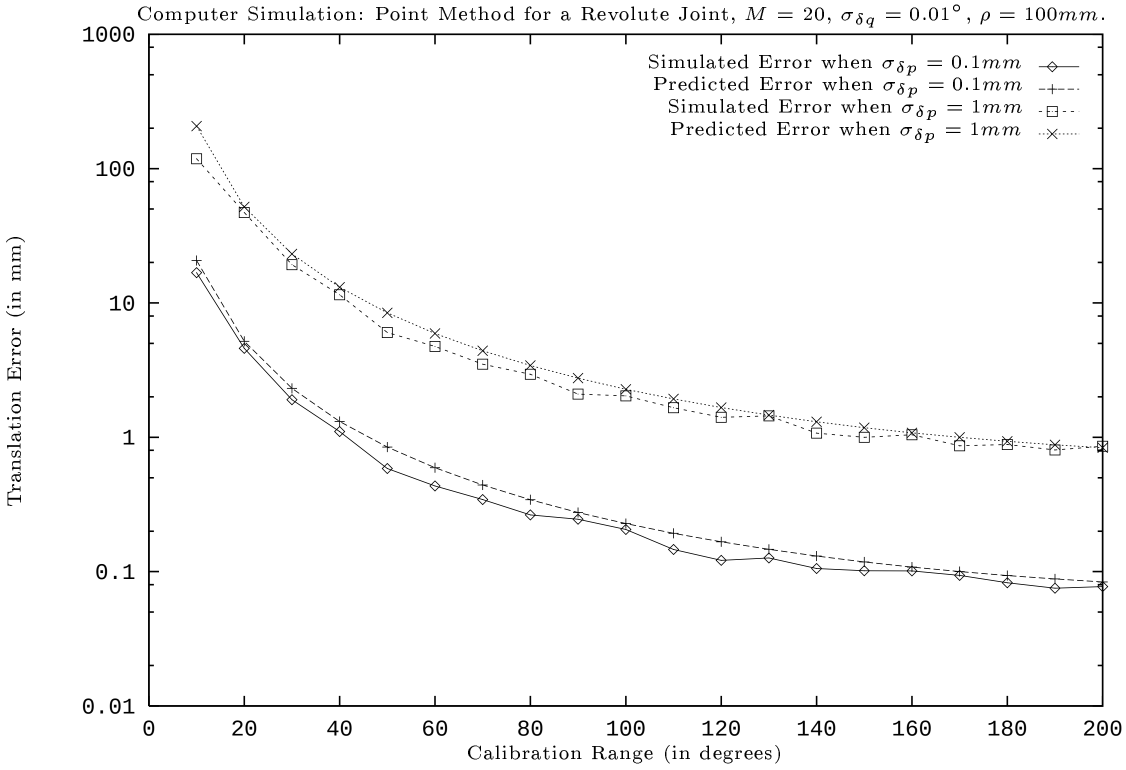

Fig. 13. The translation parameter estimation error of a revolute joint using the point method versus the calibration range.

From the error analysis results, we found that both the point method and the pose method have their pros and cons. Our analysis results can be used to serve as a guideline for selecting calibration techniques, for determining the calibration condition and even for designing a robot head or a robot arm when considering the calibration task. For example, when designing a robot head, the required rotating range of a joint may be small; however, when considering the subsequent calibration task, the rotating range should be enlarged to ensure accurate calibration results.

