2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Sensor Calibration with Unknown Correspondence: Solving AX=XB Using Euclidean-Group Invariants

Martin Kendal Ackerman[1], Alexis Cheng[2], Bernard Shiffman[3], Emad Boctor[2], Gregory Chirikjian[1]

***Abstract*— The $AX = XB$ sensor calibration problem must often be solved in image guided therapy systems, such as those used in robotic surgical procedures. In this problem, A, X, and B are homogeneous transformations with A and B acquired from sensor measurements and X being the unknown. It has been known for decades that this problem is solvable for X when a set of exactly measured A's and B's, in a priori correspondence, is given. However, in practical problems, the data streams containing the A' and B's will be asynchronous and may contain gaps (i.e., the correspondence is unknown, or does not exist, for the sensor measurements) and temporal registration is required. For the $AX = XB$ problem, an exact solution can be found when four independent invariant quantities exist between two pairs of A's and B's. We formally define these invariants, reviewing and elaborating results from classical screw theory. We then illustrate how they can be used, with sensor data from multiple sources that contain unknown or missing correspondences, to provide a solution for X.**

I. INTRODUCTION

The "AX=XB" sensor calibration problem is frequently encountered in the fields of robotics and computer vision. In this problem A, X, and B are each homogeneous transformations with A and B given from sensor measurements, and X unknown.

It is well known that it is not possible to solve for a unique X from a single exact pair (A, B), but if there are two compatible instances of independent exact measurements, (A^1, B^1) and (A^2, B^2), then the problem can be solved. Here we address the issue of sensor data that may be unsynchronized, in the sense that, even though a correspondence exists between A's and B's, it is unknown due to asynchronous timing of the measurement transmissions. Additionally, the data may contain gaps due to differing sampling rates, environmental conditions or dropped sensor readings.

In this paper we present two simple algorithms for selecting corresponding (A, B) pairs from data, with unknown or missing correspondences, that use properties of four screw-theory-based invariants. The proposed methods are valuable to any incarnation of the $AX = XB$ problem, but we motivate our study by problems in ultrasound sensor calibration, and illustrate the efficacy of our approach on simulated data.

The remainder of this section is devoted to presenting the motivating application, reviewing the literature, and establishing notation. Section II-A presents an algorithm using two of the well known screw theory invariants that corrects for simple uniform shifts. Section II-B presents an algorithm using all four of the invariants that corrects for nonuniform shifts and gaps.

A. Ultrasound: A Motivating Problem

$AX = XB$ is one of the most common mathematical formulations used in robot-sensor calibration problems. It can be found in a variety of applications including camera calibration, Cartesian robot hand calibration, robot eye-to-hand calibration [1], aerial vehicle sensor calibration [2] and image guided therapy (IGT) sensor calibration [3]. While the methods presented in this paper can be applicable to all of these fields, we choose to use the growing field of ultrasound (US) calibration, an essential component of IGT, to motivate our discussion.

Image Guided Therapy systems are commonly used in modern surgical procedures including minimally invasive surgery and robotic surgery. An IGT system could have several integrated components such as an imaging modality system, a pose tracker, and a surgical robot. The imaging modality component may be responsible for generating diagnostic images for image guidance and the pose tracker (which can be Electromagnetic (EM), optical or robotic) may be responsible for generating the position and orientation information of surgical tools or devices. A surgical robot could act as a data source if it reports position and orientation information of its surgical arms or attached devices. It could also act as a data sink if a control signal generated by the rest of the system were used to actuate the surgical robot itself.

Out of the most popular imaging modalities, such as X-ray, Computed Tomography (CT), or Magnetic Resonance Imaging (MRI), Ultrasound has several notable advantages: mobility, non-ionizing radiation, ease of use, low cost, real-time data acquisition. Due to these qualities, US is frequently integrated with tracking systems and robotic systems for Image Guided Therapy [4]-[7]. To compensate for the lower image quality and

This work was supported by NSF Grant RI-Medium: 1162095

[1]M.K. Ackerman and G.S. Chirikjian are with the Department of Mechanical Engineering and Laboratory for Computational Sensing and Robotics, Johns Hopkins University, Baltimore, MD, USA, gregc at jhu.edu

[2]A. Cheng and E.M. Boctor are with the Department of Computer Science and Radiology, and Laboratory for Computational Sensing and Robotics, Johns Hopkins University, Baltimore, MD, USA, eboctor1 at jhmi.edu

[3]B. Shiffman is with the Department of Mathematics, Johns Hopkins University, Baltimore, MD, USA. His work is supported by NSF grant DMS-1201372

