Performance Improvement of Measurement Association Using a System with two 2D Sensors and one 3D Sensor

CÉSAR CONTRERAS
JOHN LANGFORD
LARRY AMMANN
JOHN ZWECK
BRIAN MARKS

A measurement-to-measurement data association problem is formulated for a target tracking system consisting of one or two 2D sensors and a 3D sensor. Operating conditions are identified under which performance is improved by using two 2D sensors and a 3D sensor instead of one 2D sensor and a 3D sensor. To facilitate this study, two algorithms are introduced to compute near-optimal solutions of the corresponding three-way assignment problem: a single-step algorithm based on two independent two-way assignment problems, and a related iterative algorithm that explicitly enforces a compatibility condition between measurements made by the 2D sensors. Simulation studies show that the position estimates obtained with the three-sensor system are much more accurate than those obtained with a two-sensor system whenever there is large uncertainty in the 3D sensor in the dimension orthogonal to the plane of the 2D sensor in the two-sensor system. Moreover, whenever there is large uncertainty in the measurements from the 3D sensor in the common dimension of the 2D sensors, the percentage of correct matches with the the iterative assignment algorithm for the three-sensor system is significantly better than that with a two-sensor system. The degree to which the methods and results can be extended to more realistic 3D radar and 2D camera models is discussed and inferences for aerospace and missile defense applications are drawn.

Manuscript received January 18, 2019; revised April 9, 2019; released for publication July 8, 2019.

C. Contreras, J. Langford, L. Ammann, and J. Zweck are with the Department of Mathematical Sciences, The University of Texas at Dallas, Richardson, TX 75080, USA.
B. Marks is with The Johns Hopkins University Applied Physics Laboratory, Laurel, MD 20723, USA.
This project was supported under the NSF Enriched Doctoral Training Program entitled "Team Training Mathematical Scientists Through Industrial Collaborations," DMS Award #1514808.

1557-6418/19/$17.00

I. INTRODUCTION

Multitarget, multisensor tracking systems for aerospace and missile defense applications are typically based on a network of radars [1], [2], [6], [11]. In computer vision and robotics, tracking systems are often based on a network consisting of cameras or radars or both. These systems are used to track the movement of people and robots, for perception systems in autonomous vehicles, and for surveillance applications [15], [18]–[20]. Therefore, an important problem is to associate and fuse tracks generated by networks of heterogeneous 2D and 3D sensors. The simplest such data association problem is the measurement-to-measurement association (M2MA) problem, which is that of associating measurements from different sensors to form composite measurements that can then be used by a centralized tracking system to generate a single set of tracks using data from several sensors [3], [4]. In this paper, we adopt the approach of [4] and [6] in which target states and measurements are represented as random vectors. However, recently a Bayesian inference approach to target tracking based on message passing and the sum–product algorithm has also been shown to be particularly effective [12].

We classify data association problems, such as the M2MA problem, by the number of sensors in the system and the dimensions of the Euclidean spaces of the data recorded by the sensors. For example, we will classify a system with one 3D radar and two 2D cameras as a 322-sensor system, and a system with one 3D radar and one 2D camera as a 32-sensor system.

Since target tracking with three sensors is significantly more complex than with two sensors, it is important to identify situations in which the addition of a third sensor results in more accurate estimates of the target positions. Deb *et al.* [7] studied the M2MA problem for 322-, 222-, and 32-sensor systems using target scenarios that simulated a squadron of fighter jets flying in formation. They found that ghosting and resolution problems due to specific geometric configurations of the targets were a major source of position errors, but that these were smaller for a 322-sensor than a 222-sensor system.

The main contribution of this paper is to identify sensor operating conditions under which the performance of a target tracking system can be improved by using a 322-sensor system rather than a 32-sensor system. Specifically, in the context of M2MA, we study how the percentage of correct assignments and the average target position error depend on the angle between the planes of the two 2D sensors and on the orientation of the covariance ellipsoid of the 3D sensor with respect to the camera planes. By contrast, the results of Deb *et al.* [7] were obtained with fixed sensor orientations. In Section VI, we show that the percentage of correct assignments with a 322-sensor system is significantly better than that with a 32-sensor system whenever the

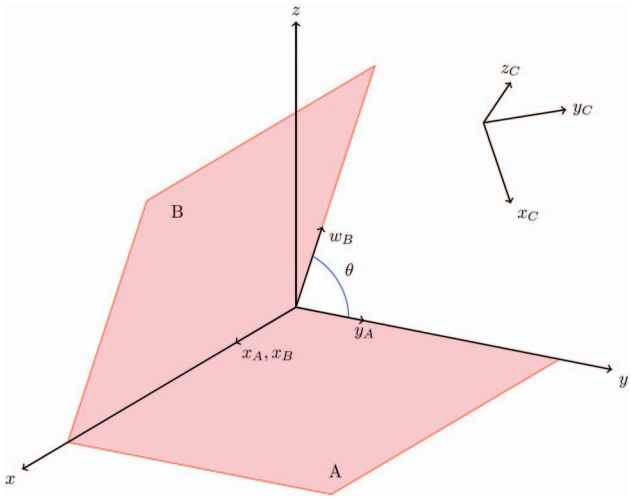


Fig. 1. The geometric configuration and coordinate systems for the three-sensor system.

uncertainty in the measurements made by the 3D sensor is large compared to that of the two 2D sensors along the line of intersection of the planes of the two 2D sensors (i.e., along the x-axis in Fig. 1). We also show that the root-mean-square error (RMSE) in the estimated target positions is much smaller for the 322-sensor system when the uncertainty in the 3D sensor measurements is large in the dimension orthogonal to the plane of the 2D sensor. In Section VII, we discuss the implications of these results for aerospace and missile defense applications. In particular, our results suggest that there can be a significant performance advantage to a target tracking system with two cameras and a 3D radar as opposed to one camera and a 3D radar.

The data association problems that arise in target tracking and sensor fusion can be formulated as multidimensional assignment problems [16]. The problem of computing the most likely assignment of measured sensor data to a collection of unknown targets involves optimizing a log-likelihood cost function subject to a collection of constraint equations that specify the set of feasible assignments. The cost function defined by Deb *et al.* [7] allows for realistic sensor models and includes spurious measurements and missed detections. They formulated the resulting three-way assignment problem as an integer linear program in which the coefficients of the cost function are defined in terms of unknown target positions. Since the number of coefficients is equal to the number of ways to select three measurements, one from each sensor, the computation of the coefficients involves solving a large number of nonlinear least-squares problems for the target positions. Indeed, Deb *et al.* found that up to 80% of the computational time of the assignment algorithm was taken up with solving these least-squares problems.

Although multidimensional assignment problems are NP-hard for $N > 3$ sensors, there are efficient iterative algorithms to compute suboptimal solutions based on greedy algorithms, simulated annealing, and Lagrangian relaxation-based methods [14], [17]. For example, in their work Deb *et al.* [7] employed a nearly optimal three-way assignment algorithm grounded in linear programming theory that trades off computational time for some loss of association accuracy. A key step in their method is to incorporate one of the constraint equations into the cost function via Lagrange multipliers and to solve the resulting two-way assignment problem using Bertsekas' auction algorithm [5] or Munkres' algorithm [13]. One advantage of this approach is that it provides upper and lower bounds for the cost of the optimal solution. In a simulation study, they found that the average gap between these bounds can be reduced to less than 2%. Such Lagrangian relaxation methods have been extended to N-way assignments [8], [17].

Since we did not have access to an implementation of a state-of-the-art assignment algorithm for the M2MA problem [8], [17], to compare the performance of 322- and 32-sensor systems we considered an idealized situation with simplified target-to-sensor transformations and with no spurious measurements or missed detections. With these simplifying assumptions, we were able to define a cost function whose coefficients do not depend on unknown target positions, thus avoiding the high computational cost of solving the nonlinear least-squares problems in [7], [8], and [17].

Furthermore, to obtain near-optimal three-way assignments we used the simplified geometric configuration of the sensors to devise two algorithms, each of which reduces the 322-sensor assignment problem to a sequence of 32-sensor problems that we solve using Munkres' algorithm. We will refer to these algorithms, which are described in Sections III and IV, as the single-step four-dimensional (4D) algorithm and the iterative five-dimensional (5D) algorithm. Although we do not prove that the iterative 5D algorithm converges to the optimal solution, we performed simulation studies (not described here) verifying that the method yields optimal assignments for small problems ($\approx$10 targets) by comparing to results obtained by an exhaustive search.

In Section VII, we will show that the simplified target-to-sensor transformations we used are valid approximations of realistic radar and camera models for applications in which the targets are located at a significant distance from the sensors and are confined to a sufficiently small region of the field of view. We will also argue that the main conclusions obtained in Section VI on the performance advantages of a 322-sensor system over a 32-sensor system should remain valid if a state-of-the-art method was used that incorporates spurious measurements and missed detections.

In Section II, we introduce the model we used for the 322-sensor association problem, and in Sections III and

IV we derive formulas for the cost function in the case of orthogonal and nonorthogonal 2D sensor planes, respectively. In Section V, we derive a formula for the estimated target positions. In Section VI, we describe the numerical simulations we performed to quantify the performance of the methods we developed. In Section VII, we discuss the extent to which our methods and results can be generalized to realistic 322-sensor systems. Finally, in Section VIII, we present our conclusions.

II. SYSTEM MODEL

In this section, we describe the system models we used to compare the performance of 322- and 32-sensor systems.

We consider a target tracking scenario with three sensors: two 2D sensors (A and B) and one 3D sensor (C). We assume that all three sensors make N measurements of the same N targets, and we assume that there is no systematic bias in any of the sensors. However, mismatches can occur because of the inherent uncertainty in the measurements made by each sensor. We modeled the 2D sensors using parallel projections rather than the more realistic perspective projections often used to model cameras. In addition, we assume that the 3D sensor uses the identity transformation to map each target to a measurement in a 3D rectangular coordinate system, rather than employing a more realistic radar model based on the transformation from XYZ to RUV space, as is described, for example, in [9]. In Section VII, we will explain why these simplified models are reasonable.

In Fig. 1, we show a schematic diagram of the geometric configuration of the three sensors. The planes of the two 2D sensors intersect in a line, L, and we let θ denote the angle between these two planes. We define a coordinate system, (x, y, z), adapted to Sensors A and B as follows. We choose the origin of the coordinate system to be a point on L, the x-axis to be along the line L, the y-axis to be perpendicular to the x-axis and in the plane of Sensor A, and the z-axis to be orthogonal to the xy-plane.

We generate the N targets in the (x, y, z) coordinate system, and suppose that each sensor records the measured positions of each target in a coordinate system adapted to that sensor. Without any loss of generality, we choose the coordinate systems (x_A, y_A) adapted to Sensor A, (x_B, w_B) adapted to Sensor B, and (x_C, y_C, z_C) adapted to Sensor C such that

$$\begin{aligned}\begin{bmatrix} x_A & y_A \end{bmatrix} &= \begin{bmatrix} x & y \end{bmatrix} \\ &= \begin{bmatrix} x & y & z \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} x & y & z \end{bmatrix} M_A \end{aligned} \tag{1}$$

$$\begin{aligned}\begin{bmatrix} x_B & w_B \end{bmatrix} &= \begin{bmatrix} x & \cos(\theta)y + \sin(\theta)z \end{bmatrix} \\ &= \begin{bmatrix} x & y & z \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \cos\theta \\ 0 & \sin\theta \end{bmatrix} \\ &= \begin{bmatrix} x & y & z \end{bmatrix} M_B, \end{aligned} \tag{2}$$

and

$$\begin{bmatrix} x_C & y_C & z_C \end{bmatrix} = \begin{bmatrix} x & y & z \end{bmatrix} R, \tag{3}$$

for some 3×3 orientation-preserving orthogonal matrix R. The projection matrices, M_A and M_B, in (1) and (2) map the 3D vector $\begin{bmatrix} x & y & z \end{bmatrix}$ to the planes of Sensors A and B.