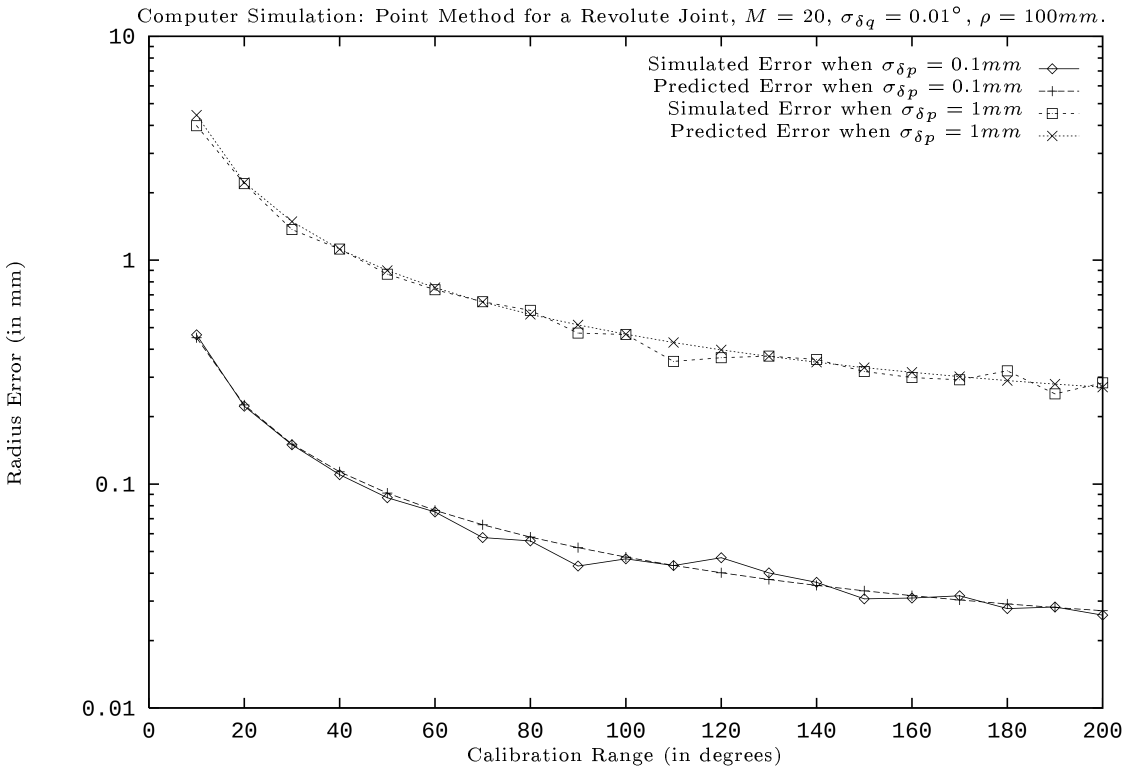

Fig. 14. The radius parameter estimation error of a revolute joint using the point method versus the calibration range.

VII. Appendices

A. *Orientation Estimation Error of the Point Method for a Prismatic Joint*

Assuming that M is an even number, and then from equation (31), we have

$$\underline{q}_{i+1}(j) = \left(j + \frac{M+1}{2}\right) \frac{\Delta Q}{M-1}, \tag{143}$$

for $i = 1, 2, ..., M$, and

$$\delta P = \sum_{j=1}^{M} \underline{q}_{i+1}(j)\ \delta p(j) - \delta \bar{p} \sum_{j=1}^{M} \underline{q}_{i+1}(j). \tag{144}$$

Notice that the second term of equation (144) on the right hand side is zero because

$$\delta \bar{p} \sum_{j=1}^{M} \underline{q}_{i+1}(j) = \bar{p} \sum_{j=1}^{M} (q_{i+1}(j) - \bar{q}_{i+1}) = 0. \tag{145}$$

