Practical Data Association for Passive Sensors in 3D

SHUO ZHANG
YAAKOV BAR-SHALOM

This paper considers the passive-sensor data association problem based on multi-dimensional assignment (MDA), a prerequisite for data fusion. The S-D algorithm has been shown to be effective for solving the MDA problem. The bottleneck of the S-D algorithm lies in its cost computation, which consumes about 95%–99% of the CPU times. Since the number of costs in the MDA problem increases exponentially with the number of sensors, the S-D algorithm becomes quite inefficient when a large number of sensors are used. We propose an efficient data association technique, "S_0-D+Seq(2-D)" algorithm, which decomposes the original problem to an S_0-dimensional assignment and several 2-dimensional assignments. The S_0-D+Seq(2-D) algorithm yields a total number of costs which only increases quadratically with the number of sensors. Simulation results show that the S_0-D+Seq(2-D) algorithm achieves a significant reduction in CPU time compared to the S-D algorithm with similar association qualities.

Manuscript received July 25, 2011; revised May 31, 2012 and November 11, 2013; released for publication March 7, 2014.

Refereeing of this contribution was handled by Ramona Georgescu.

Authors' addresses: Shuo Zhang, United Technologies Research Center, East Hartford, CT 06108, E-mail: (zhangs1@utrc.utc.com); Yaakov Bar-Shalom, ECE Department, University of Connecticut, Storrs, CT 06269-2157, U.S.A. E-mail: (shuo.zhang@engr.uconn.edu).

1557-6418/14/$17.00 © 2014 JAIF

1. INTRODUCTION

Data association is a crucial task in many surveillance systems and is a prerequisite for data fusion. In general data association solves the correspondence problem in either a "hard" or "soft" manner [2]. A typical step in tracking problems is the measurement-to-track association where it decides which measurement to update which track. There are several state-of-the-art methods in solving the type of association, for example, Joint Probabilistic Data Association (JPDA) and Multiple Hypothesis Tracking (MHT) [2]. In this paper, we consider another type of association called measurement-to-measurement association in a multisensor mutlitarget scenario, where each sensor generates a set of line of sight (LOS) or direction of arrival (DOA), i.e., incomplete position measurements of the targets and the goal is to decide which of the measurements in each sensor correspond to the same target. The measurements are grouped together by an association algorithm and are used to generate a composite (full position) measurement of a common target. In tracking applications, the composite measurements can be used in the subsequent measurement-to-track association to update existing tracks (this is fusion configuration III [2]). This measurement-to-measurement association is considered as "static" where the measurements are assumed to be synchronized, i.e., observed at the same time. The fusion of asynchronous measurements is discussed in [13].

Measurement-to-measurement association becomes especially challenging if the sensors are passive and measure LOS angles only for the targets. Measurements from multiple sensors have to be associated to determine the full positions of the targets. The brute force approach, i.e., enumerating all possible combinations and choosing the most likely one, is computationally prohibitive even for a moderate size problem. For example, the total number of combinations for a scenario of 20 targets and 2 sensors (assuming no missed detections or false alarms) is $20! = 2.4 \times 10^{18}$. A practical approach is to formulate the multisensor data association as a multiple dimensional assignment (MDA) problem [2] and then employ (constrained) optimization techniques to obtain the optimal assignment. When the number of sensors is greater than or equal to three (i.e., $S \geq 3$), the MDA is known to be NP hard. While a number of suboptimal techniques have been proposed, the Lagrangian relaxation based approaches [14], [16] have been shown to be superior to others (e.g., branch and bound, row-column heuristic) for their excellent balance between the accuracy and the efficiency. The relaxation technique in [9] is termed as the S-D (assignment) algorithm. In [18] an extended approach of determining the top m assignments (as opposed to only the best one) has been obtained by using Murty's ranking algorithm [10].

Prior to the optimization step in the S-D algorithm, the first step is to calculate the candidate association costs. It has been reported [20], [1] that this cost-calculation step consumes 95%–99% of the CPU time. Consequently, when the number of targets is large, a direct use of the S-D algorithm can become quite inefficient. Thus, for the large-scale problem clustering techniques [8] are applied before carrying out the S-D algorithm. By employing the clustering (or partitioning), the original large problem is reduced to a number of smaller subproblems, which can be solved efficiently by the S-D algorithm.

However, even with the clustering, the CPU time of the S-D algorithm increases drastically when a large number of sensors are used. The reason is that the number of costs to be computed increases exponentially with the number of sensors, although this computational burden can be mitigated by employing the gating technique [2], [7] to remove unlikely candidate associations. Note that it is not uncommon to have a large number of passive sensors since some of them have lower costs and are easier for deployment, e.g., infrared or CCD cameras. Aiming at the large-scale data association, that is, when a large number of targets are present and many sensors are used, we propose an efficient data association technique: "S_0-D+Seq(2-D)" algorithm. This algorithm first decomposes the original problem to a (fixed) S_0-dimensional assignment and $S - S_0$ 2-dimensional assignments. Then the former is solved by using the S-D algorithm and the latter is solved by a successive use of the (modified) Auction algorithm [14].

The paper is organized as follows. In Section 2 the MDA problem is formulated for passive sensors in 3D. In Section 3 the iterative least squares (ILS) technique is presented for target position estimation using the LOS measurements. In Section 4 the dihedral-angle based clustering technique is discussed. The proposed S_0-D+Seq(2-D) algorithm is given in Section 5. Simulation results are given in Section 6 based on a large-scale localization problem. Finally, conclusions are given in Section 7.

