Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems Raleigh, NC July 7-10, 1992

A Noise Tolerant Algorithm for Wrist-Mounted Robotic Sensor Calibration with or without Sensor Orientation Measurement

Hanqi Zhuang
Robotics Center and Department of Electrical Engineering, Florida Atlantic University, Boca Raton, FL 33431

Yiu Cheung Shiu
Department of Electrical Engineering, Wright State University, Dayton, OH 45435

Abstract

A noise tolerant algorithm for calibration of wrist-mounted robotic sensors is presented. The sensor-wrist calibration can be performed by solving a system of homogeneous transformation equations of the form $A_iX = XB_i$, where X is the unknown sensor position relative to the robot wrist, A_i is the ith robot motion, and B_i is the ith sensor motion [1-4]. A Jacobian relating the measurement residual errors to pose errors of the unknown matrix X is derived. Based on this relationship, X can be iteratively solved. Unlike existing approaches, the algorithm presented here solves kinematic parameters of X in one stage, thus eliminating error propagations and improving noise sensitivities. Moreover, with the proposed algorithm, the parameters of X are observable even when the rotation part of B_i is unknown. This is important in practice since position is easier to measure than orientation. Comparative simulation studies show that the accuracy performance of the iterative algorithm is, in general, better than that of noniterative two-stage algorithms, regardless whether the orientation part of B_i is used. The approach presented in this paper also has wide applications for wrist-mounted tool calibration.

1. Introduction

Wrist-mounted robotic sensor calibration is the process of computing the relative 3D position and orientation of the robot sensor with respect to the robot wrist [1-4]. This is an important task for robot applications involving 3D vision measurement, automatic sensor placement planning, visual servoing, and tactile sensing.

The robot wrist frame, also called the end-effector frame, refers to the 3D coordinate frame that the robot controller software uses. The sensor frame refers to the coordinate system used by the sensor. The position of the sensor frame can either be measured by the sensor itself or by an external device.

Since robot motion commands are specified relative to the robot wrist frame and the measured object positions are measured relative to the sensor frame, it is necessary to find the relative position between the coordinate frames of the robot wrist and a sensor mounted on it. This measurement is known as wrist-mounted sensor calibration. This relative position cannot be measured since both the robot wrist frame and the sensor frame are usually located inside the robot and the sensor.

A number of approaches have been proposed in the literature. Shiu and Ahmad [1] proposed moving the robot twice and observing the resulting sensor motion in which the robot is moved by a known mount and the resulting motion is measured. This is equivalent to solving the homogeneous transformation equations of the form,

$$A_iX = XB_i \qquad i = 1, 2 \qquad (1.1)$$

where X (4x4) is the transformation from the robot tool coordinate frame to the sensor coordinate frame, A_i (4x4) is the transformation of the robot gripper from its ith to the (i+1)th position, and B_i (4x4) is the transformation of the sensor, also from its ith to (i+1)th position. Moreover,the index i indicates the ith motion. Shiu and Ahmad proposed making two or more robot motions to obtain a unique solution under certain conditions.

Tsai and Lenz [2] solved the problem indepen-dently using a more efficient linear algorithm. Simulation and experiments were reported and analyzed to show the efficiency and accuracy of the algorithm.

It was shown in [1] that Equation (1.1) can be broken into the following equations

$$R_{A,i}R_X = R_XR_{B,i} \qquad i = 1, 2 \qquad (1.2a)$$

$$R_{A,i}\, p_X + p_{A,i} = R_X p_{B,i} + p_X \qquad i = 1, 2 \qquad (1.2b)$$

where R_X (3x3) and p_X (3x1) are the rotation and position parts of X, respectively. Similar definitions are used for $R_{A,i}$ (3x3), $R_{B,i}$ (3x3), $p_{A,i}$ (3x1) and $p_{B,i}$ (3x1). If R_X is known, p_X may then be solved linearly from the system of position equations (1.2b). Therefore more attention has been focused on solving R_X from the system of rotation equations (1.2a).

Chou and Kamel [3] presented an algorithm based on quaternions for solving Equation (1.2a). In this approach, a system of nonlinear equations is iteratively solved using the Newton-Raphson procedure. Zhuang and Roth [4] solved the rotation equations using a noniterative approach based also on quaternions.

