Simultaneous Coordinate Calibrations by Solving the AX=YB Problem without Correspondence*

Haiyuan Li[1], Qianli Ma[2], Tianmiao Wang[1] and Gregory Chirikjian[2]

***Abstract*— In image-guided systems, the relative transformations of hand-eye (X) and robot-world (Y) coordinates have to be calculated and a simultaneous solution is useful in sensor calibration problem. Due to the asynchrony of sensors' timing, the exact correspondence between A and B is unknown, and a common scenario is when there is a constant shift between the two data streams. A probabilistic method is presented to solve the homogeneous matrix equations without a priori knowledge of the correspondence. Using the Euclidean-Group invariants, an exact solution can be found. For noisy and shifted data streams, we numerically simulated the proposed method, and the results show the efficiency and robustness.**

I. Introduction

Image-guided systems have been widely used in robotics such as robot assisted surgery, automatic guided vehicle, etc. Sensors such as a camera, a laser scanner or an ultrasound probe are usually mounted on the distal end of a robotic manipulator. For a typical "hand-eye" system as described above, the relative transformation of the sensor with respect to the end-effector should be accurately calibrated, and it is often characterized as the well known **AX=XB** problem. A variation of this problem is the **AX=YB** problem, where both the hand-eye transformation and the pose of the robot base with respect to the world frame need to be calibrated. In a typical environment setup, the relationships among the sensor frame, robot frame and world frame are variant and the uncertainties exist. Therefore, simultaneous coordinate calibrations have to be determined frequently in order to enable the robots to respond to dynamic environments.

In the **AX=YB** problem, data streams As and Bs can be respectively obtained via different sensors. The data streams can be in an asynchronous fashion due to the different working frequencies of the sensors. The asynchrony causes a shift between the two streams of data which damages the correspondence between As and Bs. In this paper, a novel method is presented to solve for X and Y without the need to know a priori knowledge of the correspondence between As and Bs.

The hand-eye calibration problem can be modeled as **AX = XB**, where A and B are the homogeneous transformation matrices describing the relative motions of the end-effector and the sensor respectively. As shown in Fig. 1 part (2), $A = A^i(A^{i+1})^{-1}$ and $B = B^{i+1}(B^i)^{-1}$. Given multiple pairs of (A_i, B_i) with correspondence (note that (A_i, B_i) is the relative transformations obtained from the raw data), many methods have been proposed to solve for X. To the best of the authors' knowledge, Shiu [1] and Tsai [2] are the first to solve the $AX = XB$ sensor calibration problem. The other methods include but are not limited to the quaternion, dual quaternion, screw theory, Lie group theory, convex optimization and gradient descent methods [3]–[9]. All of the methods above assume a prior knowledge of exact correspondence between A_i and B_i. For data streams $\{A_i\}$ and $\{B_j\}$ that are asynchronous, several methods have been proposed in the literature to solve for X using data without

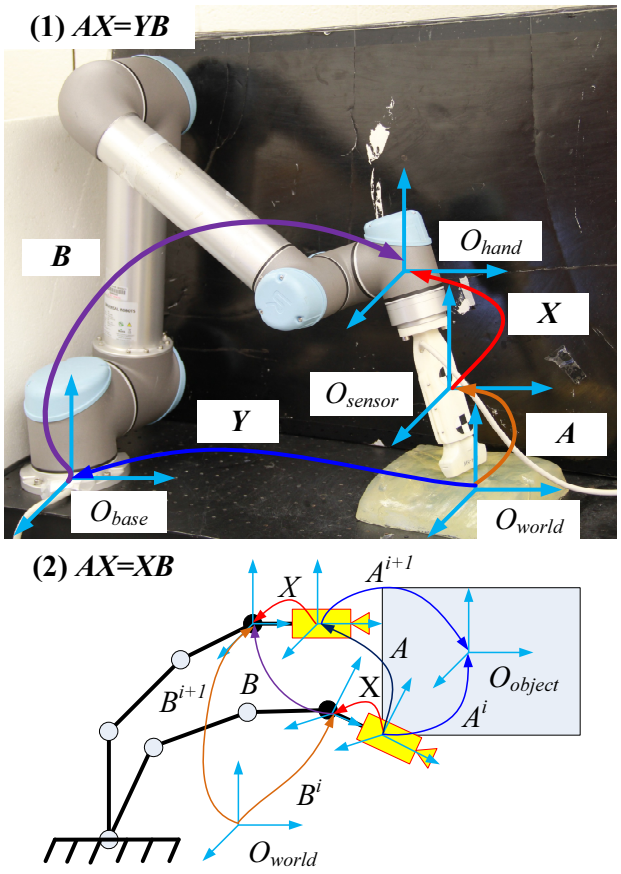

Fig. 1. (1) The Hhand-eye and Robot-world Calibration Problem Which is Formulated as AX=YB (The Universal Robot as Shown in the Picture is Owned by Professor Emad Boctor in the Johns Hopkins University). (2) The Hand-eye Calibration Problem which is Formulated as AX=XB.