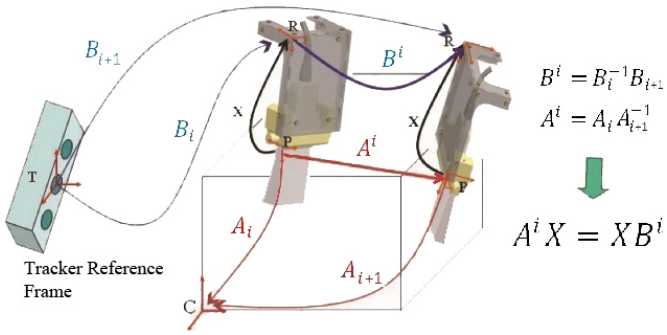

Fig. 1. Defining Reference Frames for the $AX = XB$ Problem in Ultrasound. Adapted, with permission, from Fig. 7.1 in [3].

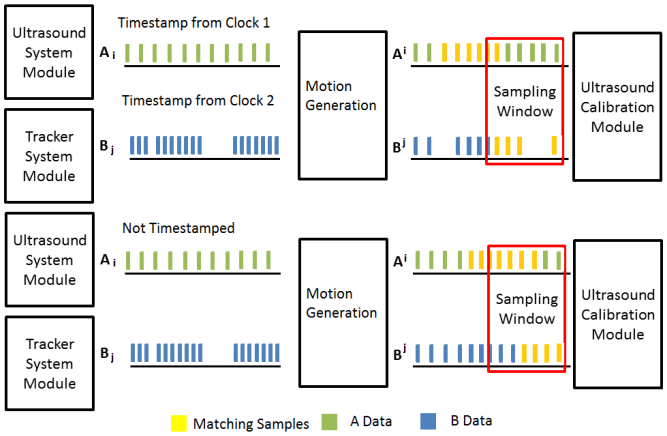

Fig. 2. Flow of Data Streams between Multiple Modules

resolution, pre-operative models in the other imaging modalities are often also integrated [7].

In the scenario where an US transducer has an attached optical marker or EM tracker, a common problem that needs to be solved is US calibration. This process refers to the recovery of the transformation between the optical marker or EM tracker to the US image space. Many researchers have presented techniques in computing this transformation [8]-[10]. As seen in Fig. 1, this calibration problem can be generalized as the $AX = XB$ formulation. Most conventional methods [11]-[13] can only solve this problem when the data streams of A and B contain known correspondences, meaning that a particular pair, (A_i, B_j), must correspond to a single motion. In the paper, we make the differentiation that A_i denotes the absolute pose of the image and A^i denotes the motion between image poses A_i and A_{i+1}, i.e. $A^i = A_i A_{i+1}^{-1}$. Similarly B_i is the absolute probe pose, in reference to the EM tracker, and B^i represents the motion between poses B_i and B_{i+1}, i.e. $B^i = B_i^{-1} B_{i+1}$.

In practice, ultrasound and tracking data are generated with different internal delays corresponding to US image formation delay and pose calculation delay respectively. The delays can cause these data streams to arrive at a module in an asynchronous fashion causing a shift between the two streams of data. Additionally, the tracking data and ultrasound data may have different frame rates that result in gaps or the need for interpolation. Here, we define gaps to be instances in a data stream where A_i does not have a corresponding B_j or vice versa. Gaps can be small and dispersed or large and connected.

B. Why Temporal Registration?

Since there are many data sources in an IGT system, their respective data streams will likely be unsynchronized. While these data streams can be used individually, they can provide much more information when combined. For example, an IGT system may have an US transducer with attached optical marker or electromagnetic tracker. These individual data streams can provide the user with a stream of medical images and also a stream of US transducer poses and orientations. However, if the data streams were temporally registered, we could know the pose and orientation of each individual US image. This may allow for registration of the US images together to create a 3D volume from a 2D transducer. Another application enabled by this synchronization is tracked US elastography [14]. The pose data from each US image allows the algorithm to select pairs of US images that generate the best elastography image.

C. Past Temporal Registration Methods

Fig. 2 is an example of what the data streams may look like either between processes on the same workstation or between workstations over a distributed network. The data streams labeled A_i and B_i represent the US and the tracker data streams respectively. It is shown here that there are gaps in the data and that the two data streams have different frame rates. Both of these data streams are sent into a motion generation module where consecutive poses are processed to form a relative motion. If timestamps are present, they can be used such that relative motions are only generated from consecutive poses and not from the poses surrounding a gap. The result of the motion generation is therefore a stream $\{A^i\}$ and $\{B^i\}$ with gaps. If timestamps are absent, then gaps are unknown at this stage and relative motions are generated for all consecutive poses. Both of the data streams in these two cases are being sent to an US calibration module which has a sampling window as shown in Fig. 2. The data colored in a lighter shade represents data that directly corresponds. It is often the case where one piece of corresponding data is not entirely sampled.