2. FORMULATION OF THE MDA PROBLEM

Assume that there are S passive sensors in a 3D space, with known positions $\mathbf{p}_s = [x_s, y_s, z_s]'$ $(s = 1,\ldots,S)$. The sensors are assumed to be synchronized.[1] For a given target, each sensor provides its LOS measurement, azimuth angle and elevation angle, namely,

$$z_{i_s} = h(\mathbf{x},\mathbf{p}_s) + w_s \qquad s = 1,\ldots,S \tag{1}$$

where i_s is the measurement index in sensor s, $\mathbf{x} = [x,y,z]'$ denotes the target's position, w_s is zero-mean white Gaussian measurement noise with covariance R_s and

$$h(\mathbf{x},\mathbf{p}_s) = \begin{bmatrix} \alpha_s \\ \varepsilon_s \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\dfrac{y - y_s}{x - x_s}\right) \\ \tan^{-1}\left(\dfrac{z - z_s}{\sqrt{(x-x_s)^2 + (y-y_s)^2}}\right) \end{bmatrix} \tag{2}$$

Each sensor may receive a number of such measurements from multiple targets, as well as false alarms. An S-tuple of measurements $Z_{i_1 i_2 \ldots i_S}$, consisting of one measurement from each sensor, represents a possible association, that is, the measurements $Z_{i_1 i_2 \ldots i_S}$ are assumed to originate from the same target. Since a target may not be detected by every sensor, a *dummy measurement* is added to each sensor with index 0, to represent the missed detection. If there is only one nondummy measurement in a S-tuple, this nondummy measurement is deemed to be a false alarm.[2] For each S-tuple there is an associated cost $c_{i_1 i_2 \ldots i_S}$, which is given by the negative log-likelihood ratio [2]

$$c_{i_1 i_2 \ldots i_S} = -\ln \frac{\Lambda(Z_{i_1 i_2 \ldots i_S} \mid \mathbf{x})}{\Lambda(Z_{i_1 i_2 \ldots i_S} \mid \emptyset)} \tag{3}$$

The numerator in (3) represents the likelihood that the S-tuple of measurements $Z_{i_1 i_2 \ldots i_S}$ originate from the same target with position $\mathbf{x}$, namely,

$$\Lambda(Z_{i_1 i_2 \ldots i_S} \mid \mathbf{x}) = \prod_{s=1}^{S} [1 - P_{Ds}]^{1-u(i_s)} [P_{Ds} p(z_{i_s} \mid \mathbf{x})]^{u(i_s)} \tag{4}$$

where P_{Ds} is the detection probability of sensor s, $u(i_s)$ is an indicator function, defined as

$$u(i_s) = \begin{cases} 0 & \text{if } i_s = 0 \\ 1 & \text{otherwise} \end{cases} \tag{5}$$

and $p(z_{i_s} \mid \mathbf{x})$ is given by

$$p(z_{i_s} \mid \mathbf{x}) = |2\pi R_s|^{-1/2} \cdot \exp(-\tfrac{1}{2}[z_{i_s} - h(\mathbf{x},\mathbf{p}_s)]' R_s^{-1} [z_{i_s} - h(\mathbf{x},\mathbf{p}_s)]) \tag{6}$$

The denominator in (3) represents the likelihood that all the measurements in the S-tuple are false alarms, namely,

$$\Lambda(Z_{i_1 i_2 \ldots i_S} \mid \emptyset) = \prod_{s=1}^{S} \lambda^{u(i_s)} \tag{7}$$

where λ denotes the spatial density [2] of the false alarms.

The MDA problem is formulated as follows [9]

$$\min_{\rho_{i_1 i_2 \ldots i_S}} \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \cdots \sum_{i_S=0}^{n_S} c_{i_1 i_2 \ldots i_S} \rho_{i_1 i_2 \ldots i_S} \tag{8}$$

[1]This corresponds to Configuration III Fusion, following the classification originated by O. E. Drummond (see [2]).

[2]We make the assumption that a target is detected by at least two sensors, otherwise the target's state is unobservable.

subject to

$$\sum_{i_2=0}^{n_2}\cdots\sum_{i_S=0}^{n_S}\rho_{i_1i_2\ldots i_S}=1;\qquad i_1=1,2,\ldots,n_1$$
$$\sum_{i_1=0}^{n_1}\cdots\sum_{i_S=0}^{n_S}\rho_{i_1i_2\ldots i_S}=1;\qquad i_2=1,2,\ldots,n_2$$
$$\vdots\qquad\qquad\vdots$$
$$\sum_{i_1=0}^{n_1}\cdots\sum_{i_{S-1}=0}^{n_{S-1}}\rho_{i_1i_2\ldots i_S}=1;\qquad i_S=1,2,\ldots,n_S \tag{9}$$

where $\rho_{i_1i_2\ldots i_S}\in\{0,1\}$. Thus, the goal is to find $\{\rho_{i_1i_2\ldots i_S}\}$, i.e., a partition of the total measurements that minimizes the global cost, subject to the constraints that each measurement is associated with one and only one measurement (including the dummy measurement) in each other sensor. When $S=2$, this MDA problem can be solved exactly by using the modified Auction algorithm [14]. In the general case, i.e., $S>2$, this problem is NP hard and can only be solved suboptimally. The S-D algorithm [9], which is based on the Lagrangian relaxation, has been shown to be an effective technique to solve this general MDA problem.

3. POSITION ESTIMATION VIA ITERATIVE LEAST SQUARES

Since the target position $\mathbf{x}$ in (4) is unknown, it is substituted by its estimate $\hat{\mathbf{x}}$ obtained from the S-tuple of measurements $Z_{i_1i_2\ldots i_S}$. While there are a number of methods to obtain $\hat{\mathbf{x}}$, the iterative least squares (ILS) technique [3] is preferred since it is easy to implement (no Hessian involved) and provides a (approximate) covariance matrix for its estimate at the same time.