We assume that the errors in the positions measured by Sensors A, B, and C are independent Gaussian random vectors with covariance matrices Σ_A, Σ_B, and $\widetilde{\Sigma}_C$, where Σ_A and Σ_B are 2×2 matrices and $\widetilde{\Sigma}_C$ is a 3×3 matrix. We assume that these variances are known from sensor calibration measurements, and are given in the coordinate system adapted to each sensor. For simplicity, henceforth we suppose that the measurements made by Sensor C have been transformed from (x_C, y_C, z_C) to (x, y, z) coordinates via (3) and that the covariance matrix $\widetilde{\Sigma}_C$ has been transformed to $\Sigma_C = R\widetilde{\Sigma}_C R^T$.

The notational conventions we use are summarized in Table I. In addition, we use the following notation for submatrices of an $N \times M$ matrix A. Let $\mathbf{i} = (i_1, \ldots, i_K)$ and $\mathbf{j} = (j_1, \ldots, j_L)$ be vectors of row and column indices of A, respectively. Then, let $A[\mathbf{i}, \mathbf{j}]$ denote the $K \times L$ submatrix of A consisting of the rows and columns indexed by $\mathbf{i}$ and $\mathbf{j}$, respectively. Also, let $A[*, \mathbf{j}] = A[(1, \ldots, N), \mathbf{j}]$ and $A[\mathbf{i}, *] = A[\mathbf{i}, (1, \ldots, M)]$.

Let $\begin{bmatrix} X & Y & Z \end{bmatrix}$ denote the $N \times 3$ matrix of the true positions of the N targets in the coordinate system (x, y, z) adapted to Sensors A and B. Let A_{err} and B_{err} denote the $N \times 2$ matrices of position errors for Sensors A and B, respectively, and let C_{err} denote the $N \times 3$ matrix of position errors for Sensor C. The rows of A_{err} are independent and identically distributed (i.i.d.) multivariate normal distributions with mean zero and covariance matrix Σ_A, $A_{\text{err}}[n, *] \sim MN(0, \Sigma_A)$, and similarly for Sensors B and C. Then, the positions of the N targets as measured by Sensors A, B, and C are given by

$$\begin{bmatrix} \widehat{X}_A & \widehat{Y}_A \end{bmatrix} = \begin{bmatrix} X_A & Y_A \end{bmatrix} + A_{\text{err}} \in \mathbf{R}^{N \times 2} \tag{4}$$

$$\begin{bmatrix} \widehat{X}_B & \widehat{W}_B \end{bmatrix} = \begin{bmatrix} X_B & W_B \end{bmatrix} + B_{\text{err}} \in \mathbf{R}^{N \times 2} \tag{5}$$

$$\begin{bmatrix} \widehat{X}_C & \widehat{Y}_C & \widehat{Z}_C \end{bmatrix} = \begin{bmatrix} X_C & Y_C & Z_C \end{bmatrix} + C_{\text{err}} \in \mathbf{R}^{N \times 3} \tag{6}$$

respectively. The matrix $\begin{bmatrix} X_A & Y_A \end{bmatrix}$ in (4) is obtained via an unknown permutation, π_A, of the rows of the matrix $\begin{bmatrix} X & Y & Z \end{bmatrix}$ using the projection defined in (1). Similarly, the matrix $\begin{bmatrix} X_B & W_B \end{bmatrix}$ in (5) is obtained via an unknown permutation, π_B, of the rows of the matrix $\begin{bmatrix} X & Y & Z \end{bmatrix}$ using the projection defined in (2). These permutations

TABLE I
Notational Conventions Used in This Paper

Variable	Meaning
$(x_A, y_A), (x_B, w_B), (x_C, y_C, z_C)$	Coordinate systems adapted to Sensors A, B, and C
$[X, Y, Z]$	$N \times 3$ matrix of true target positions
$[X_A, Y_A], [X_B, W_B], [X_C, Y_C, Z_C]$	$N \times K$ matrices of permuted true target positions for Sensors A, B ($K = 2$), and C ($K = 3$)
$\Sigma_A, \Sigma_B, \Sigma_C$	$K \times K$ covariance matrices for Sensors A, B, and C
$A_{\text{err}}, B_{\text{err}}, C_{\text{err}}$	$N \times K$ matrices of position errors for Sensors A, B, and C
$[\widehat{X}_A, \widehat{Y}_A], [\widehat{X}_B, \widehat{W}_B], [\widehat{X}_C, \widehat{Y}_C, \widehat{Z}_C]$	$N \times K$ matrices of permuted measured target positions for Sensors A, B, and C

model the fact that the three sensors do not record the data from the N targets in the same order.

Let S_N denote the set of all permutations of N symbols. With each permutation, $\pi \in S_N$, we can associate an $N \times N$ matrix, also denoted by π, with the property that left multiplication of a matrix by π permutes the rows of that matrix. Specifically, the permutation matrix corresponding to the permutation, $\pi(1, 2, \ldots, N) = (\pi(1), \pi(2), \ldots, \pi(N))$, is given by

$$\pi = \begin{bmatrix} \mathbf{e}_{\pi(1)} \\ \mathbf{e}_{\pi(2)} \\ \vdots \\ \mathbf{e}_{\pi(n)} \end{bmatrix} \tag{7}$$

where the row vector $\mathbf{e}_j$ is the jth standard basis vector. Therefore, by (4) and (5),

$$\begin{bmatrix} \widehat{X}_A & \widehat{Y}_A \end{bmatrix} = \pi_A \begin{bmatrix} X & Y \end{bmatrix} + A_{\text{err}} \tag{8}$$

$$\begin{bmatrix} \widehat{X}_B & \widehat{W}_B \end{bmatrix} = \pi_B \begin{bmatrix} X & W \end{bmatrix} + B_{\text{err}} \tag{9}$$

where $W = \pi_B^{-1} W_B$ is $N \times 1$.

Our goal is to determine the permutations, π_A and π_B, that associate each of the N targets recorded by Sensors A and B, respectively, with those recorded by Sensor C. We formulate this association problem as a combinatorial optimization problem that involves the minimization of a cost function

$$\ell : S_N \times S_N \to [0, \infty) \tag{10}$$

where $\ell(\widehat{\pi}_A, \widehat{\pi}_B)$ represents the cost of using permutations $\widehat{\pi}_A$ and $\widehat{\pi}_B$ to associate data from Sensors A and B with those from Sensor C, respectively.

In the following sections, we will define the cost function, ℓ, in terms of a Mahalanobis distance using the requirement that, in the absence of measurement error, the orthogonal projections of the target positions measured by Sensor C onto the planes of Sensors A and B should match the target positions measured by Sensors A and B, respectively. Specifically, the permutations π_A and π_B should be chosen so that

$$\pi_A^{-1} \begin{bmatrix} X_A & Y_A \end{bmatrix} = \begin{bmatrix} X_C & Y_C & Z_C \end{bmatrix} M_A \tag{11}$$

$$\pi_B^{-1} \begin{bmatrix} X_B & W_B \end{bmatrix} = \begin{bmatrix} X_C & Y_C & Z_C \end{bmatrix} M_B. \tag{12}$$

III. COST FUNCTION FOR ORTHOGONAL SENSOR PLANES

To simplify the discussion, we first consider the special case that the planes of Sensors A and B are orthogonal ($\theta = \pi/2$). In the presence of measurement error, by (4) and (6), the difference in the positions measured by Sensors A and C is

$$\begin{aligned} &\pi_A^{-1} \begin{bmatrix} \widehat{X}_A & \widehat{Y}_A \end{bmatrix} - \begin{bmatrix} \widehat{X}_C & \widehat{Y}_C \end{bmatrix} \\ &= \left(\pi_A^{-1} \pi_A \begin{bmatrix} X & Y \end{bmatrix} + \pi_A^{-1} A_{\text{err}} \right) \\ &\quad - \left(\begin{bmatrix} X & Y \end{bmatrix} + C_{\text{err}}[*, (1, 2)] \right) \\ &= A_{\text{err}} - C_{\text{err}}[*, (1, 2)] \end{aligned} \tag{13}$$

since the rows of A_{err} are i.i.d. random variables

$$\pi_A^{-1} A_{\text{err}} \overset{(d)}{=} \pi_A A_{\text{err}} \overset{(d)}{=} A_{\text{err}}$$

where $\overset{(d)}{=}$ denotes equality in distribution. Similarly,

$$\pi_B^{-1} \begin{bmatrix} \widehat{X}_B & \widehat{W}_B \end{bmatrix} - \begin{bmatrix} \widehat{X}_C & \widehat{Z}_C \end{bmatrix} = B_{\text{err}} - C_{\text{err}}[*, (1, 3)]. \tag{14}$$

Let $D_{N \times 4}(\pi_A, \pi_B)$ be the $N \times 4$ data matrix

$$D_{N \times 4}(\pi_A, \pi_B) = \begin{bmatrix} D_1 & D_2 & D_3 & D_4 \end{bmatrix}$$

where

$$\begin{aligned} D_1 &= \pi_A^{-1} \widehat{X}_A - \widehat{X}_C \\ D_2 &= \pi_A^{-1} \widehat{Y}_A - \widehat{Y}_C \\ D_3 &= \pi_B^{-1} \widehat{X}_B - \widehat{X}_C \\ D_4 &= \pi_B^{-1} \widehat{W}_B - \widehat{Z}_C. \end{aligned}$$

By (13) and (14), the vector $\vec{x}(\pi_A, \pi_B) \in \mathbf{R}^{4N}$ obtained by concatenating the columns of the matrix $D_{N \times 4}(\pi_A, \pi_B)$ is multivariate normally distributed with mean zero and a $4N \times 4N$ covariance matrix, $\widetilde{\Sigma}_{4\text{D}}$, that is determined by Σ_A, Σ_B, and Σ_C. Our goal then is to determine the permutations $(\widehat{\pi}_A, \widehat{\pi}_B)$ that maximize the likelihood function

$$L((\pi_A, \pi_B)|\vec{x}) = \exp\left[-\vec{x}(\pi_A, \pi_B)^T \widetilde{\Sigma}_{4\text{D}}^{-1} \vec{x}(\pi_A, \pi_B) \right] \tag{15}$$

which is equivalent to minimizing the Mahalanobis distance

$$\ell(\pi_A, \pi_B) = \vec{x}(\pi_A, \pi_B)^T \widetilde{\Sigma}_{4\text{D}}^{-1} \vec{x}(\pi_A, \pi_B). \tag{16}$$

The advantage of defining $D_{N\times 4}$ using differences of measurements is that the unknown positions of the targets are removed, simplifying the computation of the cost function ℓ.

Since $S_N \times S_N$ has $(N!)^2$ elements, an exhaustive search for the global minimum of ℓ is intractable for $N \gtrapprox 10$. On the other hand, the simpler problem of associating measurements made by two sensors can be formulated as an optimization problem on S_N for which $\mathcal{O}(N^3)$-algorithms, such as Munkres' algorithm, exist [13]. To obtain an efficient, approximate solution to the 322-sensor assignment problem, we devised two algorithms that use Munkres' algorithm to solve two-way assignment subproblems. We refer to these two algorithms as the 4D and 5D algorithms.

Before describing these algorithms, we derive an alternate formula for the Mahalanobis distance, ℓ, in (16). Recall that the definition of ℓ was motivated by the requirements, (8) and (9), that the projections of the measured data from Sensor C onto the planes of Sensors A and B should be aligned with the data obtained from Sensors A and B. We observe from (8) and (9) that, in the absence of measurement error, the compatibility condition

$$\pi_A^{-1} X_A = \pi_B^{-1} X_B \tag{17}$$

must also hold for the optimal choice of permutations. This condition will be used in the 5D algorithm to further constrain the search for the optimal pair of permutations.