A common method for IGT systems to temporally register multiple data streams is the use of timestamps. [15]. In Fig. 2 if all of the modules were on a single workstation, this workstation would be responsible for collecting all of the data streams and can apply a

synchronized timestamp to each data packet as it is collected. With this method, even if the data streams are arriving in a staggered fashion at a certain module due to different processing times, the module will know the time of collection and will be able to temporally register the data streams together.

IGT systems are also commonly deployed as a network distributed system. This would be the case where the modules shown in Fig. 2 are executed on separate workstations. In this setup, the timestamp method becomes more complicated to implement as each of the workstations have asynchronous clocks. There are several methods to ensure that the workstations are synchronized. First, external hardware could be used as a master controller and generate a synchronous clock for each of the workstations in the network distributed system. There are also protocols such as the Network Time Protocol (NTP) [16] which can be used to synchronize multiple workstations on a network. Another possibility is to use a workstation as a master that sends out a series of short messages to each of the other workstations. Each individual workstation then uses these messages to compute a network delay that is used for the remainder of a particular execution.

D. Euclidean-Group Invariants: Necessary and Sufficient Conditions for Unique Solution to the $AX = XB$ Problem

The problem of solving

$$A^i X = X B^i \tag{1}$$

for X when multiple corresponding pairs of A's and B's are presented has been examined in the context of many diverse applications over the past quarter century [11]-[13], [17]-[24].

From screw theory it is known that any homogeneous transformation can be written as [25]

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

where $e^{\theta N}$ denotes the matrix exponential, $\mathbb{I}_n$ is the $n \times n$ identity matrix, and $\theta \in [0, \pi]$ is the angle of rotation.

$$N = \begin{pmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{pmatrix}$$

where $\mathbf{n} = [\mathbf{n_1}, \mathbf{n_2}, \mathbf{n_3}]^\mathbf{T} \in \mathbb{R}^3$ is the unit vector describing the axis of rotation, which connects the origin and any point on the unit sphere, and $\mathbf{p} \cdot \mathbf{n} = 0$. Together, $\{\theta, d, \mathbf{n}, \mathbf{p}\}$ define the Plücker coordinates of the screw motion.

If we write (1) as

$$A^i = X B^i X^{-1} \quad \text{where} \quad i \in \{1, 2\}, \tag{2}$$

then explicitly calculating and equating the matrix product gives two invariant relations,

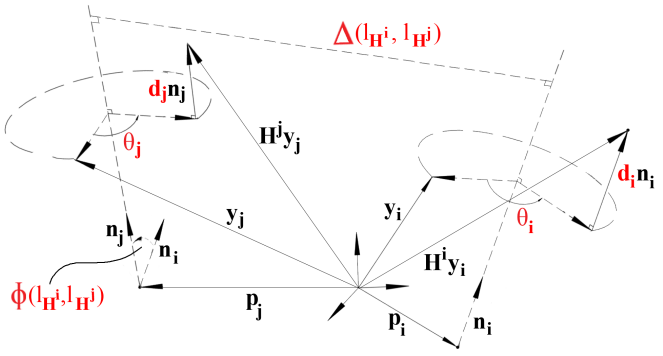

Fig. 3. Two arbitrary rigid-body motions, H^i which is shown acting on y_i and H^j which is shown acting on y_j, their Plücker coordinates and the parameters of the four described conditions (in red) [27]

$$\theta_{A^i} = \theta_{B^i} \quad d_{A^i} = d_{B^i} \tag{3}$$

where d_{A^i} and d_{B^i} are computed from A^i and B^i as in (2). Additionally, let

$$\mathbf{l}_{A^i}(t) = \mathbf{p}_{A^i} + t\mathbf{n}_{A^i} \quad \text{and} \quad \mathbf{l}_{B^i}(t) = \mathbf{p}_{B^i} + t\mathbf{n}_{B^i}$$

be the directed screw axis lines of A_i and B_i in three-dimensional Euclidean space. If the lines are not parallel or anti-parallel, i.e., if $\mathbf{n}_{A_i} \neq \pm\mathbf{n}_{B_i}$, then the distance between the two lines is given by

$$\Delta(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) = \frac{|[\mathbf{n}_{A^{i_1}}, \mathbf{n}_{A^{i_2}}, \mathbf{p}_{A^{i_2}} - \mathbf{p}_{A^{i_1}}]|}{\|\mathbf{n}_{A^{i_1}} \times \mathbf{n}_{A^{i_2}}\|} \tag{4}$$

where for any $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^3$, the triple product is $[\mathbf{a}, \mathbf{b}, \mathbf{c}] \doteq \mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$. In the current context we can think of $i_1 = 1$ and $i_2 = 2$ but in later discussion i_1 and i_2 can represent more general values.