The above algorithms have two common characteristics. (1) They all consist of two stages. R_X is solved from (1.2a) first, and p_X is then obtained from (1.2b) using the known R_X. A drawback of such an approach is that rotation estimation errors propagate to position errors. The two-stage algorithms may be optimal in each stage. However, these may not be globally optimal due to the coupling of the rotation and position equations given in (1.2). (2) They all require the complete knowledge of A_i and B_i.

In this paper (also in [12]), a one-stage iterative algorithm for wrist-mounted sensor calibration is presented. A Jacobian

0-7803-0737-2/92$03.00 1992©IEEE

relating the measurement residual errors to pose errors of the unknown matrix X is derived. Based on this relationship, X can be iteratively solved. Not only is this algorithm less sensitive to noise compared to present algorithms, but also it handles the cases in which the orientation information of the robot sensor is not available. Unlike other calibration approaches, the one-stage algorithm obtain all kinematic parameters of X simultaneously, thus eliminating possible propagation errors and improving noise sensitivities. With this algorithm, the parameters of X are observable even without sensor orientation measurement. Comparative simulation studies show that the accuracy performance of the iterative algorithm is, in general, better than that of noniterative two-stage algorithms.

2. A One-Stage Algorithm

In this section, the cost function for the problem is formulated, then the linearized relationship between the measurement residuals and the parameter errors of X is derived, in the form of a Jacobian matrix. These results are utilized in a one-stage iterative algorithm that finds an X which minimizes the cost function.

2.1 Cost Function

Let H be an arbitrary 3x3 matrix. For convenience, the following notations will be used,

$$H \equiv \begin{bmatrix} Row(H)_1^T \\ Row(H)_2^T \\ Row(H)_3^T \end{bmatrix} \equiv [\ Col(H)_1 \quad Col(H)_2 \quad Col(H)_3\ ] \quad (2.1)$$

where $Row(H)_i^T$ and $Col(H)_i$, i = 1, 2, 3, are the ith row and column of H, respectively. Both $Row(H)_i$ and $Col(H)_i$ are treated as 3x1 column vectors.

Let us now derive the cost function for the problem. Let

$$Z_i \equiv A_i X - X B_i \quad i = 1, 2, ..., m. \quad (2.2a)$$

Z_i is a 3x3 matrix, and m is the number of measurements. Denote by $M_{Z,i}$ and $p_{Z,i}$ respectively the upper left 3x3 matrix and the upper right 3x1 vector of Z_i. A vectorized form of Z_i is defined by

$$z_i \equiv \begin{bmatrix} p_{Z,i} \\ Col(M_{Z,i})_1 \\ Col(M_{Z,i})_2 \\ Col(M_{Z,i})_3 \end{bmatrix} \quad i = 1, 2, ..., m. \quad (2.2b)$$

z_i (12x1) contains all nontrivial elements of Z_i. Z_i and z_i are termed the *measurement residual matrix* and *vector of* X, respectively. The problem at hand is to choose an X that minimizes the cost function E, where

$$E = \sum_{i=1}^{m} z_i(X)^T z_i(X). \quad (2.3)$$

The constraints that the rotation part of X must be an orthonormal matrix and its determinant must be equal to one are automatically satisfied by expressing the rotation part of X as a function of its rotation parameters. See section 2.4 for more detail.

It is assumed that X^0, the nominal value of X, is provided by some means such as engineering drawings, or by a noniterative procedure. It is also assumed that X^0 is deviated by a small amount from X^*, the optimal solution. This can be achieved since results obtained by a noniterative solution procedure is quite accurate. By the small error assumption, the following equation holds

$$X^* = X^0 + dX \quad (2.4)$$

where dX is a 4x4 differential transformation [9]. Replacing X in Equation (1.1) by X^* in (2.4) yields

$$A_i(X^0 + dX) = (X^0 + dX)B_i \quad i = 1, 2, ..., m. \quad (2.5)$$

Let

$$\delta X \equiv (X^0)^{-1} dX. \quad (2.6)$$

δX, a 4x4 matrix, has the following structure [9],

$$\delta X = \begin{bmatrix} \Omega(\delta) & d \\ 0_{1\times 3} & 0 \end{bmatrix}$$

where δ and d are respectively the 3x1 rotation and position error vectors of X, and

$$\Omega(\delta) \equiv \begin{bmatrix} 0 & -\delta_z & \delta_y \\ \delta_z & 0 & -\delta_x \\ -\delta_y & \delta_x & 0 \end{bmatrix}.$$