By (4) and (5), we observe that in the presence of measurement error the difference in the x-coordinate of the measured positions of Sensors A and B is

$$\pi_A^{-1}\widehat{X}_A - \pi_B^{-1}\widehat{X}_B = A_{\text{err}}[*,1] - B_{\text{err}}[*,1]. \tag{18}$$

Then, we define an $N \times 5$ data matrix by

$$D_{N\times 5}(\pi_A,\pi_B) = \begin{bmatrix} D_{N\times 4}(\pi_A,\pi_B) & D_5 \end{bmatrix} \tag{19}$$

where

$$D_5 = \pi_A^{-1}\widehat{X}_A - \pi_B^{-1}\widehat{X}_B.$$

Let $\widetilde{\Sigma}_{5\text{D}}$ be the $5N \times 5N$ covariance matrix associated with the data in $D_{N\times 5}$. Since $D_5 = D_1 - D_3$ for any set of measurements, the covariance matrix $\widetilde{\Sigma}_{5\text{D}}$ is not positive definite but rather has rank $4N$. A calculation shows that $\widetilde{\Sigma}_{5\text{D}}$ is the 5×5 block matrix whose (k,l)-block is the $N \times N$ matrix

$$(\widetilde{\Sigma}_{5\text{D}})_{kl} \;=\; (\Sigma_{5\text{D}})_{kl}\, I_{N\times N} \tag{20}$$

where $I_{N\times N}$ is the $N \times N$ identity matrix and $\Sigma_{5\text{D}}$ is the 5×5 matrix given by

$$\Sigma_{5\text{D}} = \begin{bmatrix} \Sigma_{AC} & C_{ACBC} & C_{ACAB} \\ C_{ACBC}^T & \Sigma_{BC} & C_{BCAB} \\ C_{ACAB}^T & C_{BCAB}^T & \Sigma_{AB} \end{bmatrix} \tag{21}$$

whose entries are the matrices

$$\Sigma_{AC} = \Sigma_A + \Sigma_C[(1,2),(1,2)]$$

$$\Sigma_{BC} = \Sigma_B + \Sigma_C[(1,3),(1,3)]$$

$$\Sigma_{AB} = \Sigma_A[1,1] + \Sigma_B[1,1]$$

$$C_{ACBC} = \begin{bmatrix} \Sigma_C[1,1] & \Sigma_C[1,3] \\ \Sigma_C[1,2] & \Sigma_C[2,3] \end{bmatrix}$$

$$C_{ACAB} = \Sigma_A[*,1]$$

$$C_{BCAB} = -\Sigma_B[*,1].$$

Since $\Sigma_{5\text{D}}$ is singular, we can use the Moore–Penrose pseudoinverse of $\Sigma_{5\text{D}}$ to define the distribution of each row of D. Let

$$\Sigma_{5\text{D}} = VEV^T$$

denote the eigendecomposition of $\Sigma_{5\text{D}}$, arranged so that the last eigenvalue in E is 0 and the first four are positive. Let V_4 denote the first four columns of V and let E_4 denote the submatrix consisting of the first four rows and columns of E. Then, the Moore–Penrose pseudoinverse of $\Sigma_{5\text{D}}$ is

$$\Sigma_{5\text{D}}^{-} = V_4 E_4^{-1} V_4^T.$$

Therefore, the log-likelihood function in (16) for parameters $\widehat{\pi}_A$ and $\widehat{\pi}_B$ can also be expressed as

$$l(\widehat{\pi}_A,\widehat{\pi}_B) = \sum_{i=1}^{n} D_i(\widehat{\pi}_A,\widehat{\pi}_B) V_4 E_4^{-1} V_4^T D_i^T(\widehat{\pi}_A,\widehat{\pi}_B) \tag{22}$$

where $D_i(\widehat{\pi}_A,\widehat{\pi}_B)$ denotes the ith row of $D_{N\times 5}(\widehat{\pi}_A,\widehat{\pi}_B)$.

The problem of simultaneously finding permutations $\widehat{\pi}_A$ and $\widehat{\pi}_B$ that minimize the association cost is not a standard association problem since $\ell : S^N \times S^N \to \mathbf{R}$ rather than $\ell : S^N \to \mathbf{R}$. Instead, we can treat the problem as two separate association problems by using columns 1 and 2 of $D_{N\times 5}$ to associate the targets measured by A with those from C and using columns 3 and 4 of $D_{N\times 5}$ to associate the measurements from B with those from C. Specifically, the first association subproblem is obtained by extracting columns 1 and 2 from $D_{N\times 5}$ to obtain the $N \times 3$ data matrix

$$D^{(AC)}(\widehat{\pi}_A) = \begin{bmatrix} \widehat{\pi}_A\widehat{X}_A - \widehat{X}_C,\ \widehat{\pi}_A\widehat{Y}_A - \widehat{Y}_C \end{bmatrix} \tag{23}$$

which only depends on $\widehat{\pi}_A$. The cost function for this subproblem is defined by

$$\ell_{AC}(\widehat{\pi}_A) = \sum_{i=1}^{n} D_i^{(AC)} \Sigma_{AC}^{-1} D_i^{(AC)T}. \tag{24}$$

Similarly, the second subproblem, which is to associate the data from B with those from C, is determined by the data matrix

$$D^{(BC)}(\widehat{\pi}_B) = \begin{bmatrix} \widehat{\pi}_B^{-1}\widehat{X}_B - \widehat{X}_C,\ \widehat{\pi}_B^{-1}\widehat{W}_B - \widehat{Z}_C \end{bmatrix} \tag{25}$$

and the corresponding cost function

$$\ell_{BC}(\widehat{\pi}_B) = \sum_{i=1}^{n} D_i^{(BC)} \Sigma_{BC}^{-1} D_i^{(BC)T}. \tag{26}$$

This approach is the essence of the 4D algorithm for the 322-sensor association problem, which involves solving two independent matching problems: an A-to-C and a B-to-C match, from which an A-to-B match can be inferred. The 4D algorithm can be summarized as follows:

#initialize
$\widehat{\pi}_B = \text{argmin}(l_{BC}(\widehat{\pi}_B))$
$\widehat{\pi}_A = \text{argmin}(l_{AC}(\widehat{\pi}_A))$
end

In the 5D algorithm, we explicitly impose the additional consistency condition encoded in column 5 of $D_{N\times 5}$ that represents the differences in the observed positions between A and B in their shared dimension. With this iterative algorithm, we first use a current estimate for the A-to-C match and optimize the B-to-C match. The Mahalanobis distance objective function for this B-to-C match involves a $3N \times 3N$ covariance matrix that incorporates the additional A–B consistency condition. In the second stage of each iteration, we switch the roles of A and B and optimize the A-to-C match. Given a reasonable initial guess for the A-to-C match, only a few iterations are typically required for the algorithm to converge.

In detail, the 5D algorithm is given as follows. Suppose we have available an initial permutation, $\widehat{\pi}_B^{(0)}$, for Sensor B and wish to associate the measurements made by Sensor A with those made by Sensor C given $\widehat{\pi}_B^{(0)}$. This data association problem can be based on columns 1, 2, and 5 of (21) with $\widehat{\pi}_B^{(0)}$ fixed

$$D^{(ABC)}\big(\widehat{\pi}_A|\widehat{\pi}_B^{(0)}\big) = [D_1,\ D_2,\ D_5]. \tag{27}$$

The covariance matrix for $D^{(ABC)}$ is the submatrix consisting of rows and columns 1, 2, and 5 of (21)

$$\Sigma_{ABC} = \Sigma_{5D}[(1,2,5),(1,2,5)]. \tag{28}$$

The matrix Σ_{ABC} is nonsingular and the objective function for $\widehat{\pi}_A$ given $\widehat{\pi}_B^{(0)}$ is given by

$$\ell_{ABC}\big(\widehat{\pi}_A|\widehat{\pi}_B^{(0)}\big) = \sum_{i=1}^{n} D_i^{(ABC)}\Sigma_{ABC}^{-1}\big(D_i^{(ABC)}\big)^T. \tag{29}$$

Let $\widehat{\pi}_A^{(0)}$ denote the permutation that minimizes this cost function. Since $\ell_{ABC} : S_N \to \mathbf{R}$, an approximation to $\widehat{\pi}_A^{(0)}$ can be obtained using a two-way assignment algorithm [10], [13]. Once we have an estimate for $\widehat{\pi}_A^{(0)}$, we can then refine the initial estimate of π_B using columns 3, 4, and 5 of (21)

$$D^{(BAC)}\big(\widehat{\pi}_B|\widehat{\pi}_A^{(0)}\big) = [D_3,\ D_4,\ D_5]. \tag{30}$$

The covariance matrix for $D^{(BAC)}$ is

$$\Sigma_{BAC} = \Sigma_{5D}[(3,4,5),(3,4,5)]. \tag{31}$$

The matrix Σ_{BAC} is nonsingular and, as above, the cost function for the new estimate $\widehat{\pi}_B$ is

$$\ell_{BAC}\big(\widehat{\pi}_B|\widehat{\pi}_A^{(0)}\big) = \sum_{i=1}^{n} D_i^{(BAC)}\Sigma_{BAC}^{-1}\big(D_i^{(BAC)}\big)^T. \tag{32}$$

Now let $\widehat{\pi}_B^{(1)}$ denote the permutation that minimizes this cost function. The process of solving these alternating two-way assignment problems can be repeated using the updated estimates from the previous step, continuing until the estimates remain unchanged.

It remains to define an initial estimate for π_B. This can be obtained using columns 3 and 4 of $D_{N\times 5}$

$$D^{(BC)}(\widehat{\pi}_B) = \big[\widehat{\pi}_B\widehat{X}_B - \widehat{X}_C,\ \widehat{\pi}_B\widehat{W}_B - \widehat{Z}_C\big] \tag{33}$$

to assign the data from B to those from C without A. The cost function for this problem is

$$\ell_{BC}(\widehat{\pi}_B) = \sum_{i=1}^{n} D_i^{(BC)}\Sigma_{BC}^{-1}\big(D_i^{(BC)}\big)^T. \tag{34}$$

The resulting 5D algorithm can be expressed as follows:

#initialize
$\widehat{\pi}_B^{(0)} = \text{argmin}(l_{BC}(\widehat{\pi}_B))$
$\widehat{\pi}_A^{(0)} = \text{argmin}(l_{ABC}(\widehat{\pi}_A|\widehat{\pi}_B^{(0)}))$

#iterate
for i in 1 : maxiter
{
$\widehat{\pi}_B^{(i)} = \text{argmin}(l_{BAC}(\widehat{\pi}_B|\widehat{\pi}_A^{(i-1)}))$
$\widehat{\pi}_A^{(i)} = \text{argmin}(l_{ABC}(\widehat{\pi}_A|\widehat{\pi}_B^{(i-1)}))$
if $\widehat{\pi}_A^{(i)} == \widehat{\pi}_A^{(i-1)}$ and $\widehat{\pi}_B^{(i)} == \widehat{\pi}_B^{(i-1)}$
break
}