If in addition, $\Delta(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) \neq 0$, i.e., if the lines are skew, then the angle $\phi(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) \in [0, 2\pi)$ is uniquely specified by

$$\cos\phi(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) = \mathbf{n}_{A^{i_1}} \cdot \mathbf{n}_{A^{i_2}} \tag{5}$$

$$\sin\phi(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) = \Delta(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}})^{-1}[\mathbf{n}_{A^{i_1}}, \mathbf{n}_{A^{i_2}}, \mathbf{p}_{A^{i_2}} - \mathbf{p}_{A^{i_1}}].$$

Therefore, if $\theta_{A^{i_1}}, \theta_{A^{i_2}} \in (0, \pi)$ and $\phi(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) \notin \{0, \pi\}$, then a unique solution of (1) exists if and only if the following four conditions hold:

1) $\theta_{A^{i_1}} = \theta_{B^{i_1}}$ and $\theta_{A^{i_2}} = \theta_{B^{i_2}}$;
2) $d_{A^{i_1}} = d_{B^{i_1}}$ and $d_{A^{i_2}} = d_{B^{i_2}}$;
3) $\phi(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) = \phi(\mathbf{l}_{B^{i_1}}, \mathbf{l}_{B^{i_2}})$;
4) $\Delta(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) = \Delta(\mathbf{l}_{B^{i_1}}, \mathbf{l}_{B^{i_2}})$.

If these do not hold, then a solution will not be possible [26]. Fig. 3 illustrates the Plücker coordinates, and the parameters of the above four conditions for two arbitrary rigid-body motions [27]

II. FINDING X WITHOUT KNOWING A-B CORRESPONDENCE

In this section we pose two possible algorithmic solutions to the problem of solving for X using incomplete sensor data. The first method uses the first two, θ and d, invariants to compute a correlation. The second method uses all four invariants to extract corresponding (A_i, B_i) pairs from the data.

A. Using θ, d and Correlation

The first method presented leverages the θ and d invariants to "re-shift" temporally misaligned data. Given n, (A_i, B_i) pairs drawn from simulated sensor data, we shift the set of A's by a set amount to mimic the effects of the asynchronous data transfer. The SE(3) invariants are then extracted from each of the A_i's and B_i's in the new temporally shifted set. We can then perform a correlation of the A invariants (θ_A, d_A) with the B invariants (θ_B, d_B) using the Discrete Fourier Transform (DFT).

Given a sequence of N complex numbers, the DFT is defined as $F_n = \mathcal{F}[\{f_k\}_{k=0}^{N-1}](n)$ where

$$F_n \doteq \sum_{k=0}^{N-1} f_k e^{-2\pi i n \frac{k}{N}}$$

and the inverse transform is given as:

$$f_k \doteq \frac{1}{N} \sum_{n=0}^{N-1} F_n e^{2\pi i k \frac{n}{N}}$$

The convolution theorem for the Discrete Fourier Transform indicates that a correlation, $Corr(f, g)$, of two sequences of finite length can be obtained as the inverse transform of the product of one individual transform with the complex conjugate (*) of the other transform:

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

The location of largest correlation corresponds with the amount of shift in the A's. Fig. 4 shows an example case where the data streams are shifted by -13 units. The shifted theta streams can be seen in the top graph and the correlation plot, where the maximum value is at the predicted location (-13), is shown in the bottom graph. This method accurately recovers the shift in the data streams and then corrects the shift to calculate for X.