*This work is partially supported by NSF Grant RI-Medium: #IIS-1162095

[1]Haiyuan Li and Tianmiao Wang are with the School of Mechanical Engineering and Automation, Beihang University, ,Beijing 100191, China haiyuanli@hotmail.com, itm@buaa.edu.cn

[2]Qianli Ma and Gregory Chirikjian with the Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD 21218, USA mqianli1@jhu.edu, gregc@jhu.edu

correspondence. Theses methods assume that there is exact knowledge of the correspondence between $\{A_i\}$ and $\{B_i\}$ [10]–[12].

Simultaneous estimation of the hand-eye and robot-world transformations has been viewed as the **AX=YB** problem. As shown in Fig. 1 part (1), Y is the transformation from the robot base to the world frame, A denotes the pose of the sensor in the world frame and B is the transformation from the end-effector to its fixed base. The A and B in **AX=YB** are different from those in **AX=XB** where the former uses absolute transformations and the latter uses relative transformations. This problem has been solved by many different methods such as the Kronecker product, quaternion, dual quaternion, and nonlinear optimization methods [13]–[20]. Simulatneous calibrations of X and Y can be problematic in that all the methods above assume exact correspondence between $\{A_i\}$ and $\{B_j\}$, which is not the case in the real world, and this is why simultaneous solution for X and Y in **AX=YB** problem can be a challenging issue. Another similar problem involves the calibration of multiple robots in terms of hand-eye, tool-flange and robot-robot system, and it is formulated as the **AXB=YCZ** problem [21] which will not be discussed in detail here. In the above methods, the correspondence between A and B is known a prior. In this paper, we focus on one case of **AX=YB** problem where there is no a priori knowledge of the correspondence between the data streams.

The rest of the paper is organized as follows. In Section II, a novel probabilistic method is presented to solve for eight candidates of X and Y. In Section III, an algorithm involving both correlation theorem and Euclidean group invariants is proposed to recover the correspondence between $\{A_i\}$ and $\{B_j\}$, which is used to select the optimal solution among the candidates. The simulation results obtained by taking noisy data without correspondence are illustrated in Section IV. Finally, conclusions are drawn based on the numerical results and possible future works are pointed out.

II. Solving AX=YB using a probabilistic method on motion groups

In this section, a brief introduction to the concepts of probability density function on the special Eucledian group $SE(3)$ is presented and the probabilistic representation of $AX = YB$ are derived.

Any rigid transformation matrix can be viewed as a group element of $SE(3)$:

$$H(R,t) = \begin{pmatrix} R & t \\ 0^T & 1 \end{pmatrix} \in SE(3), \quad R \in SO(3) \tag{1}$$

where $SO(3)$ denotes the special orthogonal group, $t \in \mathbb{R}^3$ is a translational vector and H is the symbol for group element.

Given a large set of pairs $(A_i, B_i) \in SE(3) \times SE(3)$ where $i = 1, \cdots, n$, the following equation is true if the correspondence is known as a priori:

$$A_i X = Y B_i. \tag{2}$$

For a group element $H \in SE(3)$, a Dirac delta function $\delta(H)$ is defined to be finite only at the identity and zero elsewhere:

$$\delta(H) = \begin{cases} +\infty, & H = I \\ 0, & H \neq I. \end{cases} \tag{3}$$

The Dirac delta function also satisfies the identity constraint as:

$$\int_{SE(3)} \delta(H) dH = 1. \tag{4}$$

A shifted Dirac delta function can be defined as $\delta_A(H) = \delta(A^{-1}H)$. Given $K, H \in SE(3)$ and two well-defined functions $f_1, f_2 \in \left(L^1 \cap L^2\right)(SE(3))$, their convolution on $SE(3)$ is defined as:

$$(f_1 * f_2)(H) = \int_{SE(3)} f_1(K) f_2(K^{-1} \circ H) dK. \tag{5}$$

where $\circ$ denotes the group product, and in this case it is simply matrix multiplication. Employing the properties of δ function, it is straightforward to see that:

$$(f * \delta)(H) = \int_{SE(3)} f(K) \delta(K^{-1} \circ H) dK = f(H). \tag{6}$$