IV. COST FUNCTION FOR NONORTHOGONAL SENSOR PLANES

We now consider the more realistic situation in which the sensor planes A and B are not orthogonal, $(\theta \neq \pi/2)$. In this case, the $N \times 5$ data matrix is given by

$$D = \begin{bmatrix} D_1 & D_2 & D_3 & D_4 & D_5 \end{bmatrix} \tag{35}$$

where

$$\begin{aligned} D_1 &= \widehat{\pi}_A^{-1}\widehat{X}_A - \widehat{X}_C \\ D_2 &= \widehat{\pi}_A^{-1}\widehat{Y}_A - \widehat{Y}_C \\ D_3 &= \widehat{\pi}_B^{-1}\widehat{X}_B - \widehat{X}_C \\ D_4 &= \widehat{\pi}_B^{-1}\widehat{W}_B - \cos(\theta)\widehat{Y}_C - \sin(\theta)\widehat{Z}_C \\ D_5 &= \widehat{\pi}_A^{-1}\widehat{X}_A - \widehat{\pi}_B^{-1}\widehat{X}_B. \end{aligned}$$

Notice that the only column that is different from the case of orthogonal sensor planes is D_4, which corresponds to the orthogonal projection of the measurements made by Sensor C onto the tilted plane of Sensor

B. The associated $N \times 5$ error matrix is given by

$$E = \begin{bmatrix} E_1 & E_2 & E_3 & E_4 & E_5 \end{bmatrix} \tag{36}$$

where

$$\begin{aligned} E_1 &= A_{\text{err}}[*, 1] - C_{\text{err}}[*, 1] \\ E_2 &= A_{\text{err}}[*, 2] - C_{\text{err}}[*, 2] \\ E_3 &= B_{\text{err}}[*, 1] - C_{\text{err}}[*, 1] \\ E_4 &= B_{\text{err}}[*, 2] - \cos(\theta) C_{\text{err}}[*, 2] - \sin(\theta) C_{\text{err}}[*, 3] \\ E_5 &= A_{\text{err}}[*, 1] - B_{\text{err}}[*, 1]. \end{aligned}$$

Since columns 1, 2, and 5 of (35) are the same as in the nonorthogonal case, the covariance matrix Σ_{ABC} is the same as before. On the other hand, Σ_{BAC} is now given by

$$\Sigma_{BAC} = \begin{bmatrix} \Sigma_{BC}(1) & \Sigma_{BC}(1,2) & -\Sigma_B[1,1] \\ \Sigma_{BC}(1,2) & \Sigma_{BC}(2) & -\Sigma_B[2,1] \\ -\Sigma_B[1,1] & -\Sigma_B[2,1] & \Sigma_{AB}(1) \end{bmatrix} \tag{37}$$

where

$$\begin{aligned} \Sigma_{BC}(1) &= \Sigma_B[1,1] + \Sigma_C[1,1] \\ \Sigma_{BC}(1,2) &= \Sigma_B[1,2] + C_\theta \Sigma_C[1,2] + S_\theta \Sigma_C[1,3] \\ \Sigma_{BC}(2) &= \Sigma_B[2,2] + C_\theta^2 \Sigma_C[2,2] + \Gamma_\theta \Sigma_C[2,3] \\ &\quad + S_\theta^2 \Sigma_C[3,3] \end{aligned}$$

with $C_\theta = \cos\theta, S_\theta = \sin\theta$, and $\Gamma_\theta = 2 S_\theta C_\theta$.

V. ESTIMATION OF TARGET POSITIONS

Once the target assignments have been made, the target positions can be estimated using a weighted combination of the positions measured by each sensor in each coordinate, where the weights are inversely proportional to the standard deviations of the sensor position errors. For simplicity, in this section we assume that the sensor planes are orthogonal and the covariance matrices Σ_A, Σ_B, and Σ_C are diagonal. In principle, it is possible to relax these simplifying assumptions. Let

$$\widehat{P} = (\widehat{X}_A, \widehat{Y}_A, \widehat{X}_B, \widehat{W}_B, \widehat{X}_C, \widehat{Y}_C, \widehat{Z}_C)^T \tag{38}$$

denote the positions of an arbitrary target as measured by each sensor after assignments have been made for A and B and let

$$\sigma_{AX}, \sigma_{AY}, \sigma_{BX}, \sigma_{BW}, \sigma_{CX}, \sigma_{CY}, \sigma_{CZ} \tag{39}$$

denote the standard deviations of the position errors in the respective coordinates. Then, the estimated target positions are given by

$$\begin{aligned} \widehat{X} &= w_{XA}\widehat{X}_A + w_{XB}\widehat{X}_B + w_{XC}\widehat{X}_C \\ \widehat{Y} &= w_{YA}\widehat{Y}_A + w_{YC}\widehat{Y}_C \\ \widehat{Z} &= w_{ZB}\widehat{W}_B + w_{ZC}\widehat{Z}_C \end{aligned} \tag{40}$$

where $w_{XA} = \sigma_{AX}^{-1}/\delta_X$, the other weights are defined similarly, and

$$\begin{aligned} \delta_X &= \left(\sigma_{AX}^{-1} + \sigma_{BX}^{-1} + \sigma_{CX}^{-1}\right) \\ \delta_Y &= \left(\sigma_{AY}^{-1} + \sigma_{CY}^{-1}\right) \\ \delta_Z &= \left(\sigma_{BW}^{-1} + \sigma_{CZ}^{-1}\right). \end{aligned}$$

Note that a sensor with higher relative variability will have lower weight associated with its observed positions.

VI. RESULTS

In this section, we present the results of numerical simulations designed to compare the performance of a 32-sensor system, in which target matching is performed between pairs of measurements using Munkres' algorithm, to that of a 322-sensor system, in which target matching is performed between triples of measurements using the 5D algorithm. We also compare the performance of the 4D and 5D algorithms for a 322-sensor system. First, we study the percentage of correct assignments with the different methods for six choices of the covariance matrix of Sensor C in the special case that the planes of the two 2D sensors are orthogonal. Second, we investigate how the percentage of correct assignments for the 322-sensor system with the 5D algorithm depends on the angle, θ, between the planes of the two 2D sensors, and also on the orientation of the 3D sensor relative to that of the two 2D sensors. Third, we compare the RMSE in the estimated target positions obtained using a 32-sensor system, a 322-sensor system with the 4D algorithm, and a 322-sensor system with the 5D algorithm. Finally, we present results that illustrate how the execution time of the 5D algorithm depends on both the number of targets and the average separation between the targets.

A. Target and Sensor Simulation Scenarios

To evaluate how the performance of the 32- and 322-sensor system M2MA algorithms depends on the average separation between the targets, we generated data by simulating a collection of N targets with randomly distributed constant velocities that originate from the same location, and move apart like a cluster of exploding fireworks. The velocity, $\mathbf{v}_n$, of the nth target is given by

$$\mathbf{v}_n = \mathbf{v}_{\text{g}} + v_n \mathbf{u}_n \tag{41}$$

where $\mathbf{v}_{\text{g}}$ is a common group velocity and $v_n \mathbf{u}_n$ is the drift velocity of the nth target relative to the group. Here, v_n is the drift speed and $\mathbf{u}_n$ is a unit drift-direction vector. The drift speeds, v_n, are chosen from a gamma distribution with mean, μ_{d}, and standard deviation, σ_{d}, and the drift directions are sampled from a uniform distribution on the unit sphere. We ran the simulation of the target trajectories for a total time T with time steps of size Δt. We generated data in two target simulation scenarios, I

TABLE II
Parameters Used to Generate Targets in Simulation Scenarios I and II

Parameter	Meaning	Scenario I	Scenario II
N	Number of targets	50	50
$\mathbf{v}_g$	Group velocity of targets	$(1, 1, \sqrt{2})$ m/s	100 $(1, 1, \sqrt{2})$ m/s
μ_d	Mean speed for target drift	1 m/s	100 m/s
σ_d	Standard deviation of speed for target drift	0.25 m/s	20 m/s
T	Total target simulation time	1.4 s	1.4 s
Δt	Time step for target simulation	0.02 s	0.02 s

Table III
Sensor Parameters Used in the Simulations

Parameter	Meaning	Values
Σ_A, Σ_B	Sensor covariance matrices	$I_{2\times 2}$
Σ_C	Sensor covariance matrices (cigars)	diag[16, 1, 1], diag[1, 16, 1], diag[1, 1, 16]
	Sensor covariance matrices (pancakes)	diag[1, 16, 16], diag[16, 1, 16], diag[16, 16, 1]
θ	Angle between sensor planes A and B	$0^\circ, 10^\circ, 60^\circ, 90^\circ$
ϕ	Rotation angle of Sensor C	$0^\circ, 30^\circ, 60^\circ, 90^\circ$

and II. The parameter values for these two scenarios are shown in Table II.

After each time step, the target positions were orthogonally projected onto the planes of the two 2D sensors. The measurement uncertainty of Sensors A and B was modeled by adding Gaussian white noise to these target positions with mean zero and covariance matrices Σ_A and Σ_B, respectively. To simulate measurements from Sensor C, we added Gaussian white noise with mean zero and covariance matrix Σ_C to the computed target positions. We replicated each target trajectory simulation $M = 1000$ times, each with a different choice of the random parameters, v_n, $\mathbf{u}_n$, and different noise realizations in each sensor. The performance results we present below were obtained by averaging over these M replications.

The parameter values we used to specify the geometric configurations and measurement uncertainties of the sensors are summarized in Table III. For the results in Sections VI-B, VI-C, and VI-E, the planes of the two 2D sensors were orthogonal ($\theta = 90^\circ$). However, in Section VI-D we also present results for nonorthogonal sensor planes using the values of θ given in Table III. For the two 2D sensors, the covariance ellipsoid was chosen to be the unit sphere. We considered both cigar-shaped and pancake-shaped covariance ellipsoids for the 3D sensor. With cigar-shaped ellipsoids, the length of the major axis was 4 and the other two axes were of length 1. With pancake-shaped ellipsoids, the length of the minor axis was 1 and the other two axes were of length 4. Unless otherwise noted, the covariance matrix, Σ_C, of the 3D sensor was chosen to be diagonal. However, at the end of Section VI-D, we present results for a nondiagonal covariance matrix obtained by rotating the matrix diag[16, 1, 1] about the y-axis by the angles, ϕ, shown in the last row of Table III. To illustrate how the choice of covariance matrix is related to the measurement uncertainty of a realistic 3D radar, we observe that a radar that has a limited field of view, points in the z-direction, and has high range resolution and low angular resolution corresponds to a 3D sensor with a pancake-shaped covariance ellipsoid with $\Sigma_C = \text{diag}[\alpha, \alpha, \beta]$, where $\alpha \gg \beta$.

In Fig. 2a, we show the projection onto the xz-plane of the actual trajectories of the 50 targets for one random realization of the target trajectories in Scenario I. The target tracks diverge more from each other in Scenario II than in Scenario I. In Fig. 2b, we show the projection onto the xz-plane of these trajectories as measured by Sensor C in the case that $\Sigma_C = \text{diag}[16, 1, 1]$, and in Fig. 2c we show the targets in the xz-plane at the final time as measured by Sensors B and C. For this result, the planes of the 2D sensors were orthogonal.

At each time step, we computed the average distance between pairs of targets. Since $\Sigma_A = \Sigma_B = \text{Id}$, we plot the performance of each algorithm as a function of the normalized average separation between the targets, where the normalization factor is the square root of the trace of Σ_C.

B. Data Association With 32- and 322-Sensor Systems

In Fig. 3, we compare the performance of a 32-sensor system, consisting of sensors X and C ($X = A$ or B), to that of the 322-sensor system by plotting the percentage of correct two-way assignments for the X-to-C match as a function of the normalized average separation between the targets. These results were obtained in Scenario II in the case that the planes of Sensors A and B were orthogonal, and for the six choices of co-

variance matrix, Σ_C, shown in Table III. The results for the three cigar-shaped covariance ellipsoids are shown in the top row of Fig. 3, and those for the pancake-shaped ellipsoids are shown in the bottom row. In each subfigure, we compare the results obtained using the two 32-sensor systems to the 322-sensor system. For the 32-sensor systems, we used Munkres' algorithm to solve the assignment problem, which we refer to as the 2D algorithm in Fig. 3. Although the 5D algorithm produces three-way A-to-B-to-C matches, to make fair comparisons to the 32-sensor systems we only consider the percentages of correct two-way A-to-C and B-to-C assignments.