Though this approach is the most accurate with uniform shifts, it is also robust to other forms of data stream inconsistencies. If there are gaps in the data and either the gaps are large enough to distinguish (based on large changes in the invariant curves), or the data is time-stamped, we can create substreams of data between gaps. The algorithm can then predict shifts of individual stream subsets to give corresponding pairs of A's and B's. However, the algorithm begins to break down if there are largely varying, non-uniform shifts,

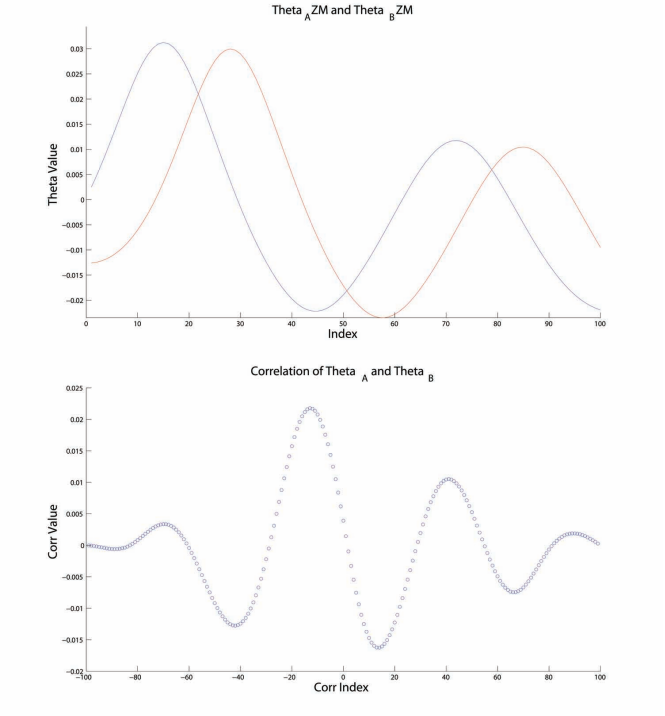

Fig. 4. Shifted Data Streams for the Theta Invariant and Their Correlation

or a large number of small gaps in the data. As an alternate option, we therefore present the following algorithm that uses all four invariants.

B. Using All Four Invariants

Suppose that data streams of sensor measurements $\mathcal{A} = \{A^i\}$ and $\mathcal{B} = \{B^j\}$ are presented and there are both significant unknown temporal shifts between these two sets, and gaps within each one. The number of points in these sets are $|\mathcal{A}| = m$ and $|\mathcal{B}| = n$.

Here we present an approach to recovering X and establishing a correspondence between the subsets $\mathcal{A}' \subset \mathcal{A}$ and $\mathcal{B}' \subset \mathcal{B}$ that do correspond where $|\mathcal{A}'| = |\mathcal{B}'| = p \leq \min(m, n)$. For such data, we find the correspondence, which is a permutation on p letters, $\pi \in \Pi_p$, such that $A^i X = X B^{\pi(i)}$ for $i = 1, ..., p$ where $A^i \in \mathcal{A}'$ and $B^{\pi(i)} \in \mathcal{B}'$.

We accomplish this using the invariants of the Special Euclidean group, $SE(3)$, under conjugation. The procedure is as follows. Compute (θ^{A^i}, d^{A^i}) for each $A^i \in \mathcal{A}$ and (θ_{B^j}, d_{B^j}) for each $B^j \in \mathcal{B}$. Next, form a 2D grid on the θ-d plane that ranges from $\min_{i,j}(\theta_{A^i}, \theta_{B^j})$ to $\max_{i,j}(\theta_{A^i}, \theta_{B^j})$ and $\min_{i,j}(d_{A^i}, d_{B^j})$ to $\max_{i,j}(d_{A^i}, d_{B^j})$. This grid will give r rectangles, e.g., if it is a 10×10 grid, then $r = 100$. Assuming that no data falls exactly on a grid line, this will partition $\mathcal{A}$ and $\mathcal{B}$ into r disjoint subsets: $\{\mathcal{A}_1, \mathcal{A}_2, ..., \mathcal{A}_r\}$ and $\{\mathcal{B}_1, \mathcal{B}_2, ..., \mathcal{B}_r\}$ where

$$\mathcal{A}_{i_1} \cap \mathcal{A}_{i_2} = \emptyset \quad \text{and} \quad \bigcup_{i=1}^{r} \mathcal{A}_i = \mathcal{A},$$

and similarly for $\mathcal{B}$.

The reason for doing this is that all candidate A's and B's that can potentially match will be in corresponding partitions $\mathcal{A}_i$ and $\mathcal{B}_i$, since having the same value of θ and d is a necessary condition for a solution to $AX = XB$ to exist. Constructing the grid with finite resolution allows for the possibility of some measurement error in A's and B's.