Assume that there are n nondummy measurements in $Z_{i_1i_2\ldots i_S}$ $(2\le n\le S)$ and we stack them to form an augmented vector $\mathbf{z}$. Then, the ILS estimate in the jth iteration can be written as

$$\hat{\mathbf{x}}^{j+1}=\hat{\mathbf{x}}^{j}+[(H^j)'R^{-1}H^j]^{-1}\cdot(H^j)'R^{-1}[\mathbf{z}-\mathbf{h}(\hat{\mathbf{x}},\mathbf{p})] \tag{10}$$

where $R=\mathrm{diag}([R_1,\ldots,R_n])$,[3] $\hat{\mathbf{z}}=[\hat{z}_1',\ldots,\hat{z}_n']'$, $\mathbf{h}(\hat{\mathbf{x}},\mathbf{p})=[\mathbf{h}(\hat{\mathbf{x}},\mathbf{p}_1)',\ldots,\mathbf{h}(\hat{\mathbf{x}},\mathbf{p}_n)']'$ and

$$H^j=\left.\frac{\partial\mathbf{h}(\hat{\mathbf{x}},\mathbf{p})}{\partial\mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}^j} \tag{11}$$

is the Jacobian matrix of the stacked measurement vector evaluated at $\hat{\mathbf{x}}^j$. In this case, the Jacobian matrix is

$$H=[H_1'\cdots H_n']' \tag{12}$$

[3]The subscript in R_i is the index of the nondummy measurement and is not the sensor index. It is different from that of the previous section. This holds for other variables.

where

$$H_i=\begin{bmatrix}\dfrac{\partial\alpha_i}{\partial x} & \dfrac{\partial\alpha_i}{\partial y} & \dfrac{\partial\alpha_i}{\partial z}\\[2mm] \dfrac{\partial\varepsilon_i}{\partial x} & \dfrac{\partial\varepsilon_i}{\partial y} & \dfrac{\partial\varepsilon_i}{\partial z}\end{bmatrix} \tag{13}$$

$$\frac{\partial\alpha_i}{\partial x}=-\frac{y-y_i}{(x-x_i)^2+(y-y_i)^2} \tag{14}$$

$$\frac{\partial\alpha_i}{\partial y}=\frac{x-x_i}{(x-x_i)^2+(y-y_i)^2} \tag{15}$$

$$\frac{\partial\alpha_i}{\partial z}=0 \tag{16}$$

$$\frac{\partial\varepsilon_i}{\partial x}=-\frac{(x-x_i)(z-z_i)}{\sqrt{(x-x_i)^2+(y-y_i)^2}\|\mathbf{x}-\mathbf{p}_i\|^2} \tag{17}$$

$$\frac{\partial\varepsilon_i}{\partial y}=-\frac{(y-y_i)(z-z_i)}{\sqrt{(x-x_i)^2+(y-y_i)^2}\|\mathbf{x}-\mathbf{p}_i\|^2} \tag{18}$$

$$\frac{\partial\varepsilon_i}{\partial z}=\frac{\sqrt{(x-x_i)^2+(y-y_i)^2}}{\|\mathbf{x}-\mathbf{p}_i\|^2} \tag{19}$$

and $\|\cdot\|$ denotes the Euclidean norm.

To start the ILS recursion an inital estimate $\hat{\mathbf{x}}^0$ is required, which is given by [12]

$$\hat{x}^0=\frac{y_2-y_1+x_1\tan\alpha_1-x_2\tan\alpha_2}{\tan\alpha_1-\tan\alpha_2} \tag{20}$$

$$\hat{y}^0=\frac{\tan\alpha_1(y_2+\tan\alpha_2(x_1-x_2))-y_1\tan\alpha_2}{\tan\alpha_1-\tan\alpha_2} \tag{21}$$

$$\hat{z}^0=z_1+\tan\varepsilon_1\left|\frac{(y_1-y_2)\cos\alpha_2+(x_2-x_1)\sin\alpha_2}{\sin(\alpha_1-\alpha_2)}\right| \tag{22}$$

which has made use of the first two measurements.

4. CLUSTERING

In the S-D algorithm, the most expensive step is computing the association costs, which consumes 95%–99% of the CPU time [20], [1]. From (8) the total number of costs to be calculated is

$$n_c=\prod_{s=1}^{S}(n_s+1) \tag{23}$$

For simplicity, assuming that the number of measurements is n_0 for every list (sensor), then

$$n_c=(n_0+1)^S \tag{24}$$

Consequently, a large n_0 is unfavorable for the efficiency of the S-D algorithm.

The clustering technique is used to reduce a large-size problem to a number of smaller subproblems, which can be solved independently. The clustering algorithm groups measurements based on a distance metric. For the passive sensors in 3D, an effective metric is the so-called dihedral angle [8]. The dihedral angle is defined as the angle between two planes, a target plane and

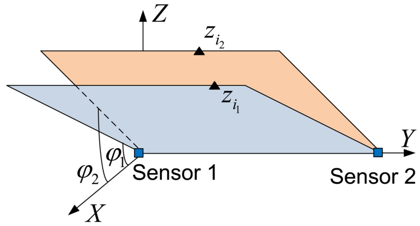


Fig. 1. Clustering using dihedral angles.

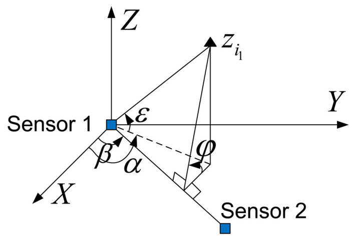


Fig. 2. Computation of the dihedral angle.

TABLE 1
Clustering Using Dihedral Angles

FOR sensor $s = 1 : S - 1$
    FOR sensor $j = s + 1 : S$
        Cluster measurements in sensors s and j using the dihedral angles
    END FOR
    Find the measurements that have not yet been clustered for the use of the next iteration
END FOR

a reference plane (see Fig. 1). The target plane passes through two sensors and one LOS measurement from either of these two sensors, while the reference plane is the XY plane in the 3D Cartesian space (assuming both of the sensors are located on the XY plane). Given two LOS measurements from two different sensors, if these measurements originate from the same target, then the two dihedral angles, one for each LOS measurement, would be close to each other. As a result, clustering the dihedral angles leads to clustering the respective LOS measurements.