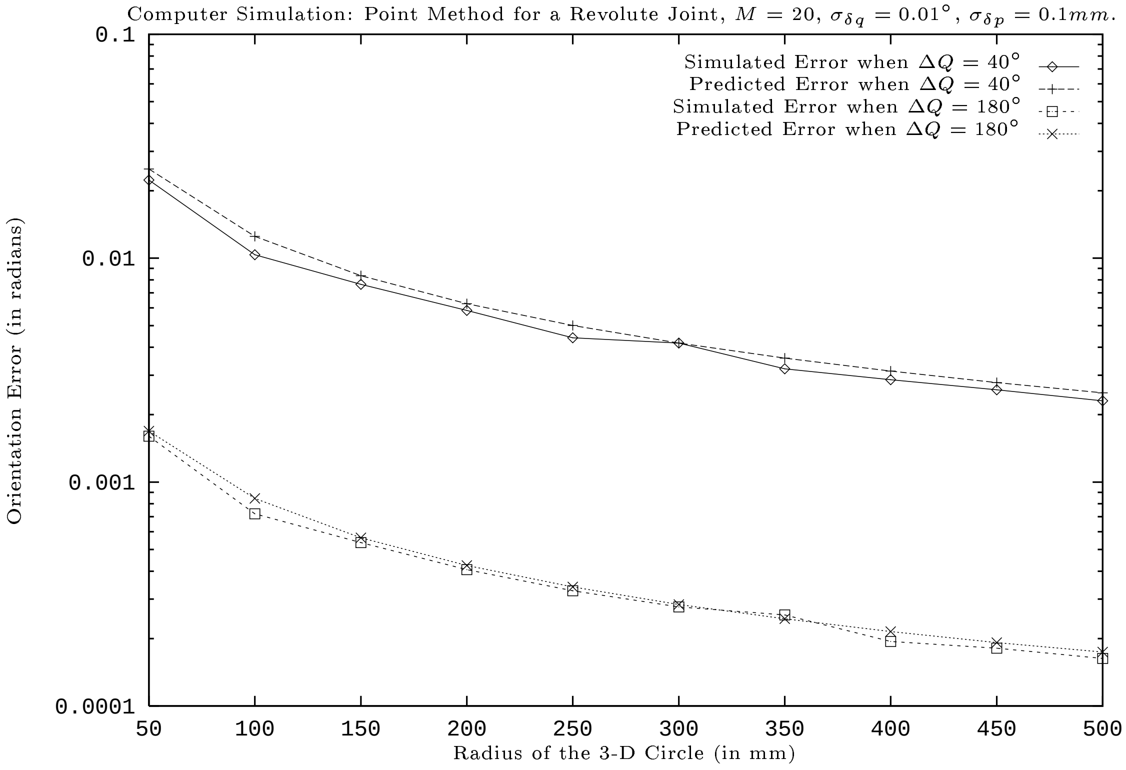

Fig. 15. The orientation parameter estimation error of a revolute joint using the point method versus the length of the radius.

From equations (144) and (145), we have

$$Cov\,[\delta P] = \left(\frac{\Delta Q}{M-1}\right)^2 \sigma^2\, I \sum_{j=1}^{M} \left(j - \frac{M+1}{2}\right)^2, \tag{146}$$

which yields

$$Cov\,[\delta P] = \sigma_{\delta P}^2\, I, \tag{147}$$

where $\sigma_{\delta P}^2 = \sigma^2\, \frac{\Delta Q^2\, M(M+1)}{12(M-1)}$. Notice that ${}^i\underline{p}(j)$ is the true vector whose length is exactly $\underline{q}_{i+1}(j)$, and all the vectors, ${}^i\underline{p}(j),\ j = 1, 2, ..., M$, have exactly the same direction as that of the true kinematic parameter, u_i, i.e., ${}^i\underline{p}(j) = \underline{q}_{i+1}(j)\, u_i$. Hence, from equations (39) and (143), we have

$$\|P\| = \sum_{j=1}^{M} \underline{q}_{i+1}(j) = \frac{\Delta Q^2\, M(M+1)}{12(M-1)}. \tag{148}$$

The covariance matrix of $\delta\beta$ can be derived from equations (147) and (148) as follows:

$$Cov\,[\delta\beta] = \sigma_{\delta\beta}^2\, I, \tag{149}$$

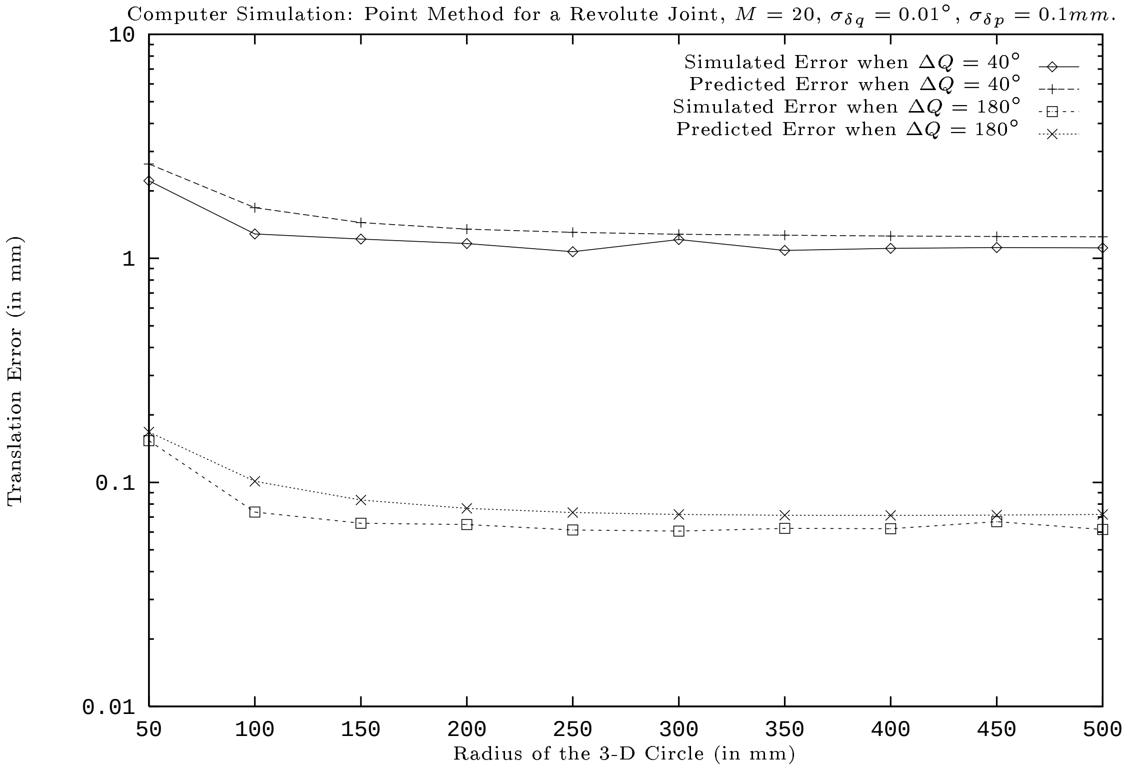

Fig. 16. The translation parameter estimation error of a revolute joint using the point method versus the length of the radius.

where

$$\sigma_{\delta\beta}^2 = \frac{\sigma_{\delta P}^2}{\|P\|^2} = \frac{12(M-1)\sigma^2}{M(M+1)\Delta Q^2}. \tag{150}$$

When the number of measurements, M, is large, the above equation can be approximated as follows:

$$\sigma_{\delta\beta}^2 = \frac{12\sigma^2}{M\Delta Q^2}. \tag{151}$$

Notice that the above equation is derived for even number of measurements. However, when the number of measurements is odd, the derived covariance matrix in the above equation can be used as a reasonable approximation.