For convenience, let $\eta \equiv [d^T, \delta^T]^T$; η, a 6x1 vector, is termed the *pose error vector of* X. Equation (2.5) can then be written as

$$Z_i^0 = -A_i X^0 \delta X + X^0\ \delta X B_i \quad i = 1, 2, ..., m \quad (2.7)$$

where $Z_i^0 \equiv A_i X^0 - X^0 B_i$. The cost function in (2.3) is a nonlinear function of X, therefore, an iterative procedure is needed to obtain its solution. Let E^k be in the form of Equation (2.3) with z_i being replaced by z_i^k, the vector representation of Z_i^k after the kth iteration. The optimization problem is now reduced to iteratively minimizing the cost function E^k by choosing $dX(\eta^k)$, an adjustment to X^k [5,6]. More detail is given in Section 2.4.

2.2 Identification Jacobian

The Identification Jacobian is a matrix relating measurement residuals to rotation and position errors of X. A simple fact is given before the derivation of the Identification Jacobian.

Lemma 2.1: Let R be a 3x3 rotation matrix, $\Omega(\delta)$ be a 3x3 skew-symetric matrix corresponding to δ, and v be a 3x1 vector. Then

$$R\,\Omega(\delta) v = -\begin{bmatrix} (Row(R)_1 \times v)^T \\ (Row(R)_2 \times v)^T \\ (Row(R)_3 \times v)^T \end{bmatrix} \delta \quad (2.8)$$

Let us now derive the linear transformation relating the residual error vector z_i to the pose error vector η. Breaking (2.7) into the rotation part and position part yields

$$M_{Z,i} = -R_{A,i}R_X\,\Omega(\delta) + R_X\,\Omega(\delta)R_{B,i} \qquad (2.9a)$$

$$p_{Z,i} = -R_{A,i}R_X d + R_X d + R_X\,\Omega(\delta)p_{B,i} \quad i = 1, 2, ..., m. \qquad (2.9b)$$

Equation (2.9a) can be written as

$$M_{Z,i} = -R_{A,i}R_X\,\Omega(\delta)I + R_X\,\Omega(\delta)R_{B,i} \quad i = 1, 2, ..., m. \qquad (2.10)$$

which can be written in a vector equation form,

$$Col(M_{Z,i})_j = -R_{A,i}R_X\,\Omega(\delta)Col(I)_j + R_X\,\Omega(\delta)Col(R_{B,i})_j$$
$$i = 1, 2, ..., m; \quad j = 1, 2, 3. \qquad (2.11)$$

By using the notation of (2.1) and Lemma 2.1, one has

$$Col(M_{Z,i})_j = \begin{bmatrix} \left(Row(R_{A,i}R_X)_1 \times Col(I)_j - Row(R_{X,i})_1 \times Col(R_{B,i})_j\right)^T \\ \left(Row(R_{A,i}R_X)_2 \times Col(I)_j - Row(R_{X,i})_2 \times Col(R_{B,i})_j\right)^T \\ \left(Row(R_{A,i}R_X)_3 \times Col(I)_j - Row(R_{X,i})_3 \times Col(R_{B,i})_j\right)^T \end{bmatrix} \delta$$
$$i = 1, 2, ..., m; \quad j = 1, 2, 3. \qquad (2.12a)$$

Similarly, (2.9b) can be put into the following form,

$$p_{Z,i} = -(R_{A,i} - I)R_X d - \begin{bmatrix} \left(Row(R_{X,i})_1 \times p_{B,i}\right)^T \\ \left(Row(R_{X,i})_2 \times p_{B,i}\right)^T \\ \left(Row(R_{X,i})_3 \times p_{B,i}\right)^T \end{bmatrix} \delta$$
$$i = 1, 2, ..., m. \qquad (2.12b)$$

