A New Heterogeneous Track Fusion with Information Decorrelation Algorithm for Target Tracking in a Multistatic Sensor System with Non-Cooperative Moving Transmitters

RONG YANG
YAAKOV BAR-SHALOM

This paper considers a target tracking problem in a non-cooperative multistatic system, where several transmitters are moving and their positions are unknown. The receiver listens to the signals from non-cooperative transmitters via direct and indirect (bouncing from targets) paths. The transmitters and targets are then tracked based on the measured bearings and the bistatic ranges (derived from the time difference of arrival of the direct and indirect path signals) simultaneously. In previous work, we proved that the transmitter trajectories are observable when the two transmitters are not located on the same line from the receiver, and developed an approximate algorithm to perform estimation based on covariance inflation (CI). In this paper, a new estimation algorithm, heterogeneous track fusion with information decorrelation (HTF-D), is developed. It aims to achieve optimal estimates without using a large augmented state consisting of all transmitter and target states. The approach tracks targets individually, and fuses these highly correlated tracks through a novel information decorrelation method. The performance of the HTF-D is evaluated through simulation tests. The results show that the HTF-D provides better estimates than the CI algorithm, and achieves the same accuracy as the optimal algorithm when the latter does not suffer from numerical problems due to its large augmented state. The HTF-D estimates are also consistent statistically.

Manuscript received January 2, 2020; revised May 4, 2020; released for publication June 30, 2020.

Refereeing of this contribution was handled by Florian Meyer.
R. Yang is with the DSO National Laboratories, 12 Science Park Drive, Singapore 118225, Singapore (E-mail: yrong@dso.org.sg). Y. Bar-Shalom is with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269, USA (E-mail: yaakov.bar-shalom@uconn.edu).

I. INTRODUCTION

Passive detection and tracking is desirable in surveillance systems as it estimates target trajectories through "listening" to the signals emitted by others without any emission of its own. It can therefore avoid being detected by the targets it observes. This paper will develop a new passive tracking algorithm primarily for passive sonar and radar applications.

Passive tracking algorithms operate according to two basic approaches. One is to listen to target emission signals and then to estimate target trajectories. The techniques such as bearings-only tracking (BOT) and Doppler-bearing tracking (DBT) belong to this category [1]–[8]. The target motion state (comprising the target position and velocity) is estimated by measuring the target bearings and, if their emissions are narrowband, also their Doppler shifted frequencies. Since the target range cannot be measured, the conventional BOT/DBT [1], [3], [5], [6] usually cannot provide accurate trajectory estimates, even when the platform of the receiver maneuvers. The recently developed unscented Gauss–Helmert filter [2], [4], [7], [8] fuses two types of target signals (e.g., an acoustic signal and an electromagnetic signal) with different propagation delays, so that a target range can be inferred from the delays. This method increases the estimation accuracy significantly. However, it is only applicable to the targets that emit these heterogeneous signals. Another approach to passive tracking is by means of the multistatic sensor concept, which does not require any emission from targets. The receiver (listener) listens to some transmitters' signals through direct and indirect (bouncing from targets) paths. The targets can then be tracked based on the measured bearings and bistatic ranges. The bistatic ranges are derived from the time difference of arrival (TDOA) of the direct and indirect path signals. This paper will present a generalization of the multistatic approach, detailed in the sequel.

The multistatic tracking has been extensively studied. The international Multistatic Tracking Working Group, organized in 2005, dedicated research to this topic. Their research results were summarized in [9]. The algorithms they used include Bayesian tracking, distributed multiple hypothesis tracking, probabilistic multiple hypothesis tracking, maximum likelihood probabilistic data association algorithm, and Gaussian mixture cardinalized probability hypothesis density tracking. All of them solve the tracking problem for the conventional multistatic system where the positions of transmitters and receivers are assumed known. However, the transmitters' positions are not always known, especially for moving transmitters, which do not inform the receiver of their positions in a timely manner. We designate this type of transmitters as non-cooperative transmitters. To widen the application of multistatic systems, research on non-cooperative transmitters is of special interest. In general, the moving target trajectories are not observable based on the measurements from a

non-cooperative transmitter only. Additional information needs to be provided to estimate the transmitter positions. A well-known method is to estimate the transmitter position from objects with known positions (e.g., landmarks, rocks, or wrecks) [10]–[12]. The environmental information such as walls and other reflectors can also be utilized [13]–[15].[1] Once the transmitter position is known, the target trajectories are observable in a multistatic system. A study on non-cooperative transmitter system without using additional information was also conducted recently [16]. It estimates the positions of targets and a transmitter simultaneously purely from the measurements. However, this approach is only applicable for stationary targets.

This paper focuses on a new non-cooperative multistatic configuration [17], [18] that can track transmitters and targets simultaneously, where targets and transmitters can be either moving or stationary, and no additional information such as reflections from known objects and environment is required. The only requirement is that the number of non-cooperative transmitters must be greater than or equal to 2. This configuration was first proposed in [17]. The observability of the problem was proved when the two transmitters are not located on the same line of sight from the receiver (see Appendix A). Simulation tests were conducted using synchronous transmitters for proof of concept. Further study [18] extended the algorithm from synchronous transmitters to the more realistic asynchronous case. This is because non-cooperative transmitters do not negotiate each other to synchronize their emission times. The covariance inflation (CI) filter [19] was used for transmitter and target trajectory estimation in [18] and its estimation results were compared to the optimal algorithm that estimates the augmented state consisting of all transmitter and target states. Unfortunately, the CI filter, being an approximate approach, has a significant accuracy gap to the optimal algorithm. The optimal algorithm is straightforward but, unfortunately, has the following issues:

1) The state size will be large when many targets exist. This will cause the estimation to be more sensitive to the system errors, and it is likely to have numerical problems.
2) The state size varies with the number of targets in the system. This complicates the real system applications.
3) The system may face scalability issues as all target and transmitter estimation is handled by one estimator.