Let $|\mathcal{A}_i| = m_i$ and $|\mathcal{B}_j| = n_j$. Then

$$\sum_{i=1}^{r} m_i = m \quad \text{and} \quad \sum_{i=1}^{r} n_j = n.$$

Pick two bins for which all of the numbers in the pairs (m_{i_1}, n_{j_1}) and (m_{i_2}, n_{j_2}) are small, but greater than 2, to allow for the fact that measurement error may result in incorrect binning, and also that the angle $\phi(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}})$ might not always be in the range $(0, \pi)$. We interrogate all $m_{i_1} \times n_{j_1} \times m_{i_2} \times n_{j_2}$ possibilities as candidates. The further necessary conditions for the existence of a solution are $\phi(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) = \phi(\mathbf{l}_{B^{j_1}}, \mathbf{l}_{B^{j_2}})$ and $\Delta(\mathbf{l}_{A^{i_1}}, \mathbf{l}_{A^{i_2}}) = \Delta(\mathbf{l}_{B^{j_1}}, \mathbf{l}_{B^{j_2}})$. From among all pairs that satisfy these conditions, we can use existing $AX = XB$ solvers (such as the Kronecker-product-solver used in [3],[28]) to determine X.

C. Solving $AX = XB$ Using Kronecker Products

To solve our, now registered, $AX = XB$ problem, we use a common method that finds a least-squares solution. The core of this approach is to use the Kronecker product. Recall that If C is a matrix, $\mathrm{vec}(C)$ is the long vector produced by stacking the columns of C. This is a linear operation in the sense that

$$\mathrm{vec}(\alpha_1 C_1 + \alpha_2 C_2) = \alpha_1 \mathrm{vec}(C_1) + \alpha_2 \mathrm{vec}(C_2).$$

Moreover, if $\otimes$ denotes the Kronecker product, and C, D, E are matrices with dimensions compatible for multiplication, then

$$\mathrm{vec}(CDE) = (E^T \otimes C)\,\mathrm{vec}(D).$$

If D is already a column vector ($n \times 1$ matrix), then it is unaltered by the $\mathrm{vec}(\cdot)$, and if D is a row vector ($1 \times n$ matrix) then $\mathrm{vec}(\cdot)$ transposes it. If C is a matrix and α is a scalar, then

$$\alpha \otimes C = C \otimes \alpha = \alpha C,$$

the scalar multiple of C by α.

This means that we can write the $AX = XB$ equation as [3]

$$J_i \mathbf{x} = \mathbf{b_i} \tag{6}$$

where

$$J_i = \begin{bmatrix} \mathbb{I}_9 - R_{B_i} \otimes R_{A_i} & \mathbb{O}_{9\times 3} \\ \mathbf{t}_{B_i}^T \otimes \mathbb{I}_3 & \mathbb{I}_3 - R_{A_i} \end{bmatrix} \tag{7}$$

$$\mathbf{x} = \begin{bmatrix} \mathrm{vec}({}^K R_X) \\ {}^K\mathbf{t}_X \end{bmatrix} \text{ and } \mathbf{b_i} = \begin{bmatrix} \mathbf{0}_9 \\ \mathbf{t}_{A_i} \end{bmatrix} \tag{8}$$

SHIFT \ GAPS	0%	10%	20%	30%	40%	50%	60%	70%	80%
0%	100	100	100	100	100	98	58	13	2
10%	100	100	100	100	100	94	48	11	0
20%	100	100	100	100	99	80	40	4	0
30%	100	100	98	100	92	64	18	1	1
40%	100	100	100	99	91	53	20	1	0
50%	100	100	100	99	78	46	9	1	1
60%	100	100	99	96	62	23	1	0	0
70%	100	99	96	74	22	0	0	0	0
80%	100	99	65	4	0	0	0	0	0

*Italicized data is percentage of trials that correctly solve for X, given the indicated shift and gaps (100 total trials).

Fig. 5. Algorithm 2 Success Rate for Varying Amounts of Shift and Number of Small Gaps

$\mathbb{I}_m$ is the $m \times m$ identity, $\mathbb{O}_{m\times n}$ is the $m \times n$ zero matrix, and $\mathbf{0}_n$ is the n-dimensional zero vector.

J_i is a 12×6 matrix and $\mathbf{b}_i$ is six-dimensional. By stacking multiple such equations for different pairs (A_i, B_i), we obtain $J\mathbf{x} = \mathbf{b}$ where J is $12n \times 6n$ and $\mathbf{b}$ is $6n$-dimensional. The least-squares solution for $\mathbf{x}$ can then be found using SVD methods or using a pseudo-inverse.

For example, the least-squares solution to $\|J\mathbf{x} - \mathbf{b}\|_M$ where $M = M^T \in \mathbb{R}^{6n\times 6n}$ is

$$\mathbf{x} = (J^T M J)^{-1} J^T M \mathbf{b}. \tag{9}$$

This is the over-constrained pseudo-inverse (as opposed to the under-constrained one typically used in redundancy resolution).