The dihedral angle φ for a LOS measurement $[\alpha,\varepsilon]'$ from sensor 1 located at the origin with reference to another sensor 2 located on the same plane as sensor 1, but at an angle β to it (see Fig. 2) is given by

$$\varphi = \tan^{-1}\left(\frac{\tan\varepsilon}{\sin\Delta}\right) \tag{25}$$

where

$$\Delta = \mathrm{mod}(|\alpha-\beta|,\pi) \tag{26}$$

Eq. (26) denotes the angle $|\alpha-\beta|$ with a modulus of π.

The dihedral angles have to be computed in pairwise between each pair of sensors. A summary of the clustering algorithm using the dihedral angles is given in Table 1.

TABLE 2
Numbers of Costs of the S-D algorithm for Different Numbers of Sensors ($n_0 = 7$)

No. of sensors S	No. of costs n_c
4	4,096
7	2,097,152
10	1,073,700,000

The dihedral angle can be also utilized in gating [2] to prune unlikely associations. If a candidate association fails in the gating test, there is no need to compute its cost, i.e., an infinitely large cost is assigned to it. For a given cluster, the calculation of the candidate costs is recursive. Beginning at z_{i_1} in list 1, we take one measurement from each list at a time. If the measurement falls inside the gate defined by the previous measurements in the tuple, this measurement is incorporated in the tuple, which advances to the next list. The cost of the tuple is only evaluated at the last list when a full tuple is achieved. For example, assuming the current list is m and the current association tuple is $Z_{i_1\ldots i_{m-1}}$, if z_{i_m} passes the gating test then it is added and form $Z_{i_1\ldots i_m}$, otherwise all the subsequent candidate associations starting with $Z_{i_1\ldots i_m}$ are discarded. Consequently, the CPU time spent in the cost computation can be saved via this gating process.

REMARK I: While the clustering technique significantly reduces the association complexity, the downside is that some measurements can be grouped incorrectly. Hence the use of clusters poses a design trade-off between complexity and accuracy. Note that the association algorithm to be presented next is not restricted to this clustering technique. It can be integrated with any clustering algorithm for passive sensors in 3D.

5. S_0-D+Seq(2-D) ALGORITHM

For a small number of sensors, the S-D algorithm (along with the clustering technique if the number of targets is large) is able to perform in real time. However, when the number of the sensors increases, the CPU time of the S-D algorithm increases drastically, which is impossible for the S-D algorithm to operate in real time. We can see from (24) the total number of costs increases exponentially with S. For example, assuming n_0 in (24) is 7 (in each cluster we expect a small number of n_0), the numbers of costs for the S-D algorithm are shown in Table 2 for different numbers of sensors.

When $S = 10$, the total number of costs is over one billion[4] for a single cluster.

We propose an efficient data association technique, called S_0-D+Seq(2-D) algorithm, for the case where more than 3 sensors are used. This algorithm consists of two steps, the S_0-D step and the Seq(2-D) step, which are presented next.

[4]The actual number of costs to be computed would be smaller due to gating.

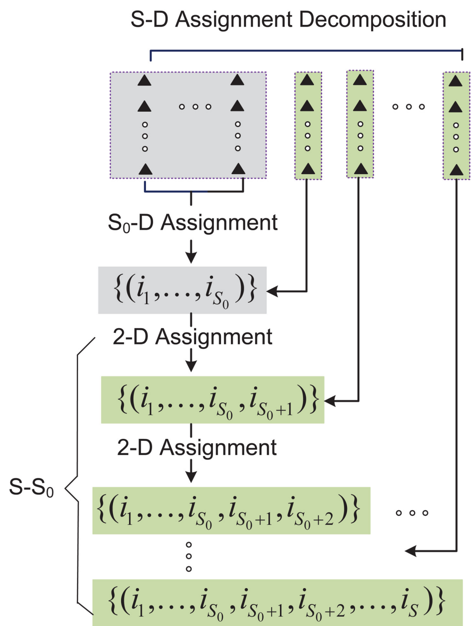


Fig. 3. An illustration of the S_0-D+Seq(2-D) algorithm.

TABLE 3
S_0-D+Seq(2-D) Algorithm

1) S_0-D step:
Solve the S_0-D assignment using the S-D algorithm and obtain the S_0-tuple association results $\{(i_1,\ldots,i_{S_0})\}$.
2) Seq(2-D) step:
FOR $n = S_0 + 1 : S$
Construct the 2-D assignment between the previous association results $\{(i_1,\ldots,i_{n-1})\}$ and the measurements $\{z_{i_n}\}$ in list n;
Solve the 2-D assignment using the modified Auction algorithm and obtain the n-tuple results $\{(i_1,\ldots,i_n)\}$.
END FOR

TABLE 4
Numbers of Costs of the S_0-D+Seq(2-D) algorithm for Different Numbers of Sensors ($n_0 = 7$)

No. of sensors S	No. of costs n'_c
4	688
7	1,888
10	4,096

S_0-D step: This steps solves a standard MDA problem with the dimension of S_0 using the S-D algorithm, that is, the data association is performed among S_0 sensors ($S_0 < S$). While the minimum value of S_0 is 2, practically S_0 should be at least 3 to achieve quality associations. This is due to the ghosting problem [2] of the passive sensors. The use of more sensors can mitigate this ghosting effect. A large gate (or no gating) is recommended for the S_0-D step to prevent discarding some less likely (due to noises) but real associations.