Let

$$J_i = \begin{bmatrix} -(R_{A,i} - I)R_X & -\begin{bmatrix} \left(Row(R_{X,i})_1 \times p_{B,i}\right)^T \\ \left(Row(R_{X,i})_2 \times p_{B,i}\right)^T \\ \left(Row(R_{X,i})_3 \times p_{B,i}\right)^T \end{bmatrix} \\ 0_{3\times 3} & \begin{matrix} \left(Row(R_{A,i}R_X)_1 \times Col(I)_1 - Row(R_{X,i})_1 \times Col(R_{B,i})_1\right)^T \\ \left(Row(R_{A,i}R_X)_2 \times Col(I)_1 - Row(R_{X,i})_2 \times Col(R_{B,i})_1\right)^T \\ \left(Row(R_{A,i}R_X)_3 \times Col(I)_1 - Row(R_{X,i})_3 \times Col(R_{B,i})_1\right)^T \end{matrix} \\ 0_{3\times 3} & \begin{matrix} \left(Row(R_{A,i}R_X)_1 \times Col(I)_2 - Row(R_{X,i})_1 \times Col(R_{B,i})_2\right)^T \\ \left(Row(R_{A,i}R_X)_2 \times Col(I)_2 - Row(R_{X,i})_2 \times Col(R_{B,i})_2\right)^T \\ \left(Row(R_{A,i}R_X)_3 \times Col(I)_2 - Row(R_{X,i})_3 \times Col(R_{B,i})_2\right)^T \end{matrix} \\ 0_{3\times 3} & \begin{matrix} \left(Row(R_{A,i}R_X)_1 \times Col(I)_3 - Row(R_{X,i})_1 \times Col(R_{B,i})_3\right)^T \\ \left(Row(R_{A,i}R_X)_2 \times Col(I)_3 - Row(R_{X,i})_2 \times Col(R_{B,i})_3\right)^T \\ \left(Row(R_{A,i}R_X)_3 \times Col(I)_3 - Row(R_{X,i})_3 \times Col(R_{B,i})_3\right)^T \end{matrix} \end{bmatrix} \qquad (2.13a)$$

where J_i is a 12x6 matrix.

If the orientation of the sensor frame is not measured, $R_{B,i}$ is not known, and above J_i cannot be used. The relevant part of the Jacobian is then

$$J_i = \begin{bmatrix} -(R_{A,i} - I)R_X & -\begin{bmatrix} \left(Row(R_{X,i})_1 \times p_{B,i}\right)^T \\ \left(Row(R_{X,i})_2 \times p_{B,i}\right)^T \\ \left(Row(R_{X,i})_3 \times p_{B,i}\right)^T \end{bmatrix} \end{bmatrix} \qquad (2.13b)$$

In this case, J_i is a 3x6 matrix. When orientation measurements of B_i are available, Equations (2.12a) and (2.12b) can be written as

$$z_i = J_i\,\eta \qquad i = 1, 2, ..., m \qquad (2.14a)$$

where J_i is given in (2.13a). If the orientations of B_i are not measured, Equation (2.12b) can be written as

$$p_{z,i} = J_i\,\eta \qquad i = 1, 2, ..., m \qquad (2.14b)$$

where J_i is given in (2.13b). Equation (2.14) gives the relationship between the measurement residual error vector z_i (or $p_{z,i}$) and the pose error vector η of X for the ith measurement. J_i is termed the *Identification Jacobian* for wrist-mounted robotic sensor calibration. By using the Identification Jacobian, η may be computed from the residual error vector, and can then be used to update the estimate of X.

2.3 Observability of the Error Parameters of X

In this section, the observability of the pose error vector η of the unknown matrix X is investigated. The observability of kinematic error parameters is defined in terms of the Identification Jacobian in robot calibration literature. It is said that if the Identification Jacobian has a full rank, the error parameters are *observable* [7].

Theorem 2.1: Let m = 2. Assume that only position measurements, that is only Equation (2.12b) is used for the identification purpose. η is observable only if

1. the rotation axes of $R_{A,1}$ and $R_{A,2}$ are neither parallel nor antiparallel one to another, and the rotation angles of $R_{A,1}$ and $R_{A,2}$ are both nonzero; and
2. $p_{B,1}$ and $p_{B,2}$ are nonzero and neither parallel nor antiparallel.