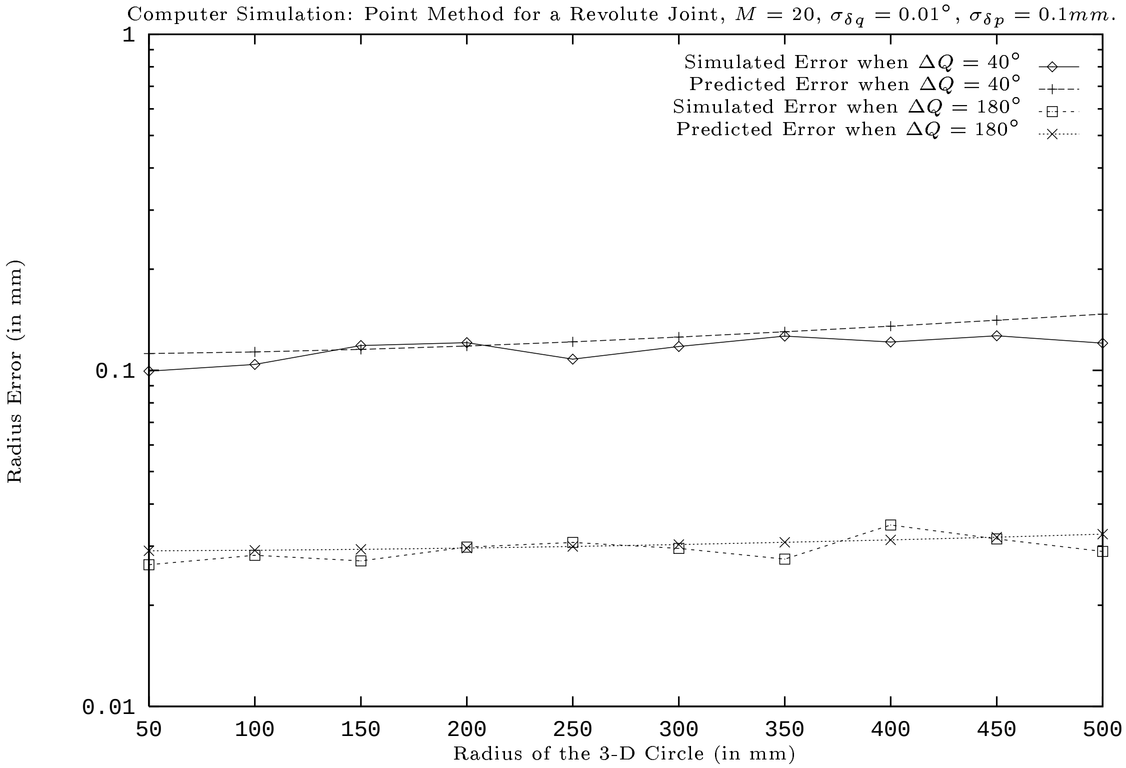

Fig. 17. The radius parameter estimation error of a revolute joint using the point method versus the length of the radius.

B. Covariance Values of the Estimated Parameters of a Revolute Joint when Using the Point Method

$$\sigma_{\delta\phi_x\,\delta t_y} = -\frac{\sigma_p^2\,\Phi_x(\Delta Q)\,t_z}{\rho^2\,M}. \tag{152}$$

$$\sigma_{\delta\phi_x\,\delta t_z} = \frac{\sigma_p^2\,\Phi_x(\Delta Q)\,t_y}{\rho^2\,M}. \tag{153}$$

$$\sigma_{\delta\phi_y\,\delta t_x} = -\frac{\sigma_p^2\,\Phi_y(\Delta Q)\,t_z}{\rho^2\,M}. \tag{154}$$

$$\sigma_{\delta\phi_y\,\delta t_z} = \frac{\sigma_p^2\,\Phi_y(\Delta Q)\,\left(t_x - \rho\,\mathrm{sinc}\left(\frac{\Delta Q}{2}\right)\right)}{\rho^2\,M}. \tag{155}$$

$$\sigma_{\delta\phi_z\,\delta t_x} = -\frac{\sigma_p^2\,\Phi_z(\Delta Q)\,t_y}{\rho^2\,M} - \frac{\sigma_{\delta q}^2\,t_y\,Psi_1}{M}. \tag{156}$$

$$\sigma_{\delta\phi_z\,\delta t_y} = \frac{\sigma_p^2\,\Phi_x(\Delta Q)\,\left(t_x - \rho\,\mathrm{sinc}\left(\frac{\Delta Q}{2}\right)\right)}{\rho^2\,M} + \frac{\sigma_{\delta q}^2\,t_x\,Psi_1}{M} - \frac{\sigma_{\delta q}^2\,\rho\,Psi_4}{2\,M}. \tag{157}$$

$$\sigma_{\delta t_x\,\delta t_y} = \frac{\sigma_p^2\,\Phi_z(\Delta Q)\,t_y\left(t_x - \rho\,\mathrm{sinc}\left(\frac{\Delta Q}{2}\right)\right)}{\rho^2\,M} - \frac{\sigma_{\delta q}^2\,t_x\,t_y\,Psi_1}{M} - \frac{\sigma_{\delta q}^2\,\rho\,t_y\,Psi_1}{2\,M}. \tag{158}$$

$$\sigma_{\delta t_x\,\delta t_z} = -\frac{\sigma_p^2\,\Phi_y(\Delta Q)\,t_z\left(t_x - \rho\,\mathrm{sinc}\left(\frac{\Delta Q}{2}\right)\right)}{\rho^2\,M}. \tag{159}$$

$$\sigma_{\delta t_x\,\delta\rho} = \frac{\sigma_p^2\,\Phi_z(\Delta Q)\,\mathrm{sinc}\left(\frac{\Delta Q}{2}\right)}{\rho^2\,M} + \frac{\sigma_{\delta q}^2\,\rho^2\,\Delta Q^2\,\sin\left(\frac{\Delta Q}{2}\right)(\Delta Q - \sin(\Delta Q))}{M(-2 + \Delta Q^2 + 2\,\cos(\Delta Q))^2} \tag{160}$$

$$\sigma_{\delta t_y\,\delta t_z} = \frac{\sigma_p^2\,\Phi_x(\Delta Q)\,t_y\,t_z}{\rho^2\,M} \tag{161}$$