Seq(2-D) step: This step solves a series of 2D assignments sequentially using the modified Auction algorithm [14]. The number of the 2D assignments is $S - S_0$. After the S_0-D step, the S_0-tuple association results are available. Then, take a new list from the remaining $S - S_0$ lists and formulate a 2-D assignment between the previous association results and the measurements in this new list. After the 2-D assignment, the length of each association is incremented by one, i.e., becoming a $S_0 + 1$-tuple. Next, take another list from the remaining $S - S_0 - 1$ lists and solve another 2-D assignment, and so on. In the end, after carrying out $S - S_0$ 2-D assignments, each association is in a full tuple, i.e., a S-tuple. The S_0-D+Seq(2-D) algorithm is summarized in Table 3, assuming the S_0-D step chooses the first S_0 lists, i.e., $s = 1,2,\ldots,S_0$. An illustration of the S_0-D+Seq(2-D) algorithm is shown in Fig. 3.

Similarly to (24), the number of costs in the S_0-D step is $(n_0 + 1)^{S_0}$. In the Seq(2-D) step, among the $S - S_0$ 2-D assignments the largest number of costs occur at the last 2-D assignment, which in the worst case is $((S-1)n_0 + 1)(n_0 + 1)$. Consequently, an upper bound of the total number of costs of the S_0-D+Seq(2-D) algorithm is given by

$$n'_c = (n_0 + 1)^{S_0} + (S - S_0)(n_0 + 1)((S-1)n_0 + 1) \quad (27)$$

which increases quadratically with S. In Table 4 the values of this upper bound are shown for different number of sensors.

The association quality of this S_0-D+Seq(2-D) algorithm is evaluated next in the simulation results.

6. SIMULATION RESULTS

We consider a localization problem using the LOS measurements. The numbers of the passive sensors used are 4, 7 and 10. The sensors are located in a circle of radius 5 km centered at (5,5) km in the XY plane, with equal angle separations. The measurement noise standard deviation is 1 mrad for both azimuth and elevation angle. All the sensors are assumed to have the same accuracy, detection probability P_D ($= 0.98$) and false alarm rate P_F ($= 10^{-5}$, which corresponds to an average 15 false alarms for each sensor). The number of targets is 300 and their positions are randomly placed in the 3D Cartesian space, where the ranges of the X, Y, Z coordinates are 0–10 km, 1–10 km, 5–10 km, respectively. There is no prior information assumed for the number of targets. The sensor-target geometry is shown in Fig. 4 for the case of 10 sensors.

Given an association tuple, if there is only one nondummy measurement, then it is deemed to represent a false alarm, otherwise it falls into one of the following 3 categories (similar to [1]):

1. Completely correct (CC) association: The measurements in an association tuple have identical origin and there is no dummy measurement associated.
2. Partially correct (PC) association: There are at least 2 measurements with common origin, and the rest may be from different origins or dummy measurements.

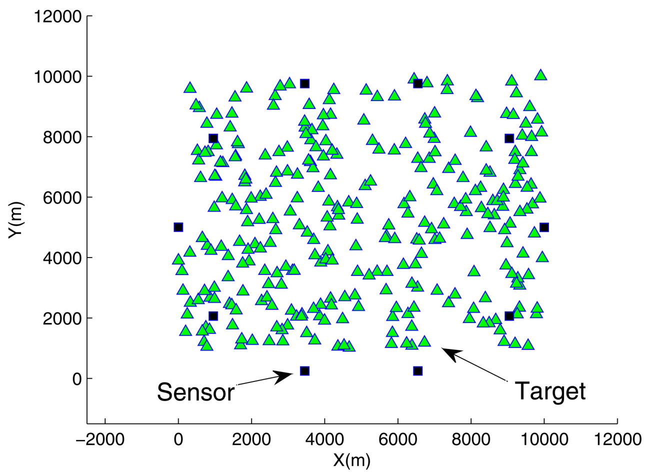


Fig. 4. 300 Targets and 10 sensors (XY projection shown).

3. Completely incorrect (CI) association: In an association tuple, there does not exit a pair of measurements that come from the same origin.

Each CC or PC association corresponds to a detected target (DT). The DT is defined as the origin that appears most in a CC or PC association tuple, and the number of times that the DT appears in a tuple is referred to as the detection index (DI). The detected targets are a subset of the total targets (TT).

Given an association tuple, if the number of the non-dummy measurements is no less than a threshold T_H ($T_H > 1$), then this association is accepted, otherwise it is rejected. To quantify the quality of the accepted associations, we introduce four metrics: fraction of correct associations, fraction of missed targets, fraction of duplicated associations and fraction of purity, which are defined below[5]

- Fraction of correct associations (FCA): $\mathrm{FCA} = \dfrac{N_{CC} + N_{PC}}{N_{CC} + N_{PC} + N_{CI}}$
- Fraction of missed targets (FMT): $\mathrm{FMT} = \dfrac{N_{TT} - N_{DT}}{N_{TT}}$
- Fraction of duplicated associations (FDA): $\mathrm{FDA} = \dfrac{N_{CC} + N_{PC} - N_{DT}}{N_{DT}}$
- Fraction of purity (FP): $\mathrm{FP} = \dfrac{\overline{DI}}{S}$

where $\overline{DI}$ denotes the average detection index. Note that only N_{TT} is independent of the threshold T_H.