Therefore, for each A_i and B_i, the following equations can be obtained:

$$(\delta_{A_i} * \delta_X)(H) = \delta(A_i^{-1} H X^{-1}) \tag{7a}$$
$$(\delta_Y * \delta_{B_i})(H) = \delta(Y^{-1} H B_i^{-1}). \tag{7b}$$

Using Eq.(2) and Eq.(3), the above two equations can be combined into a single equation as:

$$(\delta_{A_i} * \delta_X)(H) = (\delta_Y * \delta_{B_i})(H) \tag{8}$$

Define the probability density function of $\{A_i\}$ and $\{B_i\}$ as:

$$f_A(H) = \frac{1}{n} \sum_{i=1}^{n} \delta_{A_i}(H) \tag{9a}$$

$$f_B(H) = \frac{1}{n} \sum_{i=1}^{n} \delta_{B_i}(H) \tag{9b}$$

Using the distributivity of convolution, add n instances of Eq.(9) and substitute Eq.(8) in, and we will have:

$$(f_{A_i} * \delta_X)(g) = (\delta_Y * f_{B_i})(g) \tag{10}$$

If each of the data stream $\{A_i\}$ and $\{B_i\}$ is generated using Gaussian distribution over $SE(3)$, then they can be viewed as "highly focused". The convolution of two highly focused probability density functions (PDF) have some interesting properties that can be used to solve for X. For simplicity, we will drop the integral interval "$SE(3)$" for the rest of equations, and it should be self-clear given the integration variable that is used. In particular, define the mean

M and covariance Σ of a probability density function on $SE(3)$ $f(H)$ as:

$$\int log(M^{-1}H))f(H)dH = \mathbb{O} \tag{11a}$$

$$\Sigma = \int log^{\vee}(M^{-1}H)[log^{\vee}(M^{-1}H)]^T f(H)dH. \tag{11b}$$

where the explicit expression of the matrix logarithm $log(H)$ along with its vectorized form $log^{\vee}(H)$ are given in [22]. If $f_A(H)$ is given as in Eq.(9), then its corresponding discrete version of mean M_A and covariance Σ_A will be:

$$\sum_{i=1}^{n} log(M_A^{-1}A_i)) = \mathbb{O} \tag{12a}$$

$$\Sigma = \sum_{i=1}^{n} log^{\vee}(M_A^{-1}A_i)[log^{\vee}(M_A^{-1}A_i)]^T. \tag{12b}$$

Given $\{A_i\}$ with the cloud of frames A_i clustering around M_A, an iterative formula can be used for computing M_A [23] as:

$${}^{k+1}M_A =^k M_A \circ exp[\frac{1}{n}\sum_{i=1}^{n} log({}^kM_A^{-1} \circ A_i)] \tag{13}$$

An initial estimate of the iterative procedure can be chosen as ${}^0M_A = \frac{1}{n}\sum_{i=1}^{n} log(A_i)$, then a local minimum of M_A is obtained by solving a nonlinear optimization problem with the cost function of $||\sum_{i=1}^{n} log(M_A^{-1}A_i)||^2$. A similar procedure can be used to compute M_B. Σ_A and Σ_B are then straight forward to compute given known M_A and M_B.

The mean and covariance for the convolution $(f_1 * f_2)(g)$ of two highly focused functions f_1 and f_2 are calculated as in [23]:

$$M_{1*2} = M_1 M_2 \tag{14a}$$

$$\Sigma_{1*2} = Ad(M_2^{-1})\Sigma_1 Ad^T(M_2^{-1}) + \Sigma_2. \tag{14b}$$

where

$$Ad(H) = \begin{pmatrix} R & O \\ \hat{t}R & R \end{pmatrix}.$$

Because X and Y are constant, their corresponding PDF will be $\delta_X(g)$ and $\delta_Y(g)$, of which the mean and covariance are $M_X = X$, $\Sigma_X = \mathbb{O}_{6\times6}$ and $M_Y = Y$, $\Sigma_Y = \mathbb{O}_{6\times6}$, respectively. Therefore, the following equations can be obtained using Eq.(14):

$$M_A X = Y M_B \tag{15a}$$

$$Ad(X^{-1})\Sigma_A Ad^T(X^{-1}) = \Sigma_B. \tag{15b}$$

To solve the above equations, Eq.(15a) is decomposed into a rotational equation and a translational equation as below:

$$R_{M_A} R_X = R_Y R_{M_B} \tag{16a}$$

$$R_{M_A} t_X + t_{M_A} = R_Y t_{M_B} + t_Y. \tag{16b}$$

Σ_A and Σ_B can be decomposed into blocks as $\begin{pmatrix} \Sigma_A^1 & \Sigma_A^2 \\ \Sigma_A^3 & \Sigma_A^4 \end{pmatrix}$ and $\begin{pmatrix} \Sigma_B^1 & \Sigma_B^2 \\ \Sigma_B^3 & \Sigma_B^4 \end{pmatrix}$, respectively. Knowing that $X^{-1} = \begin{pmatrix} R_X^T & -R_X^T t_X \\ 0 & 1 \end{pmatrix}$, then the first two blocks of Eq.(15b) can be written as follows:

$$\Sigma_{M_B}^1 = R_X^T \Sigma_{M_A}^1 R_X \tag{17a}$$

$$\Sigma_{M_B}^2 = R_X^T \Sigma_{M_A}^1 R_X \widehat{(R_X^T t_X)} + R_X^T \Sigma_{M_A}^2 R_X. \tag{17b}$$

Because Eq.(17a) is a similarity transformation between $\Sigma_{M_B}^1$ and $\Sigma_{M_A}^1$, they share the same eigenvalues and can be eigendecomposed into $\Sigma_{M_A}^1 = Q_{M_A}\Lambda Q_{M_A}^T$ and $\Sigma_{M_B}^1 = Q_{M_B}\Lambda Q_{M_B}^T$ where Λ is a diagonal matrix whose diagonal elements are the eigenvalues of $\Sigma_{M_A}^1$ (or $\Sigma_{M_B}^1$), and Q_{M_A} (or Q_{M_B}) is a square matrix whose columns are the corresponding eigenvectors. The following equation is obtained after substituting $\Sigma_{M_B}^1$ and $\Sigma_{M_A}^1$ into Eq.(17a):

$$\Lambda = (Q_{M_A}^T R_X^T Q_{M_B})\Lambda(Q_{M_B}^T R_X Q_{M_A}) = P\Lambda P^T \tag{18}$$

where $P = Q_{M_A}^T R_X Q_{M_B}$. If Q_{M_A} and Q_{M_B} are further constrained to be rotation matrices, then a rotation matrix P that satisfies Eq.(18) can be one of $\mathcal{P}$ or $-\mathcal{P}$:

$$\mathcal{P} = \left\{ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \right\}. \tag{19}$$

Therefore, there are eight candidates of R_X which can be calculated via $R_X = Q_{M_A} P Q_{M_B}^T$, and the corresponding t_X can be obtained from Eq.(17b). Given known X, Y can be solved for by $Y = M_A^{-1} X M_B^{-1}$. At last, eight candidate pairs of $\{X_k, Y_k\}$ can be obtained as:

$$X_k = \begin{pmatrix} R_{X_k} & t_{X_k} \\ \mathbf{0}^T & \mathbf{1} \end{pmatrix}, \quad Y_k = \begin{pmatrix} R_{Y_k} & t_{Y_k} \\ \mathbf{0}^T & \mathbf{1} \end{pmatrix} \tag{20}$$

where $k = 1, 2, ..., 8$.

The problem then becomes selecting the best pair of $\{X_k, Y_k\}$ from the eight candidates. Based on the screw theory, it is known that a homogeneous transformation H can be expressed by the four screw parameters $(\theta, d, \mathbf{n}, \mathbf{p})$ that define the $Plücker$ coordinates of the screw motion as:

$$H = \begin{pmatrix} e^{\theta\hat{\mathbf{n}}} & (\mathbf{I}_3 - e^{\theta\hat{\mathbf{n}}})\mathbf{p} + d\mathbf{n} \\ \mathbf{0}^T & \mathbf{1} \end{pmatrix} \tag{21}$$

where θ is the angle of rotation, d is the translation along the rotation axis, $\mathbf{n}$ is the unit vector representing the axis of

rotation and $\mathbf{p}$ is the position of the line to the origin with $\mathbf{p} \cdot \mathbf{n} = 0$.

Moreover, $AX_k = Y_kB$ can be written as $AX_k = X_k(X_k^{-1}Y_kB)$. Define $B^k = X_k^{-1}Y_kB$, and we will have $AX_k = X_kB^k$. There exist two Euclidean-Group invariant relationships for each pair of $(A_i, B_i^k)(i = 1, \cdots, n; k = 1, \ldots, 8)$ as follows:

$$\theta_{A_i} = \theta_{B_i^k}, d_{A_i} = d_{B_i^k} \tag{22}$$

Among the eight pairs (X_k, Y_k), one can find an optimal solution which minimizes the cost function defined as:

$$(X, Y) = \underset{(X_k, Y_k)}{\mathbf{arg}min} \frac{1}{n} \sum_{i=1}^{n} (\parallel \theta_{A_i} - \theta_{B_i^k} \parallel + \parallel d_{A_i} - d_{B_i^k} \parallel) \tag{23}$$

Eight candidates of (X_k, Y_k) are calculated using the probabilistic method on $SE(3)$, which doesn't require the correspondence between A_i and B_j. However, the correspondences needs to be recovered to pick the optimal (X_k, Y_k).