C. The Right Orientation Estimation Error of $\hat{R}_i$

By definition, the rotation axis of joint i with respect to joint frame $\{i+1\}$ is

$$b_i = R_i\,e_Z, \tag{162}$$

where $e_Z = [0\ 0\ 1]^t$ and

$$R_i = \begin{bmatrix} 1 - \frac{b_x^2}{1+b_z} & \frac{-b_x\,b_y}{1+b_z} & b_x \\ \frac{-b_x\,b_y}{1+b_z} & 1 - \frac{b_y^2}{1+b_z} & b_y \\ -b_x & -b_y & b_z \end{bmatrix}. \tag{163}$$

Notice that the rotation matrix, R_i, can be reconstructed by using the unit vector, b_i. Suppose the estimated rotation axis of the ith joint with respect to the $(i+1)$st joint frame is

$$\hat{b}_i = R_i \begin{bmatrix} \delta u_x \\ \delta u_y \\ 1 \end{bmatrix}. \tag{164}$$

The estimated rotation matrix, $\hat{R}_i$, can be constructed by using the estimated rotation axis, $\hat{b}_i$, and the right orientation error, $\delta\Phi_{R_i}$, of $\hat{R}_i$ will, by definition, satisfy the following equation:

$$\hat{R}_i = R_i\,\left[I + Skew\,(\delta\Phi_{R_i})\right]. \tag{165}$$

Based on the above equation, the right orientation error, $\delta\Phi_{R_i}$, of $\hat{R}_i$ can be computed by using Mathematica to be as follows:

$$\delta\Phi_{R_i} = \begin{bmatrix} \delta u_x \\ \delta u_y \\ \frac{\delta u_y\, x - \delta u_x\, y}{1+z} \end{bmatrix}. \tag{166}$$

D. Proof of equation (128)

$$\begin{aligned}\left[\sum_{j=1}^{M}\sum_{k=1}^{M} A_{jk}^t\, D_{ij}\, (\delta t_j - \delta t_k)\right] = & - \sum_{j=1}^{M}\sum_{k=1}^{M} [I - Rot_Z\,(\Delta q_{jk})]^t\; R_i^t\, R_j^t\, \delta t_j \\ & + \sum_{j=1}^{M}\sum_{k=1}^{M} [I - Rot_Z\,(\Delta q_{jk})]^t\; R_i^t\, R_j^t\, \delta t_k. \end{aligned} \tag{167}$$

By substituting $(R_k\, R_k^t\, \delta t_k)$ for δt_k in the last term on the right hand side of the above equation, we have

$$[I - Rot_Z\,(\Delta q_{jk})]^t\; R_i^t\, R_j^t\, \delta t_k = [I - Rot_Z\,(\Delta q_{jk})]^t\; R_i^t\, R_j^t\, R_k\, R_k^t\, \delta t_k. \tag{168}$$

From equation (10), we have

$$R_j^t\, R_k = R_i\, Rot_Z(\Delta q_{jk})\, R_i^t. \tag{169}$$

By substituting equation (169) into (168), we have

$$[I - Rot_Z\,(-\Delta q_{jk})]\;\, Rot_Z(\Delta q_{jk})\, R_i^t\, R_k^t\, \delta t_k = -\,[I - Rot_Z\,(\Delta q_{jk})]\;\, R_i^t\, R_k^t\, \delta t_k. \tag{170}$$

By swapping the indices, j and k, we have (remember that $\Delta q_{jk} = -\Delta q_{kj}$)

$$-\,[I - Rot_Z\,(-\Delta q_{jk})]\;\, R_i^t\, R_j^t\, \delta t_j. \tag{171}$$

Therefore, these two terms on the right hand side of equation (167) are equal.

References

[1] D. J. Bennett and J. M. Hollerbach. Identifying the kinematics of robots and their tasks. In *IEEE International Conference on Robotics and Automation*, pages 580–586, 1989.