For the comparison, we consider the method that solves the S dimensional assignment directly, i.e., the S-D algorithm and the proposed sequential method, S_0-D+Seq(2-D), where $S_0 = 3$. The case of $S_0 = 2$ is also evaluated in the simulations. Also, we examine the sequential m-best assignment algorithm, referred to as $m[S_0$-D]+Seq(m[2-D]), where for each S_0-D and 2-D assignment, the top m-best assignments (instead of the only best one) are computed by using the Murty's ranking algorithm [10], [5]. Efficient implementations of the Murty's algorithm can be found in [11], [17], [18]. The $m[S_0$-D]+Seq(m[2-D]) algorithm is performed as follows. For each one of the m solutions obtained from the previous step, the m-best assignment algorithm is carried out, which yields m^2 solutions. Then, m best ones are selected out of these m^2 solutions and stored for the use in the next step. A simplified version, designated as $m[S_0$-D]+Seq(2-D), is also considered in which the m-best assignment is carried out only once for the initial S_0-D assignment to obtain the m best solutions, and for each one of them, the sequential 2-D assignment is used for subsequent associations. All the algorithms are coded in C++ and run on a Intel i7 2.70 GHz laptop. The results are based on 20 Monte Carlo runs.[6]

The dihedral angle based clustering technique from Section 4 is employed. The dihedral angle gating is used when incorporating a new measurement to a association tuple which has been validated before. If the new measurement passes the gate, the resulting new tuple is valid with the tuple length incremented by one, otherwise the new tuple is discarded, i.e., the whole (association) subtree starting with this new tuple is pruned out. The parameters are chosen as: $T_H = 3$, $m = 4$. Although the detection probability is 0.98, a larger value of P_D ($P_D < 1$) is used in (4). This is due to a phenomenon to be called "association splitting," in which a CC or PC association is divided into two or more PC associations, which provide an overall lower cost. A similar phenomenon was observed in [1] for track-to-track associations. This splitting phenomenon will result in incomplete associations.[7] The use of a larger (pseudo) P_D will penalize incomplete associations and prevent the association from splitting.

The CPU times and association qualities for different algorithms are compared in Table 5, where the location RMSE (averaged over all the correct associations) is also provided. The S_0-D+Seq(2-D) algorithm (S_0-D with sequential 2-D), which is, as discussed below, the preferred one, is shown in boldface. The CPU time of the S-D algorithm increases drastically with the number of sensors, S. When $S = 10$, the S-D algorithm requires too much memory that exceeds the computer capacity and no results were obtained. The S_0-D+Seq(2-D) algorithm (S_0-D with sequential 2-D), discussed in Section 5, is advantageous when a large number of sensors are used. When $S = 7$, S_0-D+Seq(2-D) reduces the CPU time of S-D by three orders of magnitude with little difference in association qualities. In terms of either correct

[5] N_X represents the number of X, e.g., N_{DT} denotes the number of detected targets.

[6] This is because a single run of S-D on 7 lists/sensors took half an hour while for 10 lists it became infeasible.

[7] For example, an association of measurements from sensors {1,2,3} and one from {4,5,6,7} are found "cheaper" (when all of them have the same origin) than associating all of them together.

TABLE 5
Comparison of different algorithms

No. of sensors	Algorithm	FCA	FMT	FDA	FP	RMSE (m)	CPU Time (s)
4	S-D	97.1%	4.8%	3.8%	90.9%	37.6	9.6
4	**S_0-D+Seq(2-D)**	**98.3%**	**6.6%**	**4.1%**	**93.1%**	**39.7**	**2.8**
4	Seq(2-D)	98.6%	10.0%	3.9%	88.7%	42.4	2.0
4	$m[S_0$-D]+Seq(m[2-D])	97.3%	4.1%	4.2%	92.5%	37.2	8.9
4	m[2-D]+Seq(m[2-D])	98.0%	4.6%	4.1%	92.8%	37.4	9.6
4	$m[S_0$-D]+Seq(2-D)	97.3%	4.1%	4.2%	92.5%	37.2	7.3
4	m[2-D]+Seq(2-D)	97.6%	4.7%	4.1%	92.7%	37.9	6.9
7	S-D	98.2%	3.6%	6.3%	80.1%	33.9	1316.8
7	**S_0-D+Seq(2-D)**	**97.6%**	**5.2%**	**4.5%**	**86.2%**	**39.8**	**9.9**
7	Seq(2-D)	97.4%	6.6%	4.2%	83.1%	42.7	9.2
7	$m[S_0$-D]+Seq(m[2-D])	96.8%	2.7%	4.9%	86.7%	35.0	61.1
7	m[2-D]+Seq(m[2-D])	97.7%	3.5%	3.9%	87.1%	35.6	61.8
7	$m[S_0$-D]+Seq(2-D)	96.8%	3.0%	4.5%	86.9%	35.1	36.4
7	m[2-D]+Seq(2-D)	96.8%	4.0%	4.4%	87.0%	36.4	35.4
10	S-D	—	—	—	—	—	—
10	**S_0-D+Seq(2-D)**	**96.8%**	**4.8%**	**5.3%**	**85.9%**	**40.3**	**20.2**
10	Seq(2-D)	95.8%	5.7%	4.9%	83.3%	44.4	19.5
10	$m[S_0$-D]+Seq(m[2-D])	96.1%	2.4%	5.5%	86.7%	35.1	169.4
10	m[2-D]+Seq(m[2-D])	97.2%	3.1%	4.1%	87.3%	36.0	169.2
10	$m[S_0$-D]+Seq(2-D)	96.2%	2.6%	5.2%	86.8%	35.5	75.9
10	m[2-D]+Seq(2-D)	96.3%	3.6%	5.0%	86.9%	37.7	76.4

associations or duplicated associations (FCA and FDA), which one of S-D and S_0-D+Seq(2-D) is better depends on the number of sensors used. The S-D algorithm has fewer missed targets (FMT) and smaller RMSE, while the S_0-D+Seq(2-D) algorithm yields purer associations (FP). With the increase of the number of sensors, both the missed targets and the purities decline for both S-D and S_0-D+Seq(2-D) algorithms. Computationally, Seq(2-D) is the least expensive. S_0-D+Seq(2-D) consumes more CPU time than Seq(2-D) (for large n the time difference is negligible), however, it outperforms the latter in terms of FMT, FP and RMSE. For instance, when $n = 4$ the FMT values imply that S_0-D+Seq(2-D) reduces the missed targets of Seq(2-D) by more than 30%. This is obtained at the cost of extra 40% in CPU time. For large n the improvement is 20% for an extra CPU time of 5–8%.