We begin with two preliminary observations. First, due to the inherent symmetry, for the 2D algorithm the percentage of correct B-to-C matches in Fig. 3b is statistically identical to the percentage of correct A-to-C matches in Fig. 3c. However, for the 5D algorithm, the A-to-C performance curve in Fig. 3b is slightly different from the B-to-C performance curve in Fig. 3c since this iterative algorithm is not invariant to the y-to-z coordinate switch. (The first iteration always involves a B-to-C match.) Second, in Fig. 3b, the performance of both algorithms is significantly better for the B-to-C match than for the A-to-C match since when $\Sigma_C = \text{diag}[1, 16, 1]$, the projection of Σ_C onto the plane of Sensor A is diag[1, 16], whereas the projection onto the plane of Sensor B is diag[1, 1].

Next, we compare the percentage of correct two-way matches for the 2D and 5D algorithms. In Fig. 3, we observe that the performance of the two algorithms is comparable whenever the (1,1) entry of Σ_C is comparable to that of Σ_A and Σ_B (see Fig. 3b–d). The reason is that when the uncertainty in the A–C and B–C data in their common x-dimension is comparable to the uncertainty in the A–B data, there is no advantage to be gained by applying the A-to-B consistency check in the 5D algorithm. On the other hand, when the (1,1) entry of Σ_C is significantly larger than that of Σ_A and Σ_B, the performance is generally much better with the 5D algorithm (see Fig. 3a, e, and f).

C. 322-Sensor System: 4D Versus 5D Algorithm

In this subsection, we compare the performance of the 4D and 5D algorithms for the 322-sensor system. In Fig. 4, we plot the percentage of correct three-way A-to-B-to-C matches as a function of the normalized average separation between the targets. By a correct assignment for the A-to-B-to-C match, we mean that the A-to-C and B-to-C (and hence the A-to-B) matches are all correct. As in Fig. 3, these results were obtained in Scenario II in the case that the planes of Sensors A and B were orthogonal, $\theta = 90^\circ$.

First, we discuss how the percentage of correct three-way matches for the 4D algorithm depends on the choice of Σ_C. In Fig. 4a and b, we see that when the normalized average separation is 10, the percentage of correct assignments is 75% with $\Sigma_C = \text{diag}[1, 16, 1]$ but only 65% with $\Sigma_C = \text{diag}[16, 1, 1]$. The reason for the poorer performance when the uncertainty in the Sensor C measurements is larger on the x-axis is that this axis is common to both 2D sensors. Consequently, the large uncertainty in the Sensor C data remains when the data are projected onto the xy-plane of Sensor A and onto the xz-plane of Sensor B. On the other hand, when $\Sigma_C = \text{diag}[1, 16, 1]$ the large uncertainty in the Sensor C data only remains when the data are projected onto one of these two planes, leading to better performance of the 4D algorithm in this case.

We now compare the percentage of correct three-way matches with the 4D and 5D algorithms. For the same reasons as in the discussion of Fig. 3, the 5D algorithm significantly outperforms the 4D algorithm when the (1,1) entry of Σ_C is significantly larger than that of Σ_A and Σ_B, while the performance of the two algorithms is comparable in the other three cases. The reason the 5D algorithm outperforms the 4D algorithm in some cases is

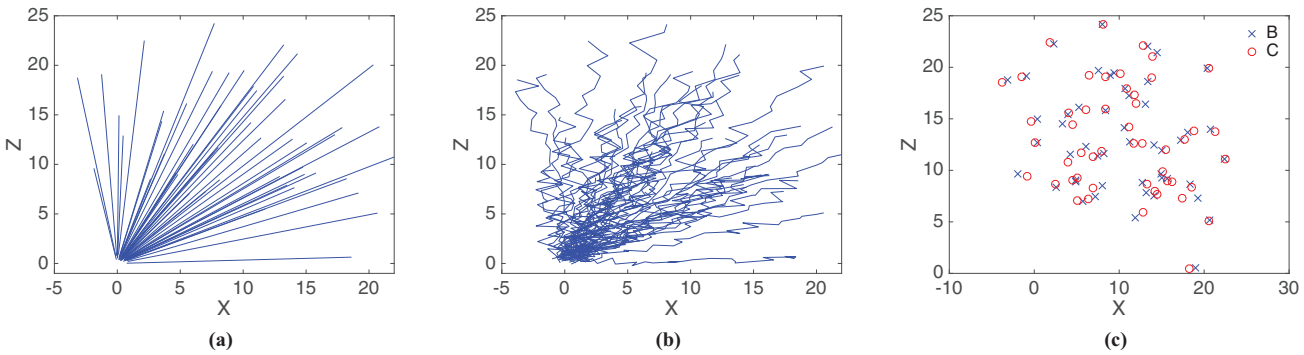
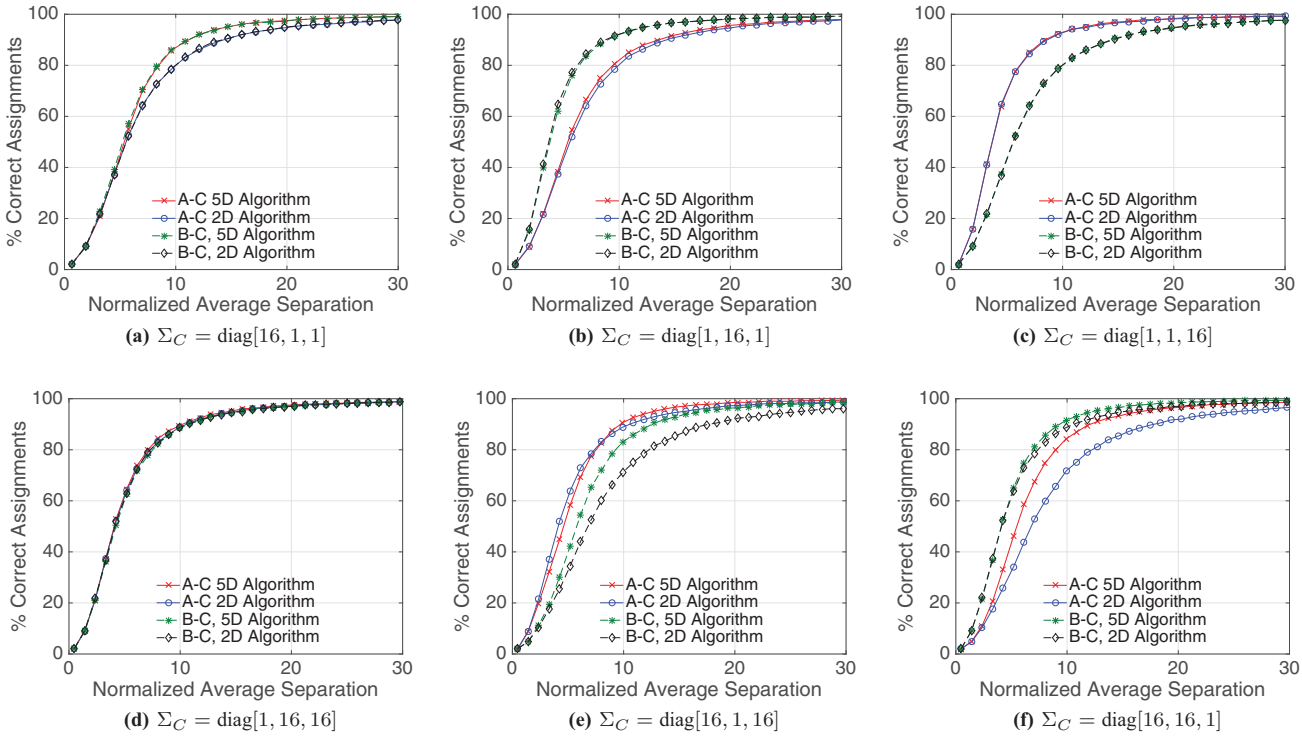


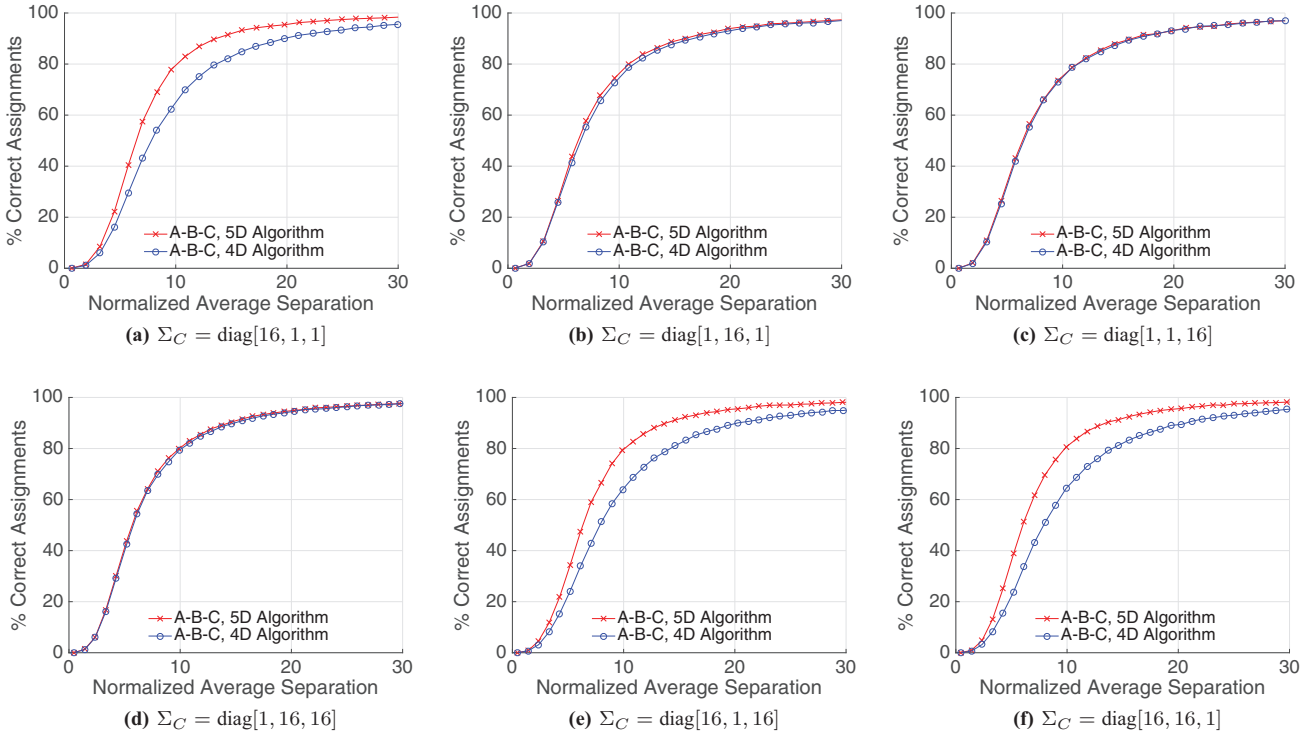
Fig. 2. (a) Projection into the x–z plane of the actual trajectories of 50 moving targets for one random realization of the target trajectory simulation in Scenario I. (b) Projection into the x–z plane of the trajectories of the same 50 moving targets as observed by Sensor C after the addition of Gaussian white noise with covariance matrix $\Sigma_C = \text{diag}[16, 1, 1]$. (c) Measured targets in the x–z plane as observed by Sensor B together with the projection into the x–z plane of the measured targets as observed by Sensor C. The planes of the two 2D sensors were orthogonal ($\theta = 90^\circ$) and the measurements were taken at the final time.