III. Solution with unknown correspondence between A_i and B_i^k

In most cases, the two sets of homogeneous transformations $\{A_i\}$ and $\{B_j\}$ are calculated based on the data obtained from different sensors. Due to the asynchronous timing of the sensor readings, the correspondence between $\{A_i\}$ and $\{B_j\}$ is usually unknown. This section deals with the case where there is a shift between $\{A_i\}$ and $\{B_j\}$, and the Euclidean-Group invariants are used to recover the correspondence between the data streams. The advantage of the above probabilistic solution lies in that X and Y can be calculated even if there is not a priori knowledge of the correspondence. However, there are still eight possible candidates of (X_k, Y_k) to choose from and by using Euclidean-Group invariants, it is straightforward to determine which pair is the optimal one if the correspondence between A_i and B_i^k can be known.

The Discrete Fourier Transform (DFT) decomposes a time-domain signal into its constituent frequencies. The input is a finite list of equally spaced samples of a function. Given a discrete signal consisting of a sequence of N complex numbers $x_0, x_1, \cdots, x_{N-1}$, the DFT is denoted by $X_\kappa = \mathcal{F}(x_n)$ as:

$$X_\kappa = \sum_{n=0}^{N-1} x_n \cdot exp(-i\frac{2\pi}{N} n\kappa). \tag{24}$$

where i here is the imaginary unit.

The Inverse Discrete Fourier transform (IDFT) is denoted as:

$$x_n = \frac{1}{N} \sum_{n=0}^{N-1} X_\kappa \cdot exp(i\frac{2\pi}{N} n\kappa). \tag{25}$$

The discrete convolution of two sequences f_n and g_n are defined

$$(f * g)(\tau) = \sum_{i=0}^{N} f(t_i) g(t_i - \tau). \tag{26}$$

In the convolution theorem, the Fourier transform of a convolution is the product of the Fourier transforms, namely:

$$f * g = \mathcal{F}^{-1}[\mathcal{F}(f) \cdot \mathcal{F}(g)]. \tag{27}$$

The correlation theorem indicates that the correlation function, $Corr(f, g)$, will be larger for a shift vector where the two sequences f_n and g_n can share more similar features. The correlation can be obtained based on the convolution theorem. The DFT of $Corr(f, g)$ is equal to the product of the DFT of f_n and the complex conjugate $\mathcal{F}^*$ of the DFT of g_n:

$$Corr(f, g) = f \star g = \mathcal{F}^{-1}[\mathcal{F}(f) \cdot (\mathcal{F}(g))^*]. \tag{28}$$

Compared to the standard time-domain convolution algorithm, the complexity of the convolution by multiplication in the frequency domain is significantly reduced with the help of the convolution theorem and the fast Fourier transform (FFT).

Given two sequences $\{\theta_{A_i}\}$ and $\{\theta_{B_i^k}\}$ corresponding to $\{A_i\}$ and $\{B_i^k\}$, the shift that is needed to recover the data correspondence is obtained as below. Firstly, θ_{Ai} and $\theta_{B_i^k}$ are normalized as:

$$\theta_{1i} = \frac{(\theta_{A_i} - \mu_A)}{\sigma_A}, \theta_{2i} = \frac{(\theta_{B_i^k} - \mu_{B^k})}{\sigma_{B^k}} \tag{29}$$

where $\mu_A(\mu_{B^k})$ is the mean of $\theta_{A_i}(\theta_{B_i^k})$ and $\sigma_A(\sigma_{B^k})$ is the standard deviation. Here, the correlation function $Corr(\theta_1, \theta_2)$ is the function of the time sequence index n which describes the probability of these two sequences being separated by this particular index. The index corresponding to the maximum of $Corr(\theta_1, \theta_2)$ indicates the amount of shift τ_{shift} between $\{\theta_{A_i}\}$ and $\{\theta_{B_i^k}\}$.

$$\tau_{shift} = \underset{index}{\mathbf{arg}max}(Corr(\theta_1, \theta_2)) \tag{30}$$

Therefore, the correspondence between the two sequences can be found. The data of θ_{A_i} or d_{A_i} are shifted by $-\tau_{shift}$ to obtain a sequence of new pairs $(\theta_{A_i}(i + \tau_{shift}), \theta_{B_i^k})$ and $(d_{A_i}(i + \tau_{shift}), d_{B_i^k})$, where $max(0, \tau_{shift}) \leq i \leq min(n, n + \tau_{shift})$. The data stream can be shifted back to regain correspondence to synchronize the data streams once the shift is computed, and the correct solution of X and Y can also be recovered by minimizing the cost function Eq.(23) using the Euclidean-Group invariants as shown in Section II.