Proof: Equation (2.13b) can be rewritten as

$$J_i = [\,(I - R_{A,i})R_X \quad -R_X\Omega(p_{B,i})\,] \quad i = 1, 2, ..., m \qquad (2.15)$$

If there are two sets of measurements, by stacking Equation (2.14b), one has

$$p_z = J\,h \qquad (2.16a)$$

where $p_z = [p_{z,1}^T, p_{z,2}^T]^T$, and

$$J = \begin{bmatrix} (I - R_{A,1})R_X & -R_X\Omega(p_{B,1}) \\ (I - R_{A,2})R_X & -R_X\Omega(p_{B,2}) \end{bmatrix}. \qquad (2.16b)$$

From Equation (2.16b), h is observable if the rank of the 6x6 Jacobian matrix J is 6. $Rank(J) = 6$ only if its columns are linearly independent. According to Lemma 2.2, the first three columns of J are linearly independent if the first condition is satisfied. The last three columns can be written as

$$\begin{bmatrix} R_X\Omega(p_{B,1}) \\ R_X\Omega(p_{B,2}) \end{bmatrix} = \begin{bmatrix} R_X & 0_{3\times3} \\ 0_{3\times3} & R_X \end{bmatrix} \begin{bmatrix} \Omega(p_{B,1}) \\ \Omega(p_{B,2}) \end{bmatrix} \qquad (2.17)$$

These columns are linearly independent only if the two multiplicative matrices in the right hand side of Equation (2.17) have full rank. Since R_X is always full rank, the condition is reduced to $Rank(P^TP) = 3$, where

$$P \equiv \begin{bmatrix} \Omega(p_{B,1}) \\ \Omega(p_{B,2}) \end{bmatrix}.$$

Let D_P be the determinant of P. The necessary (and also the sufficient) condition for $Rank(P^TP) = 3$ is that $D_P \neq 0$. By using the symbolic reduction package *MACSYMA* [8], it was shown that

$$D_P = (\|p_{B,1}\|^2 + \|p_{B,2}\|^2)(\|p_{B,1} \times p_{B,2}\|^2).$$

$D_P \neq 0$ if and only if the second condition is satisfied. Q.E.D.

Remarks:

1. Theorem 2.1 suggests that X may be obtained by using the position submatrices of B_i only.
2. Physically, $p_{B,1} \times p_{B,2} \neq 0$ means that the robot sensor has moved along different directions during the measurement process. This condition is imposed due to the fact that only sensor position measurements are used for identification.

2.4 An Iterative Procedure

In this section, we propose a procedure to update the unknown matrix X. The method employs only the Identification Jacobian given in Section 2.2. Moreover, it preserves the orthonormality of R_X.

Let $z = [z_1, z_2, ..., z_m]^T$, $p_z = [p_{z,1}, p_{z,2}, ..., p_{z,m}]^T$, and $J = [J_1^T, J_2^T, ..., J_m^T]^T$. Recall that z_i is the residual vector from the ith measurement, $p_{Z,i}$ is a column vector consisting of the first three elements of z_i, and m is the number of measurements. Note that if complete sensor pose measurements are taken, J_i is in the form of (2.13a), in which case J is a (12m)x6 matrix, and (2.14a) can be written in a single matrix form,

$$z = J\,\eta. \tag{2.18}$$

On the other hand, if only sensor position measurements are available, J_i is in the form of (2.13b), and J is a (3m)x6 matrix. Equation (2.14b) then be written as

$$p_z = J\,\eta. \tag{2.19}$$

Given z and J, η may be solved from Equation (2.18) using the least squares technique. The solution of the problem at hand is obtained iteratively as indicated next. From a given initial guess X^0, the sequence $X^0, X^1, ..., X^k, ...$ is generated by

$$X^{k+1} = X^k + dX^k \tag{2.20}$$

where $dX^k = X^k \delta X^k$. Recall that the pose error vector is made up of d^k and d^k, i.e. $\eta^k \equiv [d^k, \delta^k]^T$. From d^k and δ^k, δX^k can be computed by

$$\delta X^k = \begin{bmatrix} \Omega(\delta^k) & d^k \\ 0_{1\times3} & 0 \end{bmatrix}$$

However, (2.20) does not preserve the orthonormality of R_X, which can be avoided by using the following update rule [9],

$$X^{k+1} = X^k Trans(dx^k, dy^k, dz^k) Rot(x, \delta x^k) Rot(y, \delta y^k) Rot(z, \delta z^k) \tag{2.21}$$

The kth pose error vector η^k may be computed as the least squares solution of Equation (2.19), that is

$$\eta^k = J^{k+} z^k \tag{2.22}$$

where $J^{k+} = ((J^k)^T J^k)^{-1} (J^k)^T$. A more robust way is to use the Levenberg-Marquardt algorithm [6], or the so-called damped least-squares method [10,11]. The solution is obtained by solving the following equation

$$\eta^k = ((J^k)^T J^k + \lambda I)^{-1} (J^k)^T z^k \tag{2.23}$$

where λ is a small positive number.