(a) $\Sigma_C = \text{diag}[16, 1, 1]$ **(b)** $\Sigma_C = \text{diag}[1, 16, 1]$ **(c)** $\Sigma_C = \text{diag}[1, 1, 16]$

(d) $\Sigma_C = \text{diag}[1, 16, 16]$ **(e)** $\Sigma_C = \text{diag}[16, 1, 16]$ **(f)** $\Sigma_C = \text{diag}[16, 16, 1]$

Fig. 3. Percentage of correct assignments for two-way matches as a function of the normalized average separation between the targets in Scenario II with orthogonal 2D sensor planes ($\theta = 90^\circ$) and $\phi = 0^\circ$. The results shown are for the A-to-C match using the 5D algorithm (red lines with crosses), the A-to-C match using Munkres' algorithm (blue lines with circles), the B-to-C match using the 5D algorithm (green dashed lines with stars), and the B-to-C match using Munkres' algorithm (black dashed lines with diamonds). The results with cigar-shaped covariance ellipsoids for Sensor C are shown in the top row, and those with pancake-shaped ellipsoids are shown in the bottom row.



(a) $\Sigma_C = \text{diag}[16, 1, 1]$ **(b)** $\Sigma_C = \text{diag}[1, 16, 1]$ **(c)** $\Sigma_C = \text{diag}[1, 1, 16]$

(d) $\Sigma_C = \text{diag}[1, 16, 16]$ **(e)** $\Sigma_C = \text{diag}[16, 1, 16]$ **(f)** $\Sigma_C = \text{diag}[16, 16, 1]$

Fig. 4. Percentage of correct assignments for the three-way A-to-B-to-C match as a function of the normalized average separation between the targets in Scenario II with orthogonal 2D sensor planes ($\theta = 90^\circ$) and $\phi = 0^\circ$. The results shown were obtained using the 5D algorithm (red line with crosses) and the 4D algorithm (blue line with circles). The results with cigar-shaped covariance ellipsoids for Sensor C are shown in the top row, and those with pancake-shaped ellipsoids are shown in the bottom row.

that the application of the A–B consistency check in column 5 of $D_{N\times 5}$ rules out some incorrect A-to-B matches made by the 4D algorithm. As a consequence, during the iteration process the 5D algorithm also corrects for some of the matching errors made by the poorer performing of the A-to-C and B-to-C matches.

D. Relative Sensor Orientation Effects for 322-Sensor

Having identified situations in which the performance with the 5D algorithm is significantly better than that with the other two methods, we now focus on how the percentage of correct three-way matches with the 5D algorithm depends on the relative orientation of the three sensors. We begin by studying how the 5D algorithm performs as the angle, θ, between the planes of the 2D sensors decreases from $\theta = 90^\circ$ to $\theta = 0^\circ$. These results were obtained with target simulation Scenario II. As $\theta \to 0^\circ$, the plane of Sensor B converges to that of Sensor A and neither of the 2D sensors has the ability to distinguish between two targets with the same (x, y)-value but different z-values. Although this observation suggests that the percentage of correct assignments should decrease as $\theta \to 0^\circ$, we will show that in many cases this decrease is slight. Moreover, in certain situations the performance is actually significantly better when the 2D sensor planes are parallel than when they are perpendicular, due to the suppression of noise from Sensor C that is orthogonal to the common plane of the 2D sensors.

In Fig. 5a and d, the percentage of correct three-way matches with the 5D algorithm is insensitive to the angle, θ, between the planes of Sensors A and B. The reason for this performance insensitivity is that the covariance ellipsoid for Sensor C has a circular cross section in the yz-plane, and so the projection of Σ_C onto the plane of Sensor B is independent of θ. Consequently, the performance of the B-to-C match (not shown) is completely independent of θ. Nevertheless, we do not expect the percentage of correct three-way matches with the 5D algorithm to be totally independent of θ, since the relative alignment of the two 2D sensors changes as θ changes.

The scenario in Fig. 5f, for which $\Sigma_C = \text{diag}[16, 16, 1]$, closely corresponds to a 322-sensor system in which all three sensors are mounted on a single platform with the 2D sensors pointing in similar directions (so that the A-to-B match solves the stereopsis problem), with the 3D sensor being a radar with good range accuracy, and with the 2D sensors having better angular measurement accuracy than the radar. In Fig. 5b and f, the percentage of correct three-way matches decreases more as $\theta \to 0^\circ$ than in Fig. 5a and d since in these cases the area of the covariance ellipse of the projection of Σ_C onto the plane of Sensor B increases as $\theta \to 0^\circ$. As a consequence, the performance of the B-to-C match (not shown) degrades as $\theta \to 0^\circ$. When $\theta = 0^\circ$, the performance is worse when $\Sigma_C = \text{diag}[16, 16, 1]$ than with the other five covariance matrices, since the covariance ellipse of the projection of Σ_C onto the xy-plane is largest in this case.

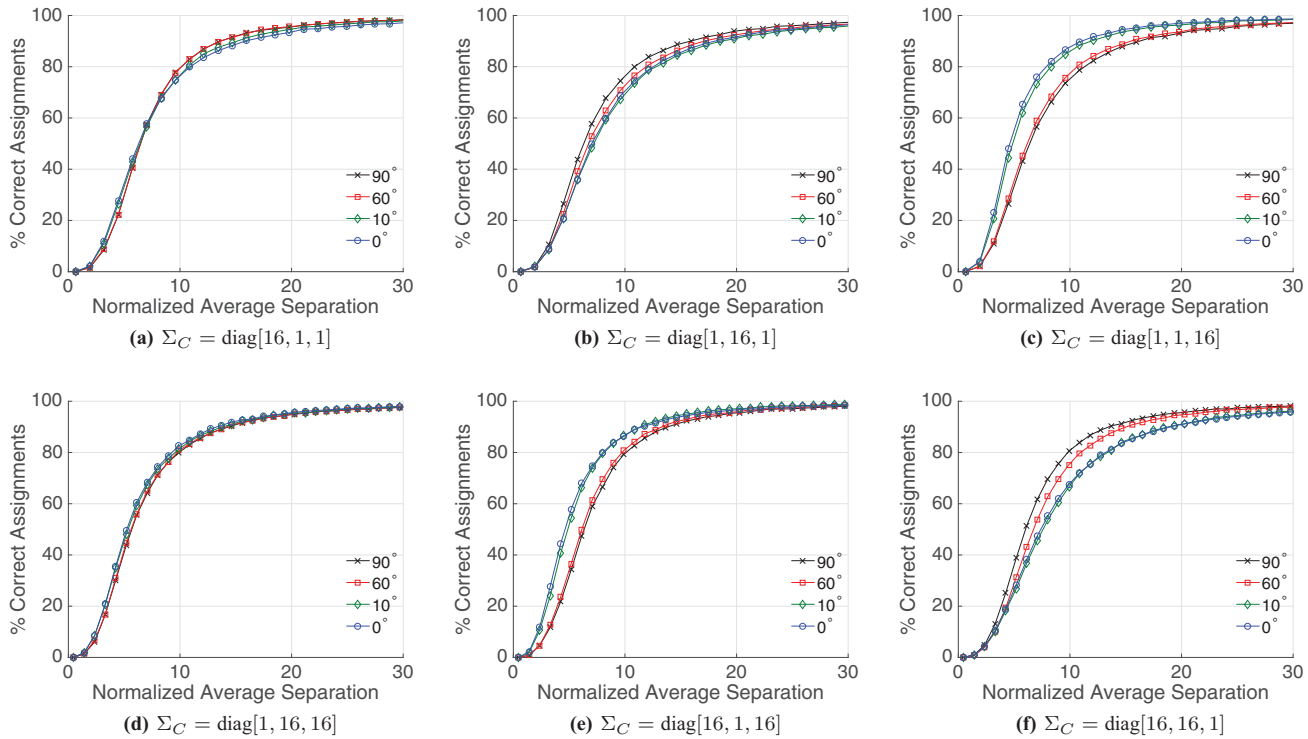


Fig. 5. Percentage of correct assignments for the A-to-B-to-C match using the 5D algorithm as a function of the normalized average distance between the targets in Scenario II with $\phi = 0^\circ$. The different curves show the results for different values of the angle, θ, between the planes of the two 2D sensor. The results with cigar-shaped covariance ellipsoids for Sensor C are shown in the top row, and those with pancake-shaped ellipsoids are shown in the bottom row.

In Fig. 5c and e, we see that rather than decreasing, the percentage of correct three-way matches increases significantly as $\theta \to 0^\circ$. The reason for this behavior is that when θ is small the large uncertainty of the Sensor C measurements in the z-dimension is (almost) completely suppressed when the Sensor C data are (almost) orthogonally projected onto the planes of both Sensors A and B. Since the uncertainty in the z-dimension is relatively large and affects all the measurements made by Sensor C, its suppression plays a much larger role in increasing the performance than the very minor reduction in performance that occurs whenever two targets with similar (x, y)-coordinates but different z-coordinates cannot be distinguished by nearly horizontal 2D sensors.

In Fig. 6a, we show the results we obtained when Σ_C is not diagonal. These results were obtained with target simulation Scenario II and $\theta = 90^\circ$. Specifically, we chose Σ_C to be a rotation about the y-axis by an angle, ϕ, of the matrix diag[16, 1, 1]. For these simulations, we chose $\phi = 0^\circ, 30^\circ, 60^\circ$, and 90°, as in Table III, so that when $\phi = 0^\circ$, $\Sigma_C = \text{diag}[16, 1, 1]$, and when $\phi = 90^\circ$, $\Sigma_C = \text{diag}[1, 1, 16]$. These results reproduce the expected decrease in the percentage of correct three-way matches that we see when comparing the $\theta = 90^\circ$ plots in Fig. 5a and b, and therefore serve to verify the correctness of the theory we developed for nondiagonal covariance matrices.

E. Target Position Estimates

For each replication, we computed the estimated target positions using (40) for three different assignment methods: Munkres' algorithm using data from one 2D sensor (A) and the 3D sensor, and the 4D and 5D algorithms. In Fig. 7, we plot the normalized RMSE in the estimated target positions as a function of the normalized average separation for the three assignment methods. The RMSE is the square root of the trace of the covariance matrix of the estimated target positions. In Fig. 7, both the RMSE and the average separation between the targets are normalized by the square root of the trace of the covariance matrix of Sensor C. Since this normalization factor equals the RMSE of the target positions as measured by Sensor C alone, when the normalized RMSE is less than 1 the RMSE performance of the algorithm is better than the RMSE performance with Sensor C alone. These results were obtained in target simulation Scenario I with $\theta = 90^\circ$ and $\phi = 0^\circ$. (The targets are less well separated in Scenario I than in Scenario II.)

In Fig. 7a, we show the results with $\Sigma_C = \text{diag}[16, 1, 1]$. The initial rapid rise of the RMSE occurs simply because the distance between the targets increases, thereby increasing the RMSE for *any* choice of assignment. However, once the target separation exceeds the uncertainty in the Sensor C data, i.e., once the normalized average separation is greater than 1, the performance of all three algorithms is better than the performance of Sensor C alone, and all three of the RMSE curves decrease as the target separation increases. Therefore, whenever the target separation exceeds the uncertainty in the Sensor C measurements, it is advantageous to use a 32- or 322-sensor system instead of a 3D sensor alone, provided that the uncertainty of the 2D sensor measurements is small compared to that of the 3D sensor. When the average separation distance is greater than 2, the performance is worst with the two-sensor assignment method and best with the 5D algorithm. On the other hand, when the normalized average separation is less than 1, the 5D algorithm does not perform as well as the other two methods. This result holds for essentially the same reasons we gave for the poorer performance of the 5D algorithm for the A-to-C match in Fig. 3e.