Finally, the Kronecker product solution does not guarantee that ${}^K R_X$ in (8) is in SO(3). However there are procedures for projecting ${}^K R_X$ back into the group SO(3) to result in R_X. For a more in-depth discussion of these methods and proofs of their veracity see [3],[28].

III. RESULTS

A. Simulated Sensor Data

Fig. (5) shows the success of this algorithm with different amounts of shift and gaps in the data. The percentages of shifts correspond with the percentage of data that does not overlap between the two data streams. The percentage of gaps correspond to the percentage of data that is missing from either of the two shifted data streams. Clearly the algorithm is highly robust to unknown and missing correspondences of all kinds.

B. Effectiveness of the Proposed Algorithms for $SO(3)$

For some applications of the $AX = XB$ problem, the sensor data is only rotational (such as IMU data in [2]). The (A_i, B_j) pairs are now drawn from the group of rigid-body rotations, $SO(3)$, a subgroup of $SE(3)$. In this case there is no d or Δ. The presented algorithms are still successful for data of this type, despite the absence of translation information. For the

algorithm using correlations we are still able to use θ to successfully match the data streams. The second algorithm, which uses the binning procedure, is also successful, using only the θ and ϕ invariants.

IV. CONCLUSIONS AND FUTURE WORK

We establish that the $AX = XB$ problem can make use of the invariants inherent in the structure of the A's and B's to correct for unknown and missing correspondences in the sensor data streams. These invariants are used in two algorithms, the first of which can realign uniformly asynchronous data and, in some cases, data with gaps. The second algorithm solves for X for most instances of shifts and gaps in the data streams. The problem is motivated by an ultrasound calibration problem, though the results will be applicable to many scenarios in which the $AX = XB$ problem arises.

For the $AX = XB$ problem, it is also recognized that information obtained from the sensors will have errors other than unknown correspondences. Each sensor reading will contain uncertainty about the quantity measured, and therefore, even if correspondences can be determined, the data may still contain noise on each of the motions (A_i, B_i). Our future goal is to leverage these algorithms when this form of noise is present, to achieve a reliable estimation of X.

REFERENCES

[1] Tsai, R., Lenz, R., "A New Technique for Fully Autonomous and Efficient 3D Robotics Hand/Eye Calibration, " *IEEE Transactions on Robotics and Automation*, VOL5.3, June 1989

[2] Mair, E., Fleps, M., Suppa, M., Burschka, D., "Spatio-temporal initialization for IMU to camera registration," *2011 IEEE ROBIO*, pp. 557 - 564, Dec. 2011.

[3] Boctor, E.M., "Enabling Technologies For Ultrasound Imaging In Computer-Assisted Intervention," 2006, Computer Science Department, Johns Hopkins University.Thesis.

[4] Boctor, E.M., Stolka, P., Kang, H. J., Clarke, C., Rucker, C., Croom, J., Burdette, E. C., Webster, R. J., III, "Precisely shaped acoustic ablation of tumors utilizing steerable needle and 3D ultrasound image guidance," *SPIE Medical Imaging 2010*, 2010.

[5] Boisvert, J., Gobbi, D., Vikal, S., Rohling, R., Fichtinger, G., Abolmaesumi, P., "An open-source solution for interactive acquisition, processing and transfer of interventional ultrasound images," *MICCAI 2008, International Workshop on System and Architectures for Computer Assisted Interventions*, 2008.

[6] Billings, S., Deshmukh, N., Jae Kang, H., Taylor, R., Boctor, E.M., "System for robot-assisted real-time laparoscopic ultrasound elastography." *Proc. SPIE 8316, Medical Imaging 2012: Image-Guided Procedures, Robotic Interventions, and Modeling*, 83161W (February 23, 2012); doi:10.1117/12.911086.

[7] Foroughi, P., Csoma, C., Rivaz, H., Fichtinger, G., Zellars, R., Hager, G., Boctor, E.M., "Multi-modality fusion of CT, 3D ultrasound, and tracked strain images for breast irradiation planning," *SPIE Medical Imaging 2009*, 2009, p. 72651B.

[8] Boctor, E.M., Viswanathan, A., Choti, M.A., Taylor, R.H., Fichtinger, G., Hager, G.D., "A Novel Closed Form Solution for Ultrasound Calibration, *IEEE Int Symp. On Biomedical Imaging*, 2004, pp. 527-530.

[9] Mercier, L., Langø, T., Lindseth, F., Collins, D.L., "A review of calibration techniques for freehand 3-D ultrasound systems, *Ultrasound Med Biol*, 2005 Feb, vol 31(2), pp. 143-165.