IV. Simulation Studies

For the numerical experiments in this section, the rotational and translational errors for X and Y are measured as $Error(R_X) = \parallel log^\vee(R_{X_{Solved}}^T R_{X_{true}}) \parallel$, $Error(t_X) = \parallel$

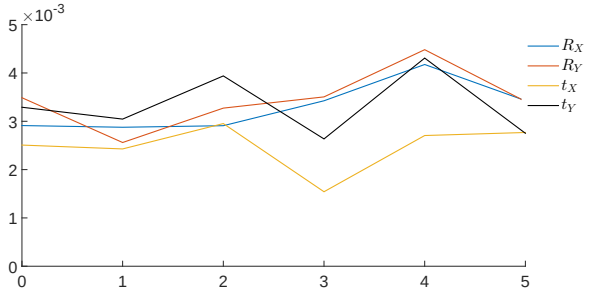

Fig. 2. The Errors of Translation and Rotation Versus the Shift between Data Streams $\{A_i\}$ and $\{B_i\}$.

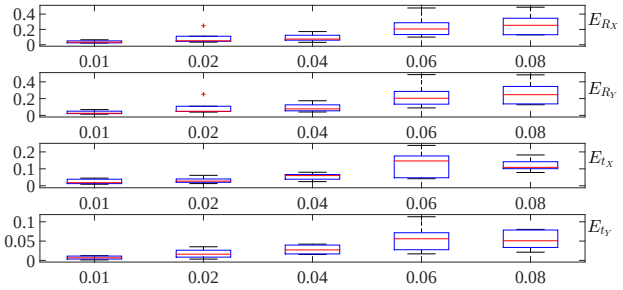

Fig. 3. Box-and-Whisker Plots of Translational and Rotational Errors Versus the Covariance of Noise on Data Stream $\{B_i\}$.

$t_{X_{Solved}} - t_{X_{true}} \parallel$, $Error(R_Y) = \parallel log^{\vee}(R^T_{Y_{Solved}} R_{Y_{true}}) \parallel$ and $Error(t_Y) = \parallel t_{Y_{Solved}} - t_{Y_{true}} \parallel$ respectively.

There are multiple ways of generating the data streams $\{A_i\}$ and $\{B_i\}$. One way is to first generate $\{B_i\}$ and then map it to $\{A_i\}$ using $A = YBX^{-1}$. $\{B_i\}$ can be obtained by randomly sampling on the Lie algebra of B from a zero mean multivariate Gaussian distribution as follows:

$$\delta_i \in \mathcal{N}(\mathbf{0}; \Sigma) \subset \mathbb{R}^6 \tag{31a}$$

$$B_i = \exp(\hat{\delta}_i)\exp(\mu) \tag{31b}$$

where the mean $\mu = \mathbf{0} \in se(3)$ and the covariance matrix $\Sigma \in \mathbb{R}^{6\times 6}$ is a diagonal matrix with same diagonal elements σ. The hat operator $\hat{\delta}$ converts a 6 by 1 vector into its corresponding Lie algebra. The data stream $\{A_i\}$ can be

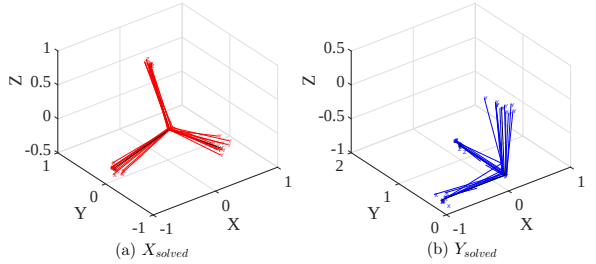

Fig. 4. (a) The Solved X (in Red) and the Actual X (in Black) for 10 Trials of Simulation with the Covariance of Noise of 0.05 and the Shift of 2.(b) The Solved Y (in Blue) and the Actual Y (in Black) for 10 Trials of Simulation with the Covariance of Noise of 0.05 and the Shift of 2.