Finally, in Fig. 7c, we see that with $\Sigma_C = \text{diag}[1, 1, 16]$, the RMSE obtained using the two-sensor method remains large irrespective of the average separation between the targets, since the z-coordinate of the estimated target position in (40) is based only on the estimate, $\widehat{Z}_C$, which has large uncertainty, $\sigma_{CZ} = 4$. This result demonstrates that there can be a significant advantage to using a second 2D sensor, especially in situations where there is a large uncertainty in the measurements made by Sensor C in the dimension that is orthogonal to the plane of the first 2D sensor.

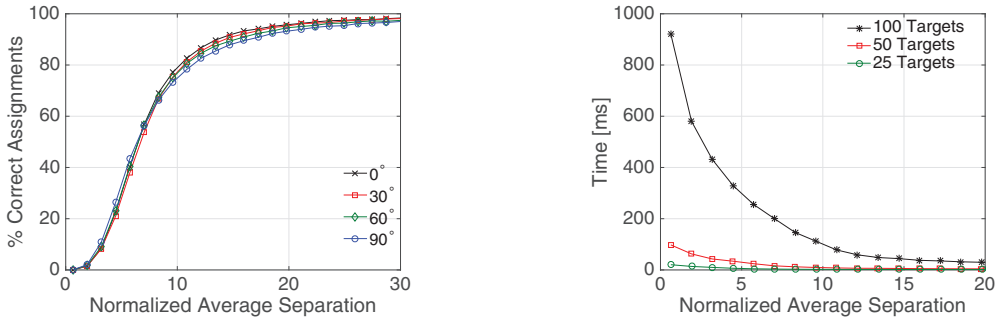
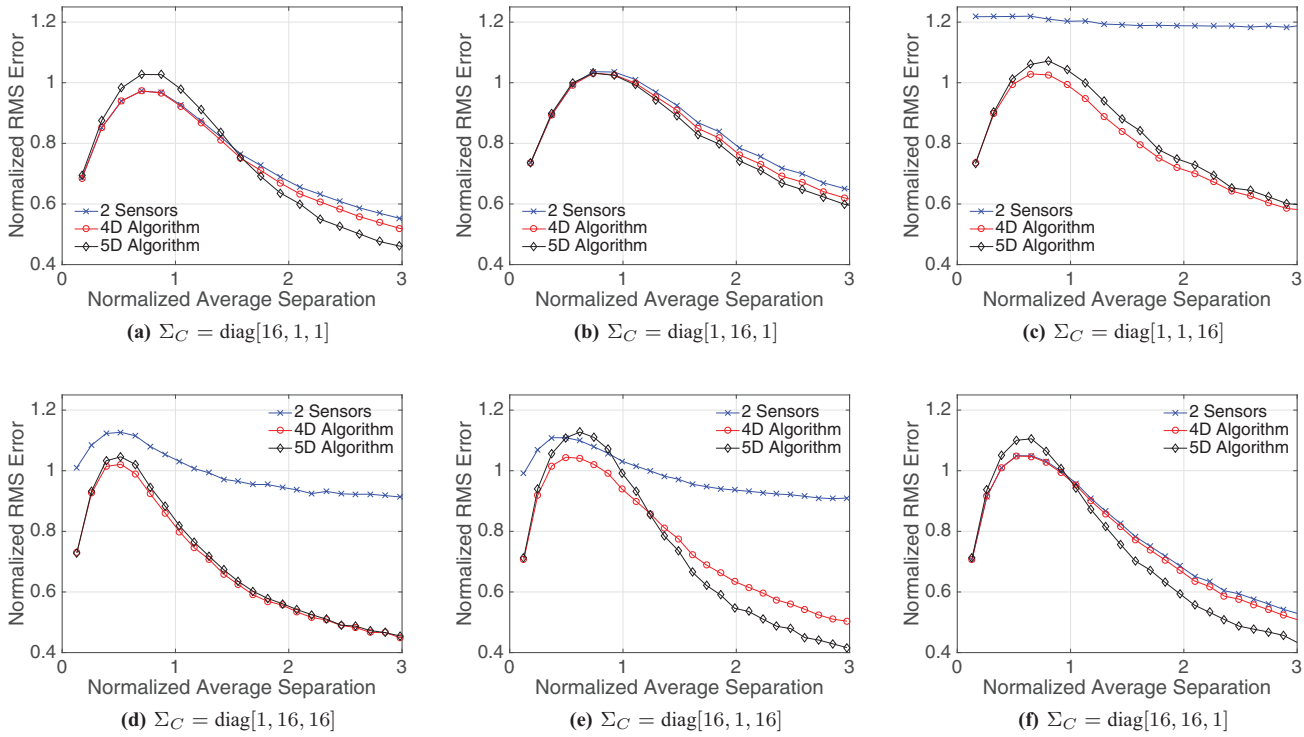


Fig. 6. (a) Percentage of correct assignments for the 5D algorithm as a function of the normalized average separation between the targets in Scenario II with orthogonal 2D sensor planes ($\theta = 90^\circ$). The different curves show the results for covariance matrices, Σ_C, obtained by rotating the matrix diag[16, 1, 1] about the y-axis through different angles, ϕ. (b) Average execution time in milliseconds for $N = 25, 50$, and 100 targets.



(a) $\Sigma_C = \text{diag}[16, 1, 1]$ **(b)** $\Sigma_C = \text{diag}[1, 16, 1]$ **(c)** $\Sigma_C = \text{diag}[1, 1, 16]$

(d) $\Sigma_C = \text{diag}[1, 16, 16]$ **(e)** $\Sigma_C = \text{diag}[16, 1, 16]$ **(f)** $\Sigma_C = \text{diag}[16, 16, 1]$

Fig. 7. Normalized RMSE in the estimated target positions as a function of normalized average distance between targets in Scenario I with $\theta = 90^\circ$ and $\phi = 0^\circ$. The results shown are for position estimates obtained after matching data from two sensors, A and C, using Munkres' algorithm (blue line with crosses), from all three sensors using the 4D algorithm (red lines with circles), and from all three sensors using the 5D algorithm (black lines with diamonds). The results with cigar-shaped covariance ellipsoids for Sensor C are shown in the top row, and those with pancake-shaped ellipsoids are shown in the bottom row.

F. Execution Time of Algorithms

In Fig. 6b, we show how the execution time of the 5D algorithm scales as the number of targets and the normalized average separation between the targets are varied. For this study, we implemented the algorithm in MATLAB on a MacBook Pro laptop with a 2.5 GHz Intel Core I5 processor. The three curves in the plot were obtained using $N = 25, 50$, and 100 targets. Each curve shows the average execution time (in milliseconds) of the 5D algorithm as a function of the normalized average separation between the targets. The increase in the computational time as the normalized average separation decreases is due to the fact that Munkres' algorithm requires more iterations to obtain a two-way match when the targets are more closely spaced. Indeed, we found that the number of iterations of the 5D algorithm is largely independent of the normalized average separation. The mean of the ratio of the average execution times of the 5D and 4D algorithms, taken over all values of the average separation, was 2.1 for 25 targets, 2.3 for 50 targets, and 3.2 for 100 targets. Since the 4D algorithm involves two applications of Munkres' algorithm, for 50 targets it is therefore about 4.6 times as expensive to obtain an assignment for the 322-sensor system using the 5D algorithm as it is for the 32-sensor system using Munkres' algorithm.

VII. DISCUSSION

In the previous sections, we used idealized sensor models to investigate how the performance of 322- and 32-sensor systems depends on the relative alignment of the 2D sensors and on the orientation of the covariance matrix of the 3D sensor. In this section, we first discuss the extent to which the theory we developed can be generalized to more realistic 3D radar and 2D camera models. Then, we summarize the main results from Section VI and interpret them in the context of more realistic systems.

Let $\mathbf{x} = (x_1, x_2, x_3)$ denote the position of a target in $\mathbb{R}^3$. In place of Sensor A, we consider a camera located at the origin of $\mathbb{R}^3$ pointing in the x_3-direction. We model camera measurements using a perspective projection as

$$(\widehat{w}_1, \widehat{w}_2) = \widehat{\mathbf{w}} = \psi_{2D}(\mathbf{x}) + \mathbf{u} = \alpha \left(\frac{x_1}{x_3}, \frac{x_2}{x_3} \right) + (u_1, u_2) \tag{42}$$

where α is the focal length and $\mathbf{u} \sim MN(\mathbf{0}, \Sigma_{2\times 2})$. In place of Sensor C, we consider a 3D radar located at $\mathbf{y} \in \mathbb{R}^3$ that makes measurements [16]

$$\widehat{\mathbf{z}} = \psi_{3D}(\mathbf{x} - \mathbf{y}) + \mathbf{v} \tag{43}$$

where ψ_{3D} is the rectangular to spherical coordinate transformation and $\mathbf{v} \sim MN(\mathbf{0}, \Sigma_{3\times 3})$. As in (11), we

transform the 3D radar measurement to the measurement plane of the camera via

$$\widehat{\boldsymbol{\xi}} \;=\; \Psi(\widehat{\mathbf{z}}) \tag{44}$$

where $\Psi = \psi_{2\mathrm{D}} \circ \psi_{3\mathrm{D}}^{-1}$ is the transformation from the space of radar measurements to the camera plane.

Since the main advantage of a third sensor is to resolve cluttered scenes [7], it is reasonable to assume that the targets are confined to a sufficiently small region, $\mathcal{R}$, of $\mathbb{R}^3$. Then

$$\widehat{\boldsymbol{\xi}} \;\approx\; \psi_{2\mathrm{D}}(\mathbf{x}-\mathbf{y}) + T_{\mathbf{x}}\mathbf{v} \tag{45}$$

where $T_{\mathbf{x}}$ is the 2×3 matrix of the linearization of Ψ at $\psi_{3\mathrm{D}}(\mathbf{x}-\mathbf{y})$. If $\bar{\mathbf{x}}$ denotes a nominal center for $\mathcal{R}$, which can be determined from radar measurements, then $T_{\mathbf{x}} \approx T_{\bar{\mathbf{x}}}$, which is independent of the particular target. As in (13), we take the difference of the measurements (42) and (45) in the camera plane to obtain

$$\widehat{\boldsymbol{\xi}} - \widehat{\mathbf{w}} \approx \psi_{2\mathrm{D}}(\mathbf{x}-\mathbf{y}) - \psi_{2\mathrm{D}}(\mathbf{x}) + T_{\bar{\mathbf{x}}}\mathbf{v} - \mathbf{u}. \tag{46}$$

For aerospace and defense applications, we may assume that $\mathcal{R}$ is at a large distance from the camera, so that $x_3 = \bar{x}_3 + \Delta x_3$, with $\Delta x_3 \ll \bar{x}_3$. Therefore,

$$(w_1, w_2) = \psi_{2\mathrm{D}}(x_1, x_2, x_3) \approx \frac{\alpha}{\bar{x}_3}(x_1, x_2) \tag{47}$$

and so as in (13), we can eliminate the unknown target location, $\mathbf{x}$, to obtain

$$\widehat{\boldsymbol{\xi}} - \widehat{\mathbf{w}} \approx -(y_1, y_2) + T_{\bar{\mathbf{x}}}\mathbf{v} - \mathbf{u}. \tag{48}$$

Similarly, in place of the compatibility condition (18) for Sensors A and B, we have that

$$\frac{\bar{x}_3}{\alpha_A}\widehat{w}_1^A \;-\; \frac{\bar{x}_2}{\alpha_B}\widehat{w}_1^B \;=\; \frac{\bar{x}_3}{\alpha_A}u_1^A \;-\; \frac{\bar{x}_2}{\alpha_B}u_1^B. \tag{49}$$

Using (48) and (49) and a formula for $T_{\bar{\mathbf{x}}}$, we can then compute replacements for the covariance matrices, $\Sigma_{4\mathrm{D}}$ in (15) and $\Sigma_{5\mathrm{D}}$ in (21). In this manner, the 4D and 5D algorithms can be extended to the scenario described above in which the targets are confined to a sufficiently small region far from the sensors. Moreover, the results in Section VI can also be applied to this scenario. These results can be summarized as follows.