The convergence criterion is

$$\|\eta^k\| < \varepsilon \tag{2.24}$$

where ε is a prescribed tolerance.

Note that if only sensor positions are measured, z^k should be replaced by p_z^k in (2.22) and (2.23).

3. Simulation Results

We tested the algorithm by generating various samples of X and A_i from which B_i can be computed. We chose 80 different parameters uniformly covering the parameter space of X. The norms of p_X were kept at 800 mm. For each X, up to 15 "measurements" are computed. A_i were also randomly selected with constraints that the norms of p_A were in a range of [500mm, 1000mm]. The homogeneous transformations B_i were solved from $B_i = X^{-1} A_i X$.

Three algorithms were extensively tested in the simulation:

1. the two-stage method proposed in [4],
2. the iterative one-stage method with full pose measurements of B_i, and
3. the iterative one-stage method with position measurements of B_i only.

If measurement noise is not injected into the system, all of the three algorithms produces correct results for all the tries.

Let us now consider an actual system. A_i is computed by using a forward kinematic model of the robot and joint readings; thus it is orthonormal, although it may not be accurate. Adding angular noise preserves orthonormality of A_i. On the other hand, B_i is obtained in general by an external sensing device; therefore it may not be orthonormal. Orthonormalization of B_i before plugging them into either the Jacobian matrix or (1.1) is unnecessary since both the two-stage and one-stage algorithms can handle the case that B_i is not orthonormal. To simulate a real system, measurement uncertainties were modeled as uniformly distributed random numbers added to the three Euler angles of A_i and the first two columns of B_i. In the latter case, the third column is obtained by taking the cross product of the respective first two columns.

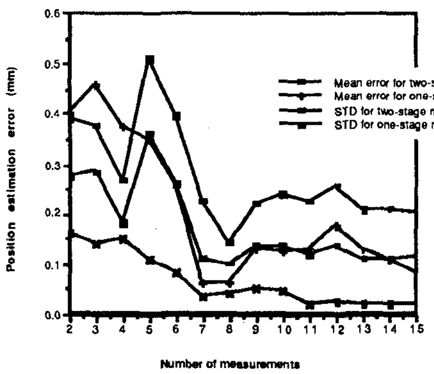
Different types of noise intensity were tested. $\|d_i\|$ and $\|\delta_i\|$ computed from Equation (2.6) were used to define the *position and rotation errors of* X, that is, the deviation of an estimate from the true rotation. Means and standard deviations of $\|d_i\|$ and $\|\delta_i\|$ were computed for each set of measurements from all 80 samples of X, providing essentially the first two moments of the ensemble statistics.

Table 3.1 lists three different types of noise used in simulations. U[a, b] denotes the uniformly distributed random noise in a range of [a, b]. Type 1 noise models a moderate noise level for both position and orientation, type 2 noise emphasizes large orientation measurement errors, and type 3 noise exaggerates position measurement errors.

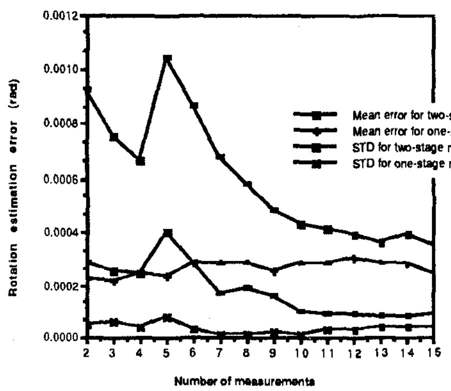
Table 3.1 Noise intensity levels used in simulation

	Position noise (mm)	Noise on angles of R_A(rad)	Noise on elements of R_B
1	U[-0.25, 0.25]	U[-0.0005, 0.0005]	U[-0.0005, 0.0005]
2	U[-0.25, 0.25]	U[-0.0025, 0.0025]	U[-0.0025, 0.0025]
3	U[-1.0, 1.0]	U[-0.0005, 0.0005]	U[-0.0005, 0.0005]

The performance of the algorithms is illustrated by Figures 3.1-3.3. Simulations consistently showed that the one-stage iterative algorithm using either pull poses of B_i or its position vectors produced almost identical results; therefore, only one curve, namely the curve for the one-stage method, is used for both algorithms.