When $S = 4$, the $m[S_0$-D]+Seq(m[2-D]) algorithm (m-best S_0-D with sequential m-best 2-D) has the same association qualities as the $m[S_0$-D]+Seq(2-D) algorithm (m-best S_0-D with sequential 2-D). For $S = 7$ and $S = 10$, $m[S_0$-D]+Seq(m[2-D]) outperforms $m[S_0$-D]+Seq(2-D) in terms of FMT and RMSE, but not by a large margin. The corresponding CPU times show that $m[S_0$-D]+Seq(m[2-D]) is more expensive than $m[S_0$-D]+Seq(2-D). The CPU time of $m[S_0$-D]+Seq(2-D) is approximately proportional to m. The m[2-D]+Seq(m[2-D]) algorithm consumes similar CPU time as the $m[S_0$-D]+Seq(m[2-D]) algorithm, however, the former has degraded performance in terms of FMT and RMSE. The same situation occurs for the $m[S_0$-D]+Seq(2-D) and m[2-D]+Seq(2-D) algorithms. It is also observed that the (sequential) m-best algorithms slightly outperform the (sequential) algorithms that choose the single best solution in FMT and RMSE, but require more CPU time.

REMARK II: Similarly to the n-scan pruning approach [4] used in the dynamic association, one can also apply a sequential S_0-dimensional assignment to this static association problem, that is, solve the S_0-D assignment on sensors $1,\ldots,S_0$, then make a hard decision on sensor 1 and solve the S_0-D assignment on $2,\ldots,S_0+1$, etc. However, compared to S_0-D+Seq(2-D), at each step the S_0-D assignment (assuming $S_0 > 2$) is more costly than the 2-D assignment for both cost evaluations and optimization. In terms of association performance, from the above results we can see that even the S-D assignment is similar to S_0-D+Seq(2-D), thus the possible improvement of the sequential S_0-D assignment over S_0-D+Seq(2-D) is quite limited.

REMARK III: Although the performance using more sensors appears no better (or worse for some metrics) than using fewer sensors (as the generation of a valid association tuple becomes more demanding), from the robustness point of view the use of more sensors is always beneficial.

7. CONCLUSIONS

This paper presented an efficient data association technique, S_0-D+Seq(2-D) algorithm, for passive sensors in 3D. The passive-sensor data association is a challenging problem, since the line of sight (LOS) measurements from the passive sensors only provide partial knowledge of a target position. The assignment-based methods have been shown to be very effective for data association, where the data association is first formulated as a multiple dimension assignment (MDA) problem and then solved (suboptimally) by using the Lagrangian-relaxation based S-D algorithm. The bottleneck of the S-D algorithm lies in the cost computation, which consumes about 95%–99% of the CPU times. The number of costs to be evaluated in the MDA problem increases exponentially with the number of lists (sensors), which renders the S-D algorithm quite inefficient when a large number of sensors are used. The proposed S_0-D+Seq(2-D) algorithm has a total number of costs increasing quadratically with the number of sensors. As a result, it reduces the number of costs drastically in comparison with the S-D algorithm. For 7 sensors the S_0-D+Seq(2-D) algorithm achieves a CPU time reduction of 3 orders of magnitude compared to the S-D algorithm. The CPU time can be further reduced by introducing parallelization to process different clusters concurrently. The S-D and S_0-D+Seq(2-D) algorithms have similar association qualities. A good choice of S_0 has been shown to be 3. The (sequential) m-best algorithms slightly outperform the (sequential) non m-best algorithms, but are more costly. It has also been shown that the $m[S_0$-D]+Seq(2-D) algorithm is preferred over the $m[S_0$-D]+Seq(m[2-D]) algorithm.

REFERENCES

[1] J. Areta, Y. Bar-Shalom, M. Levedahl and K. R. Pattipati
Hierarchical tracking association and fusion for a networked surveillance system.
Journal of Advances in Information Fusion, 1(2):140–156, Dec. 2006.

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

[3] Y. Bar-Shalom, X. R. Li and T. Kirubarajan
Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software.
New York: Wiley, 2001.

[4] S. S. Blackman
Multiple Hypothesis Tracking For Multiple Target Tracking.
IEEE Aerospace and Electronic Systems Magazine, 19(1):5–18, Jan. 2004.

[5] I. J. Cox and M. Miller
On finding ranked assignments with application to multitarget tracking and motion.
IEEE Transactions on Aerospace and Electronic Systems, 31(1):486–489, Jan. 1995.

[6] I. J. Cox and S. L. Hingorani
An effcient implementation of reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 18(2):138–150, Feb. 1996.

[7] J. B. Collins and J. K. Uhlmann
Efficient gating in data association with multivariate Gaussian states.
IEEE Transactions on Aerospace and Electronic Systems, 28(3):909–916, July 1992.

[8] M. R. Chummun, T. Kirubarajan, K. R. Pattipati and Y. Bar-Shalom
Fast Data Association Using Multidimensional Asignment With Clustering.
IEEE Transactions on Aerospace and Electronic Systems, 37(3), July 2001.

[9] S. Deb, M. Yeddanapudi, K. R. Pattipati and Y. Bar-Shalom
A generalized S-dimensional assignment for multisensor-multitarget state estimation.
IEEE Transactions on Aerospace and Electronic Systems, 33(3):523–538, Apr. 1997.

[10] K. Murty
An algorithm for ranking all the assigments in order of increasing cost.
Operations Research, 16(3):682–687, 1968.

[11] M. L. Miller, H. S. Stone and I. J. Cox
Optimizing Murty's ranked assignment method.
IEEE Transactions on Aerospace and Electronic Systems, 33(3):851–862, July 1997.

[12] R. W. Osborne, III and Y. Bar-Shalom
Statistical efficiency of composite position measurements from passive sensors.
IEEE Transactions on Aerospace and Electronic Systems, 49(4):2799–2806, Oct. 2013.