[10] Poon, T., Rohling, R., "Comparison of calibration methods for spatial tracking of a 3-D ultrasound probe." *Ultrasound in Medicine and Biology* 31(8), 10951108 (2005)

[11] Park, F.C., Martin, B.J., "Robot Sensor Calibration: Solving $AX = XB$ on the Euclidean Group," *IEEE Trans. Robotics and Automation*, Vol. 10, No. 5, pp. 717-721, Oct. 1994.

[12] Daniilidis, K., "Hand-Eye Calibration Using Dual Quaternions," *The International Journal of Robotics Research* Vol 18, pp. 286-298, 1999.

[13] Fassi, I., Legnani, G., "Hand to Sensor Calibration: A Geometrical Interpretation of the Matrix Equation AX=XB," Journal of Robotic Systems 22(9), pp. 497-506, 2005.

[14] Foroughi, P., Rivaz, H., Fleming, I. N., Hager, G. D., Boctor, E. M. ,"Tracked Ultrasound Elastography (TrUE), *International Conference on Medical Image Computing and Computer-Assisted Intervention MICCAI 2010*, Proceedings in LNCS Vol. 6362, pp 9-16, Springer, 2010.

[15] Kang, H. J., Cheng, A., Boctor, E. M., "MUSiiC ToolKit 2.0: Bidirectional Real-time Software Framework for Advanced Interventional Ultrasound Research.," *MICCAI 2012, International Workshop on System and Architectures for Computer Assisted Interventions*, 2012.

[16] Mills, D. L., "Internet Time Synchronization: Network Time Protocol. *IEEE Transactions on Communications* 39(10), 1482-1493 (1991)

[17] Arun, K.S., Huang, T.S., Blostein, S.D., "Least-Squares Fitting of Two 3-D Point Sets," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 9, No. 5, pp. 698-700, Sept. 1987.

[18] Chou, J.C.K., Kamel, M., "Finding the Position and Orientation of a Sensor on a Robot Manipulator Using Quaternions," *The International Journal of Robotics Research*, Vol. 10, No. 3, pp. 240-254, June 1991.

[19] Shiu, Y.C., Ahmad, S., "Calibration of Wrist-Mounted Robotic Sensors by Solving Homogeneous Transform Equations of the Form $AX = XB$," *IEEE Trans. Robotics and Automation*, Vol. 5, No. 1, pp. 16-29, Feb. 1989.

[20] Bai, S., Teo, M.Y., "Kinematic Calibration and Pose Measurement of a Medical Parallel Manipulator by Optical Position Sensors," *Journal of Robotic Systems* 20(4), pp. 201-209, 2003.

[21] Agrawal, M., "A Lie Algebraic Approach for Consistent Pose Registration for General Euclidean Motion," *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems October 9 - 15, 2006, Beijing, China* pp. 1891-1897.

[22] Jordt, A., Siebel, N., Sommer, G., "Automatic High-Precision Self-Calibration of Camera-Robot Systems," *2009 IEEE International Conference on Robotics and Automation Kobe International Conference Center* pp. 1244-1249, May 2009.

[23] Kim, S., Jeong, M., Lee, J., Lee, J., Kim, K., You, B., Oh, S., "Robot Head-Eye Calibration Using the Minimum Variance Method," *Proceedings of the 2010 IEEE International Conference on Robotics and Biomimetics* pp. 1446 - 1451., Dec. 2010

[24] Dai, Y., Trumpf, J., Li, H., Barnes, N., Hartley, R., "Rotation Averaging with Application to Camera-Rig Calibration," H. Zha, R.-i. Taniguchi, and S. Maybank (Eds.): *ACCV 2009*, Part II, LNCS 5995, pp. 335-346, 2010.

[25] Chen, H.H., "A Screw Motion Approach to Uniqueness Analysis of Head-Eye Geometry" *IEEE Conference on Computer Vision and Pattern Recognition, 1991*, pp. 145-151, 1991.

[26] Chirikjian, G.S., Shiffman, B., "Mathematical Aspects of Molecular Replacement: Packing Rigid Bodies Subject to Crystallographic Symmetry Constraints," (in preparation).

[27] Chirikjian, G.S., Kyatkin, A.B., *Engineering Applications of Noncommutative Harmonic Analysis*, CRC Press, Boca Raton, FL 2001.

[28] Andreff, N., Horaud, R., Espiau, B., 'Robot Hand-Eye Calibration Using Structure-from-Motion" *The International Journal of Robotics Research*, 2001.