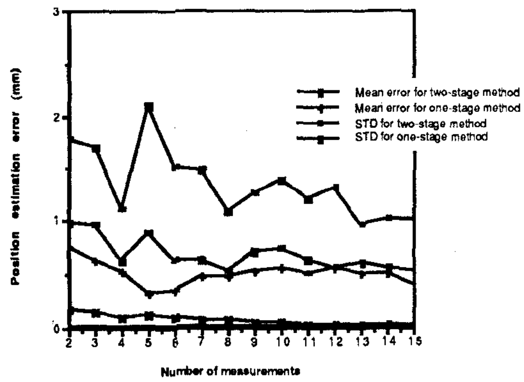


(a)

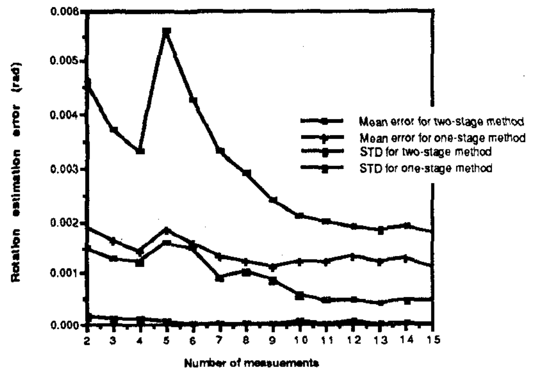


(b)

Figure 3.1 Estimation errors for type 1 noise



(a)



(b)

Figure 3.2 Estimation errors for type 2 noise

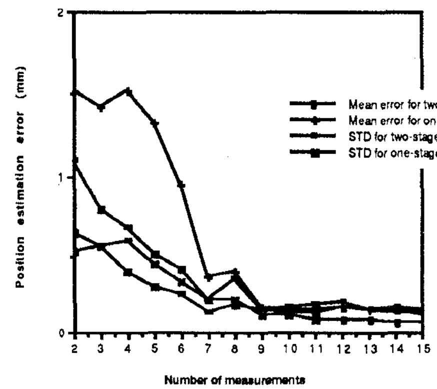
The horizontal axes in Figures 3.1-3.3 represent the number of homogeneous transformation equations for the estimation problem. The vertical axes of Figures 3.1-3.3 show the mean and standard deviation of the rotation and position estimation errors.

By studying the simulation results the following remarks are made:

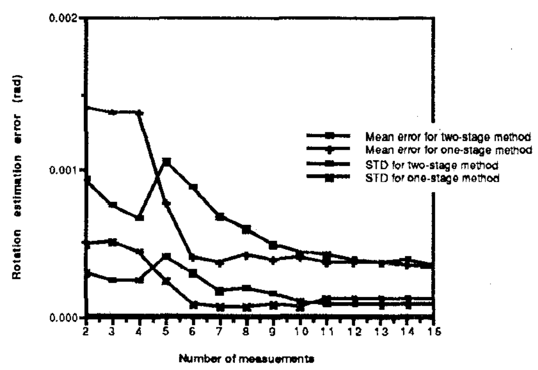
1. In general, the one-stage algorithm performs better than the two-stage algorithm.
2. When the noise intensity of rotation measurement is increased, as modeled by type 2 noise, the advantage of the one-stage algorithm diminished, since small rotational noise causes little prorogation errors to the estimation of the position vector of X for the two-stage method.
3. When the noise intensity of position measurement is dominant, as modeled by type 3 noise, the performance of the two-stage algorithm is improved in terms of position estimation errors. This is due to smaller rotation errors propagated to position errors. At the extreme, when there is no rotation measurement noise, there will be no propagation error in the two-stage algorithm.
4. The iterative algorithm usually converges after a few

iterations if the initial conditions are within a ball of 20 mm in radius for position parameters of X and a ball of 0.01 rad for its rotation parameters; both are centered at their optimal paramters.