[2] J. L. Caenen and J. C. Angue. Identification of geometric and non geometric parameters of robots. In *IEEE International Conference on Robotics and Automation*, pages 1032–1037, 1990.

[3] S. Chen. Robot calibration using stereo vision. Master's thesis, Florida Atlantic University, Boca Raton, Florida, 1987.

[4] J. Denavit and R. S. Hartenberg. A kinematic notation for lower-pair mechanisms based on matrices. *Transations of ASME – Journal of Applied Mechanices*, 22(2):215–221, 1955.

[5] A. Goswami, A. Quaid, and M. Peshkin. Complete parameter identification of a robot from partial pose information. In *International Conference of Robotics and Automatins*, pages 168–173. IEEE, 1993.

[6] S. Hayati, K. Tso, and G. Roston. Robot geometry calibration. In *IEEE International Conference on Robotics and Automation*, pages 947–951, 1988.

[7] R. P. Judd and A. B. Knasinski. A technique to calibrate industrial robots with experimental verification. In *International Conference of Robotics and Automatins*, pages 351–357. IEEE, 1987.

[8] V. Karimäki. Effective circle fitting for particle trajectories. *Nuclear Instruments and Methods in Physics Research*, A305:187–191, 1991.

[9] D. H. Kim, K. H. Cook, and J. H. Oh. Identification and compensation of robot kinematic parameter for positioning accuracy improvement. *Robotica*, 9:99–105, 1991.

[10] R. K. Lenz and R. Y. Tsai. Calibrating a cartesian robot with eye-on-hand configuration independent of eye-to-hand relationship. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(9):916–928, 1989.

[11] B. W. Mooring and T. J. Pack. Calibration procedure for an industrial robot. In *IEEE International Conference on Robotics and Automation*, pages 786–791, 1988.

[12] L. Moura. A direct method for least-squares circle fitting. *Computer Physics Communications*, 64:57–63, 1991.

[13] S. W. Shih, Y. P. Hung, and W. S. Lin. Comments on 'a linear solution to kinematic parameter identification of robot manipulator' and some modifications. *IEEE Transactions on Robotics and Automation*, 11(5):777–780, 1995.

[14] S. W. Shih, Y. P. Hung, and W. S. Lin. Kinematic parameter identification of a binocular head using stereo measurements of single calibration point. In *IEEE International Conference on Robotics and Automation*, pages 1796–1801, Nagoya, Japan, 1995.

[15] S. W. Shih, J. S. Jin, K. H. Wei, and Y. P. Hung. Kinematic calibration of a binocular head using stereo vision with the complete and parametrically continuous model. In *SPIE Proceedings, Intelligent Robots and Computer Vision XI*, volume 1825, pages 643–657, 1992.

[16] M. E. Sklar. *Metrology and Calibration Techniques for the Performace Enhancement of Industrial Robots.* PhD thesis, University of Texas at Austin, 1988.

[17] H. W. Stone. *Kinematic Modeling, Identification, and Control of Robotic Manipulators.* Norwell, MA: Kluwer Academic, 1987.

[18] S. Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):376–380, 1991.

[19] W. K. Veitschegger and C. Wu. A method for calibrating and compensating robot kinematic errors. In *IEEE International Conference on Robotics and Automation*, pages 39–44, 1987.

[20] G. S. Young, T. H. Hong, M. Herman, and J. C. S. Yang. Kinematic calibration of an active camera system. In *IEEE Proceeding of the International Conference on Computer Vision Pattern Recognition*, pages 748–751, 1992.

[21] H. Zhuang and Z. S. Roth. A linear solution to the kinematic parameter identification of robot manipulator. *IEEE Transactions on Robotics and Automation*, 9(2):174–185, 1993.

[22] H. Zhuang, Z. S. Roth, and F. Hamano. A complete and parametrically continuous kinematic model for robot manipulators. *IEEE Transactions on Robotics and Automation*, 8(4):451–463, 1992.

[23] H. Zhuang, K. Wang, and Z. S. Roth. Simultaneous calibration of a robot and a hand-mounted camera. *IEEE Transactions on Robotics and Automation*, 11(5):649–660, 1995.