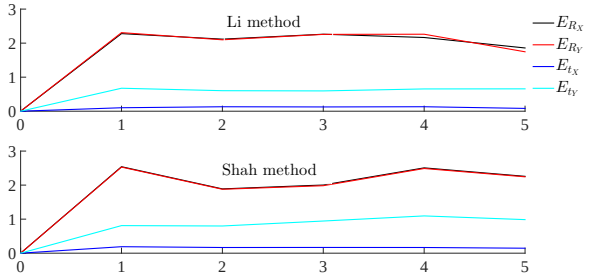

Fig. 5. ErrorS of Orientation and Translation of X and Y Versus Shift Using Li and Shah Method without Correspondence.

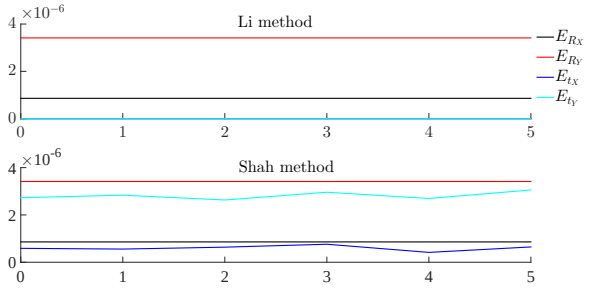

Fig. 6. Errors of Orientation and Translation of X and Y Versus Shift Using Li and Shah Method with Correspondence.

easily obtained as described above. After employing the proposed probabilistic method, 8 sets of sequences $(\theta_{A_i}, \theta_{B_i^k})$ and $(d_{A_i}, d_{B_i^k})$ can be obtained respectively where $i = 1, \cdots, 100$ and $k = 1, \ldots, 8$.

If the data stream $\{A_i\}$ is shifted by m units relative to $\{B_i\}$, then the maximum of the cross correlation can be used to recover the shift. After that, we can shift the data stream $\{A_i\}$ back to its original position to recover the correct correspondence with $\{B_i\}$, which will be used to find a correct solution satisfying the Euclidean-Group invariants as defined in Eq.(22). Therefore, a unique pair of (X_k, Y_k) can be selected to minimize the cost function. In Fig. 2, because the shift between $\{A_i\}$ and $\{B_i\}$ is calculated accurately, the errors of translations and rotations fluctuate by only a small amount compared to the errors of no-shift data streams.

To test the robustness of the proposed method, noises are exerted onto $\{B_i\}$ by employing $B_i^{noise} = B_i \exp(\widehat{\mathbf{x}}_{noise})$, where each element of the Lie Algebra $\mathbf{x}_{noise}$ belongs to the Gaussian distribution defined as $N \sim (\mu_{noise}, \sigma_{noise})$. In Fig. 3, as the covariance of the noise σ_{noise} increments from 0.01 to 0.08, the errors of R_X, R_Y, t_X, and t_Y increase as shown in the box-and-whisker plot. There are several outliers outside the whiskers, while the median is calculated as the final solved X and Y. Fig. 4 shows the solved (X,Y)s in red and blue with the actual (X,Y) in black when the noise covariance $\sigma = 0.05$ and the shift $n = 2$.

The probabilistic method in this paper can recover the correspondence between shifted data streams, which is useful for other sensor calibration methods. In $AX = YB$ problem,

there have been many calibration methods developed for solving X and Y given data streams with correspondence. However, few of them considered the cases without correspondence. When data streams of A and B are shifted or asynchronous, most of these methods fail to give a valid solution. To further test the effectiveness of our method, we shift the data sequence of $\{A_i\}$ by $n = 0, 1, 2, 3, 4, 5$ with respect to the data sequence of $\{B_i\}$. We augment other $AX = YB$ solvers with our probabilistic approach. In Li's method [18], X and Y are solved for at the same time while Shah [19] uses separable solutions to solve for X and Y. In Fig. 5, the correspondence between data streams is unknown and the errors on both rotations and translations are significant. After synchronizing the data streams by using the probabilistic method, Li and Shah's methods achieve the same level of performance as in the no-shift situation as shown in Fig. 6.

V. Conclusions

In this paper, we developed a probabilistic approach to simultaneously obtain X and Y in $AX = YB$ sensor calibration problem. Without a prior knowledge of the correspondence between $\{A_i\}$ and $\{B_j\}$, the proposed probabilistic method on Lie group is used to constrain the possible solutions of X and Y to eight pairs of candidates. Given shifted data streams of $\{A_{i+s}\}$ and $\{B_i\}$, using the correlation theorem with Euclidean group invariants, the correspondence is recovered to determine the correct solution among the eight candidates. In the numerical simulation, the method performs well with different sets of data samples. Future work will be to further develop the existing algorithm to solve for more general asynchronous cases, such as flipped data pairs or missing data in the data streams.