[13] R. W. Osborne, III, Y. Bar-Shalom and P. Willett
Fusion of Asynchronous Passive Measurements.
17th International conference on information fusion, Salamanca, Spain, July 2014.

[14] K. R. Pattipati, S. Deb, Y. Bar-Shalom and R. B. Washburn
A new relaxation algorithm and passive sensor data association.
IEEE Transactions on Automatic Control, 37(2):198–213, Feb. 1992.

[15] A. B. Poore, N. Rijavec, T. Barker and M. Munger
Data association problems posed as multidimensional assignment problems: algorithm development.
Proc. SPIE Conf. on Signal Processing, Sensor Fusion and Target Recognition, 1955:175–183, 1993.

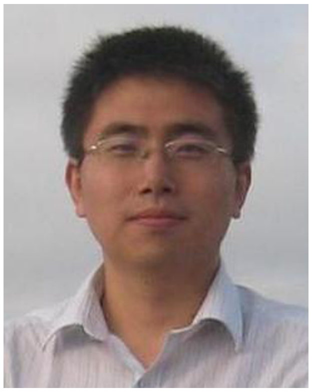
[16] A. B. Poore
Multidimensional Assignment formulation of data association problems arising from multitarget and multisensor tracking.
Computational Optimization and Applications, 3:27–57, 1994.

[17] R. Popp, K. R. Pattipati and Y. Bar-Shalom
An m-Best 2-D assignment algorithm and multilevel parallelization.
Transactions on Aerospace and Electronic Systems, 35(4): 1145–1160, Oct. 1999.

[18] R. L. Popp, K. R. Pattipati and Y. Bar-Shalom
m-best S-D assignment algorithm with application to multitarget tracking.
IEEE Transactions on Aerospace and Electronic Systems, 37(1):22–34, Jan. 2001.

[19] J. K. Uhlmann
Algorithms for multiple-target tracking.
American Scientist, 80:128–141, Mar. 1992.

[20] H. Wang, T. Kirubarajan and Y. Bar-Shalom
Precision large scale air traffic surveillance using IMM/assignment estimators.
IEEE Transactions on Aerospace and Electronic Systems, 35(1):255–266, Jan. 1999.



Shuo Zhang was born in 1982. He received the B.E. degree in 2005 and M.E. degree in 2007 from Harbin Institute of Technology and Ph.D. degree from The University of Connecticut in 2011, all in Electrical and Computer Engineering. His research interests include estimation, perception and convex optimization. Currently He is a Senior Research Scientist at United Technologies Research Center (UTRC). He has published 7 journal papers and more than 10 peer-reviewed conference papers.

Yaakov Bar-Shalom was born on May 11, 1941. He received the B.S. and M.S. degrees from the Technion, Israel Institute of Technology, in 1963 and 1967 and the Ph.D. degree from Princeton University in 1970, all in electrical engineering.

From 1970 to 1976 he was with Systems Control, Inc., Palo Alto, California. Currently he is Board of Trustees Distinguished Professor in the Dept. of Electrical and Computer Engineering and Marianne E. Klewin Professor in Engineering at the University of Connecticut. He is also Director of the ESP (Estimation and Signal Processing) Lab.

His current research interests are in estimation theory, target tracking and data fusion. He has published over 500 papers and book chapters in these areas and in stochastic adaptive control. He coauthored the monograph *Tracking and Data Association* (Academic Press, 1988), the graduate texts *Estimation and Tracking: Principles, Techniques and Software* (Artech House, 1993), *Estimation with Applications to Tracking and Navigation: Algorithms and Software for Information Extraction* (Wiley, 2001), the advanced graduate texts *Multitarget-Multisensor Tracking: Principles and Techniques* (YBS Publishing, 1995), *Tracking and Data Fusion* (YBS Publishing, 2011), and edited the books *Multitarget-Multisensor Tracking: Applications and Advances* (Artech House, Vol. I, 1990; Vol. II, 1992; Vol. III, 2000).

He has been elected Fellow of IEEE for "contributions to the theory of stochastic systems and of multi-target tracking." He has been consulting to numerous companies and government agencies, and originated the series of Multitarget-Multisensor Tracking short courses offered via UCLA Extension, at Government Laboratories, private companies and overseas.

During 1976 and 1977 he served as Associate Editor of the *IEEE Transactions on Automatic Control* and from 1978 to 1981 as Associate Editor of *Automatica*. He was Program Chairman of the 1982 American Control Conference, General Chairman of the 1985 ACC, and Co-Chairman of the 1989 IEEE International Conference on Control and Applications. During 1983–87 he served as Chairman of the Conference Activities Board of the IEEE Control Systems Society and during 1987–89 was a member of the Board of Governors of the IEEE CSS. He was a member of the Board of Directors of the International Society of Information Fusion (1999–2004) and served as General Chairman of FUSION 2000, President of ISIF in 2000 and 2002 and Vice President for Publications in 2004–13.

In 1987 he received the IEEE CSS Distinguished Member Award. Since 1995 he is a Distinguished Lecturer of the IEEE AESS and has given numerous keynote addresses at major national and international conferences. He is co-recipient of the M. Barry Carlton Award for the best paper in the *IEEE Transactions on Aerospace and Electronic Systems* in 1995 and 2000 and recipient of the 1998 University of Connecticut AAUP Excellence Award for Research. In 2002 he received the J. Mignona Data Fusion Award from the DoD JDL Data Fusion Group. He is a member of the Connecticut Academy of Science and Engineering. In 2008 he was awarded the IEEE Dennis J. Picard Medal for Radar Technologies and Applications, and in 2012 the Connecticut Medal of Technology. He has been listed by academic.research.microsoft (top authors in engineering) as #1 among the researchers in Aerospace Engineering based on the citations of his work.