5. On the whole, by using more measurements, estimation errors decrease gradually. This trend slows down significantly after the number of measurements are greater than eight.
6. Adding measurements can sometimes degrade estimation quality, as seen in Figures 3.1-3.3.
7. The computation time of the iterative algorithms is 2-7 times longer (not listed) than that of the two-stage algorithm. However, since wrist-mounted sensor calibration is usually done off-line, this is not a major concern.
8. A weighting matrix may be used in the cost function (2.3). By selecting different values of the elements in the weighting matrix, one can emphasize the position estimation accuracy over the orientation estimation accuracy or vice versa.



(a)



(b)

Figure 3.3 Estimation errors for type 3 noise

4. Conclusions

Accuracy is the most important aspect for robot wrist-mounted sensor calibration. The calibration accuracy of the iterative one-stage algorithm is in general higher than that of two-stage algorithms since propagation errors are eliminated. This one-stage algorithm has the additional advantage of being able to identify the unknown transformation without sensor orientation measurement. This is important in practice since position is easier to measure than orientation. The iterative algorithm needs more computation time to produce results. However, due to the fact that wrist-mounted sensor calibration is usually performed off-line, this is not a major concern. Owing to its merits, the iterative approach can be a viable candidate for calibration of robot wrist-mounted sensors.

Since only position measurements of the sensor are needed, the approach presented in this paper has also great potential applications for wrist-mounted tool calibrations.

References

[1] Shiu, Y.C. and S. Ahmad, "Calibration of Wrist-Mounted Robotic Sensors by Solving Homogeneous Transform Equations of the Form AX=XB," *IEEE Trans. Robot. Automat.*, Vol. 5, No. 1, Feb. 1989, pp. 16-27.

[2] Tsai, R.Y. and R. K. Lenz, "A New Technique for Fully Autonomous and Efficient 3D Robotics Hand/Eye Calibration," *IEEE Trans. Robot. Automat.*, Vol. 5, No. 3, June 1989, pp. 345-357.

[3] Chou, J. K. and M. Kamel, "Quaternions Approach to Solve the Kinematic Equation of Rotation, $A_d A_x = A_x A_b$, of a Sensor-Mounted Robotic Manipu-lator," *Proc. IEEE Int. Conf. Robot. Automat.*, PA, April 1988, pp. 656-662.

[4] Zhuang , H. and Z.S. Roth, "Comments on 'Calibration of Wrist-Mounted Robotic Sensors by Solving Homogeneous Transform Equations of the Form AX = XB'," *IEEE Trans. Robot. Automat.*, Vol. 7, No. 6, Dec. 1991, pp. 877-878.

[5] Gill, P.E., W. Murray, and M.H. Wright, *Practical Optimization*, Academic Press, 1981.

[6] Marquardt, D.W., "An Algorithm for Least-squares Estimation of Nonlinear Parameters," *J. Soc. Indust. Appl. Math.*, Vol. 11, No. 2, June 1963, pp. 431-441.

[7] Zhuang, H., Z.S. Roth, and F. Hamano, "Observability Issues in Kinematic Error Parameter Identification of Manipulators," *ASME J. Dynamic Systems, Meas. Control*, in press.

[8] *MACSYMA Reference Manual*, Symbolics, Inc., 1985.

[9] Paul, R., *Robot Manipulators: Mathematics, Programming, and Control*, MIT Press, 1981.

[10] Nakamura, Y.,H. Hanafusa, and T. Yoshikawa, "Task-Priarity Based Redundancy Control of Robot Manipulators," *Int. J. Robot. Res.*, Vol. 6, No. 1, 1987, pp 3-15.

[11] Wampler, C.W., "Manipulator Inverse Kinematic Solutions Based on Vector Formulations and Damped Least-Squares Methods," *IEEE Trans. SMC*. Vol. 16, No. 1, 1986, pp. 93-101.

[12] Zhuang, H and Y.C. Shiu, "An Iterative Algorithm for Wrist-Mounted Robotic Sensor Calibration with or without External Orientation Measurement," *proc. of ACC '92*.