ACKNOWLEDGMENT

Chirikjian's contribution to this material is based upon work supported by (while serving at) the National Science Foundation. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] Y. C. Shiu and S. Ahmad, "Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form ax= xb," *Robotics and Automation, IEEE Transactions on*, vol. 5, no. 1, pp. 16–29, 1989.

[2] R. Y. Tsai and R. K. Lenz, "A new technique for fully autonomous and efficient 3d robotics hand/eye calibration," *Robotics and Automation, IEEE Transactions on*, vol. 5, no. 3, pp. 345–358, 1989.

[3] C.-C. Wang, "Extrinsic calibration of a vision sensor mounted on a robot," *Robotics and Automation, IEEE Transactions on*, vol. 8, no. 2, pp. 161–175, 1992.

[4] F. C. Park and B. J. Martin, "Robot sensor calibration: solving ax= xb on the euclidean group," *IEEE Transactions on Robotics and Automation (Institute of Electrical and Electronics Engineers);(United States)*, vol. 10, no. 5, 1994.

[5] R. Horaud and F. Dornaika, "Hand-eye calibration," *The international journal of robotics research*, vol. 14, no. 3, pp. 195–210, 1995.

[6] K. Daniilidis, "Hand-eye calibration using dual quaternions," *The International Journal of Robotics Research*, vol. 18, no. 3, pp. 286–298, 1999.

[7] I. Fassi and G. Legnani, "Hand to sensor calibration: A geometrical interpretation of the matrix equation ax= xb," *Journal of Robotic Systems*, vol. 22, no. 9, pp. 497–506, 2005.

[8] Z. Zhao, "Hand-eye calibration using convex optimization," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2947–2952.

[9] M. K. Ackerman, A. Cheng, E. Boctor, and G. Chirikjian, "Online ultrasound sensor calibration using gradient descent on the euclidean group," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4900–4905.

[10] M. K. Ackerman and G. S. Chirikjian, "A probabilistic solution to the ax= xb problem: Sensor calibration without correspondence," in *Geometric Science of Information*. Springer, 2013, pp. 693–701.

[11] M. K. Ackerman, A. Cheng, B. Shiffman, E. Boctor, and G. Chirikjian, "Sensor calibration with unknown correspondence: Solving ax= xb using euclidean-group invariants," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 1308–1313.

[12] M. K. Ackerman, A. Cheng, and G. Chirikjian, "An information-theoretic approach to the correspondence-free ax= xb sensor calibration problem," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 4893–4899.

[13] H. Zhuang, Z. S. Roth, and R. Sudhakar, "Simultaneous robot/world and tool/flange calibration by solving homogeneous transformation equations of the form ax= yb," *Robotics and Automation, IEEE Transactions on*, vol. 10, no. 4, pp. 549–554, 1994.

[14] F. Dornaika and R. Horaud, "Simultaneous robot-world and hand-eye calibration," *Robotics and Automation, IEEE Transactions on*, vol. 14, no. 4, pp. 617–622, 1998.

[15] R. L. Hirsh, G. N. DeSouza, and A. C. Kak, "An iterative approach to the hand-eye and base-world calibration problem," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 3. IEEE, 2001, pp. 2171–2176.

[16] F. Ernst, L. Richter, L. Matthäus, V. Martens, R. Bruder, A. Schlaefer, and A. Schweikard, "Non-orthogonal tool/flange and robot/world calibration," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 8, no. 4, pp. 407–420, 2012.

[17] K. H. Strobl and G. Hirzinger, "Optimal hand-eye calibration," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE, 2006, pp. 4647–4653.

[18] A. Li, L. Wang, and D. Wu, "Simultaneous robot-world and hand-eye calibration using dual-quaternions and kronecker product," *Inter. J. Phys. Sci*, vol. 5, no. 10, pp. 1530–1536, 2010.

[19] M. Shah, "Solving the robot-world/hand-eye calibration problem using the kronecker product," *Journal of Mechanisms and Robotics*, vol. 5, no. 3, p. 031007, 2013.

[20] J. Heller, D. Henrion, and T. Pajdla, "Hand-eye and robot-world calibration by global polynomial optimization," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 3157–3164.

[21] J. Wang, L. Wu, M. Q.-H. Meng, and H. Ren, "Towards simultaneous coordinate calibrations for cooperative multiple robots," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 410–415.

[22] G. S. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 2: Analytic Methods and Modern Applications*. Springer Science & Business Media, 2011, vol. 2.

[23] Y. Wang and G. S. Chirikjian, "Nonparametric second-order theory of error propagation on motion groups," *The International journal of robotics research*, vol. 27, no. 11-12, pp. 1258–1273, 2008.