First, in Section VI-B we showed that whenever the measurement uncertainty of the 3D sensor is large compared to that of the 2D sensors along the common axis of the planes of the 2D sensors, the percentage of correct two-way matches for the 322-sensor system with the 5D algorithm is significantly better than that of a 32-sensor system with the 2D algorithm. Under the same operating conditions, the percentage of correct three-way matches for the 322-sensor system with the 5D algorithm is significantly better than that with the 4D algorithm. With other relative sensor orientations, the percentage of correct matches was mostly independent of the choice of algorithm.

Second, with the aid of the bottom row of Fig. 5 we can make some conclusions about the performance of a 322-sensor system that includes a 3D radar with higher range resolution than angular resolution. Here, we also suppose that the 2D cameras have better angular resolution than the radar, and that the targets are confined to a small region, $\mathcal{R}$, as above. In this context, we refer to the vector $\bar{\mathbf{x}} - \mathbf{y}$ as the pointing direction of the radar. By the pointing direction of a camera, we mean the normal vector to the camera plane. We first consider a scenario in which all three sensors are mounted on the same platform with one of the cameras pointing in the same direction as the radar. In this scenario, the percentage of correct three-way matches decreases as the pointing direction of the second camera is lined up with that of the first (see Fig. 5f). On the other hand, for the scenario in Fig. 5d, the percentage of correct three-way matches is largely independent of the angle, θ, between the pointing directions of the cameras, and is moreover somewhat greater than for the scenarios in Fig. 5e and f. The covariance matrix that yields the largest percentage of correct three-way matches, i.e., that in Fig. 5d, is given by $\Sigma_C = \mathrm{diag}[1, 16, 16]$. This orientation of Sensor C corresponds to a scenario in which the radar is pointing along the $+x$-axis, perpendicular to the pointing directions of both cameras. Since all three sensors are observing the same small region, $\mathcal{R}$, this configuration can only be achieved if the sensors are viewing the targets from different locations.

Finally, for the above scenario in which the pointing direction of the radar is perpendicular to the pointing directions of the cameras, the results in Fig. 7d and e show that the RMSE in the estimated target positions is unacceptably large for a 32-sensor system. This is so because the large uncertainty in the radar measurements in the pointing direction of the camera cannot be resolved by the camera measurement. However, with the addition of the second camera, the RMSE performance is greatly improved.

Because we compare sensor measurements to sensor measurements rather than sensor measurements to unknown target locations, it is unlikely that the algorithms we developed can be extended to deal with missed detections as in [7] and [16]. Nevertheless, if we were to use the approach of [7], the above conclusions should still hold with an unknown number of targets, missed detections, and spurious measurements. Indeed, we would expect the performance gains for the 322-sensor to be even greater in this situation and/or when employing the Lagrangian relaxation assignment algorithm [17] in place of the 5D algorithm.

VIII. CONCLUSION

We used idealized sensor models and a special-purpose assignment algorithm for the M2MA problem to investigate the potential advantages of target tracking based on a 322-sensor system rather than a 32-sensor system. Depending on the relative alignment of the 2D sensors and the orientation of the covariance ellipsoid

of the 3D sensor, we found that position estimates obtained with the 322-sensor system can be much more accurate than those obtained with a 32-sensor system. Furthermore, we identified cases in which the percentage of correct assignments was significantly larger for the 322-sensor with the 5D algorithm than for the 32-sensor with Munkres' algorithm. Finally, we discussed applications of our results to specific scenarios based on realistic 3D radar and 2D camera models.

The overarching conclusion from our simulation study is that the performance of a 32- or 322-sensor system in which the covariance ellipsoid of the 3D sensor sensor is sufficiently aspherical can exhibit a strong dependence on the relative directions from which the sensors view the scene. Moreover, provided that the sensors are appropriately aligned, there can be a significant performance advantage to a target tracking system with two cameras and a radar as opposed to one camera and a radar.

ACKNOWLEDGMENTS

The authors thank Cody Harmon and Joseph Burnett for illuminating conversations and the reviewers for their suggestions that significantly improved this paper.

REFERENCES

[1] Y. Bar-Shalom and W. D. Blair
Multitarget–Multisensor Tracking: Applications and Advances, vol. III. Norwood, MA: Artech House, 2000.

[2] Y. Bar-Shalom and X.-R. Li
Multitarget–Multisensor Tracking: Principles and Techniques. Storrs, CT: YBS Publishing, 1995.

[3] Y. Bar-Shalom and R. W. Osborne
"Data association,"
in *Encyclopedia of Systems and Control*, J. Baillieul and T. Samad
Eds. Berlin, Germany: Springer, 2015, pp. 251–259.

[4] Y. Bar-Shalom, P. Willet, and X. Tian
Tracking and Data Fusion. Storrs, CT: YBS Publishing, 2011.

[5] D. P. Bertsekas
"Auction algorithms for network flow problems: A tutorial introduction,"
Comput. Optim. Appl., vol. 1, no. 1, pp. 7–66, 1992.

[6] S. Blackman and R. Popoli
Design and Analysis of Modern Tracking Systems. Norwood, MA: Artech House, 1999.

[7] S. Deb, K. Pattipati, and Y. Bar-Shalom
"A multisensor–multitarget data association algorithm for heterogeneous sensors,"
IEEE Trans. Aerosp. Electron. Syst., vol. 29, no. 2, pp. 560–568, Apr. 1993.

[8] S. Deb, M. Yeddanapudi, K. Pattipati, and Y. Bar-Shalom
"A generalized *S*-D assignment algorithm for multisensor–multitarget state estimation,"
IEEE Trans. Aerosp. Electron. Syst., vol. 33, no. 2, pp. 523–538, Apr. 1997.

[9] S. Herman and A. Poore
"Nonlinear least-squares estimation for sensor and navigation biases,"
in *Signal and Data Processing of Small Targets 2006*, vol. 6236. Bellingham, WA: International Society for Optics and Photonics, 2006, p. 623617.

[10] R. Jonker and A. Volgenant
"A shortest augmenting path algorithm for dense and sparse linear assignment problems,"
Computing, vol. 38, no. 4, pp. 325–340, 1987.

[11] W. Koch
Tracking and Sensor Data Fusion: Methodological Framework and Selected Applications. Berlin, Germany: Springer, 2016.

[12] F. Meyer *et al.*
"Message passing algorithms for scalable multitarget tracking,"
Proc. IEEE, vol. 106, no. 2, pp. 221–259, Feb. 2018.

[13] J. Munkres
"Algorithms for the assignment and transportation problems,"
J. Soc. Ind. Appl. Math., vol. 5, no. 1, pp. 32–38, 1957.

[14] C. H. Papadimitriou and K. Steiglitz
Combinatorial Optimization: Algorithms and Complexity. Chelmsford, MA: Courier Corporation, 1982.

[15] A. Petrovskaya and S. Thrun
"Model based vehicle detection and tracking for autonomous urban driving,"
Auton. Robots, vol. 26, no. 2–3, pp. 123–139, 2009.

[16] A. B. Poore
"Multidimensional assignment formulation of data association problems arising from multitarget and multisensor tracking,"
Comput. Optim. Appl., vol. 3, no. 1, pp. 27–57, 1994.

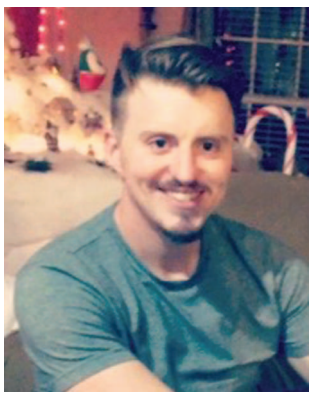
[17] A. B. Poore and A. J. Robertson III
"A new Lagrangian relaxation based algorithm for a class of multidimensional assignment problems,"
Comput. Optim. Appl., vol. 8, no. 2, pp. 129–150, 1997.

[18] D. Smith and S. Singh
"Approaches to multisensor data fusion in target tracking: A survey,"
IEEE Trans. Knowl. Data Eng., vol. 18, no. 12, pp. 1696–1710, Dec. 2006.

[19] G. Thomaidis, L. Spinoulas, P. Lytrivis, M. Ahrholdt, G. Grubb, and A. Amditis
"Multiple hypothesis tracking for automated vehicle perception,"
in *Proc. Intell. Veh. Symp*. Piscataway, NJ: IEEE, 2010, pp. 1122–1127.

[20] Z. Wu, N. I. Hristov, T. H. Kunz, and M. Betke
"Tracking-reconstruction or reconstruction-tracking? Comparison of two multiple hypothesis tracking approaches to interpret 3D object motion from several camera views,"
in *Proc. Workshop Motion Video Comput*. Piscataway, NJ: IEEE, 2009, pp. 1–8.

César Contreras received the Bachelor's degree and the Master's degree in electronics engineering from Nuevo Laredo Institute of Technology, Nuevo Laredo, Tamaulipas, Mexico, in 2000 and 2004, respectively, and the Master's degree in mathematics from Texas A&M International University, Laredo, TX, USA, in 2014. He held several positions in the industry and is currently working toward the Ph.D. degree in applied mathematics from The University of Texas at Dallas. His research interests include control theory, nonlinear dynamical systems, electromagnetic theory, and semiconductor physics.



John Langford received the B.S. degree in physics from Texas A&M University, College Station, TX, USA, in 2011, and the M.S. degree in physics from The University of Texas at Dallas, Richardson, TX, USA, in 2014. He is currently working toward the Ph.D. degree in statistics at The University of Texas at Dallas. His current research interests include applications of deep learning in genomics.

Larry Ammann received the B.A. degree in physics and the Ph.D. degree in statistics from Florida State University, Tallahassee, FL, USA, in 1969 and 1976, respectively. He is currently a Professor of Statistics with the Department of Mathematical Sciences, The University of Texas at Dallas, Richardson, TX, USA. His research interests include robust multivariate statistical methods, signal processing, statistical computing, applied probability, and remote sensing.

John Zweck received the B.Sc. degree with honors from the University of Adelaide, Adelaide, Australia, in 1988, and the Ph.D. degree in mathematics from Rice University, Houston, TX, USA, in 1993. He has performed research in differential geometry, computer vision, mathematical modeling of engineering systems, and nonlinear waves. He is currently a Professor with the Department of Mathematical Sciences, The University of Texas at Dallas, Richardson, TX, USA. From 2006 to 2012, he was an Associate Professor of Mathematics and Statistics and an Affiliate Associate Professor of Computer Science and Electrical Engineering with the University of Maryland Baltimore County (UMBC). From 2003 to 2006, he was an Assistant Professor with the Department of Mathematics and Statistics, UMBC, and from 2000 to 2003, he was a Research Associate with the Department of Computer Science and Electrical Engineering, UMBC. He is a member of the Society for Industrial and Applied Mathematics.

Brian Marks received the B.S. degree in physics and mathematics from North Carolina State University, Raleigh, NC, USA, in 1995, and the Ph.D. degree in engineering sciences and applied mathematics from Northwestern University, Evanston, IL, USA, in 2000. He is currently a senior member of the technical professional staff at the Johns Hopkins University Applied Physics Laboratory, Laurel, MD, USA, where he has been working since 2013. He previously held academic positions at University of Maryland, Baltimore County and at Indiana University Bloomington. His research interests include applications of mathematics and statistics in the domains of engineering and physics, with recent focus on remote sensing, signal processing, tracking, and multisensor integration.