Thus, a better tracking algorithm is needed.

[1]Strictly speaking, the systems addressed in these references are not multistatic systems since they consider only transmitters and receivers. Non-emitting targets are not there. However, the approaches for the transmitter/receiver location estimation in these papers can be utilized in a non-cooperative multistatic system when the additional environmental information is available.

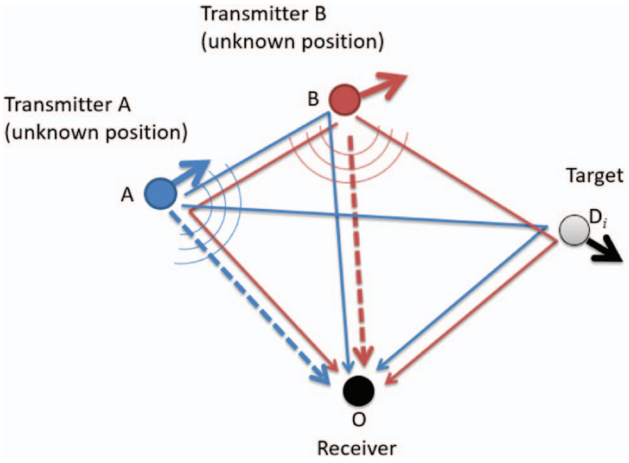

Fig. 1. Noncooperative multistatic system with two asynchronous transmitters with unknown positions. Transmitters A and B move and emit "blue" and "red" signals (on different frequencies), respectively. The dashed lines are the direct signal paths, and the solid lines are the indirect signal paths.

In this paper, we present a novel heterogeneous track fusion with information decorrelation (HTF-D) algorithm that provides similar accuracy to the optimal algorithm and estimates each target trajectory individually, so that the drawbacks of the optimal algorithm caused by the large state size are avoided. The technical challenge is that the cross-covariances among the target estimation errors cannot be addressed properly if we track the targets individually. The optimal algorithm estimates a large augmented state so that these cross-covariances can be computed easily. The HTF-D, tracking the targets individually, does not compute the cross-covariances between individual tracks. To overcome this, we develop a new *heterogeneous and correlated state fusion method* on top of the individual tracks, so that the cross-covariances can be handled properly through the fusion.

The rest of paper is organized as follows. Section 2 describes the problem formulation. Section 3 reviews the existing estimation algorithms relevant to our problem. Section 4 describes the HTF-D algorithm. Section 5 shows the simulation results, and conclusions are given in Section 6.

II. PROBLEM FORMULATION

The problem is illustrated in Fig. 1. A stationary receiver[2] is located at the origin of the Cartesian coordinate system $O(0, 0)$. Two moving platforms A and B carry emitters that emit asynchronous pulse signals periodically with different frequencies (represented in red and blue). One moving target (without emitter) reflects the signals to the receiver. Note that only one target is shown in Fig. 1 for simplicity. The approach is

[2]The approaches described in the paper are not limited to a stationary receiver; they can also work on a moving receiver, assuming the trajectory of the receiver is known.

applicable to multiple targets as shown later in the simulation results. The positions of the transmitters and target are unknown to the receiver. The direct signal paths from the two transmitters are represented by dashed lines, while the indirect signal paths reflected via another transmitter or target are represented by solid lines. It is easy for the receiver to differentiate the direct and indirect signals from their amplitudes, as the direct signals are much stronger than the indirect signals [20].

The two transmitters emit signals asynchronously, based on their own schedules. The receiver needs to perform estimation when it receives signals from any one of transmitters.

The states to be estimated are

$$\mathbf{x}^{\mathrm{A}}(t_k) = [x^{\mathrm{A}}(t_k)\ \ y^{\mathrm{A}}(t_k)\ \ \dot{x}^{\mathrm{A}}(t_k)\ \ \dot{y}^{\mathrm{A}}(t_k)]', \tag{1}$$

$$\mathbf{x}^{\mathrm{B}}(t_k) = [x^{\mathrm{B}}(t_k)\ \ y^{\mathrm{B}}(t_k)\ \ \dot{x}^{\mathrm{B}}(t_k)\ \ \dot{y}^{\mathrm{B}}(t_k)]', \tag{2}$$

$$\mathbf{x}^{\mathrm{D}_i}(t_k) = \left[x^{\mathrm{D}_i}(t_k)\ \ y^{\mathrm{D}_i}(t_k)\ \ \dot{x}^{\mathrm{D}_i}(t_k)\ \ \dot{y}^{\mathrm{D}_i}(t_k)\right]', \tag{3}$$

where $\mathbf{x}^{\mathrm{A}}$ and $\mathbf{x}^{\mathrm{B}}$ are the states of the transmitters A and B, respectively, $\mathbf{x}^{\mathrm{D}_i}$ is the target state with $i = 1, \ldots, n$, and n is the number of targets in the system. The state transition model assuming a nearly constant velocity (NCV) model is

$$\mathbf{x}^*(t_k) = \mathbf{F}(T_{k,k-1})\mathbf{x}^*(t_{k-1}) + \mathbf{v}^*(T_{k,k-1}), \tag{4}$$

where $\boxtimes^*$ represents $\boxtimes^{\mathrm{A}}$, $\boxtimes^{\mathrm{B}}$, or $\boxtimes^{\mathrm{D}_i}$,

$$\mathbf{F}(T_{k,k-1}) = \begin{bmatrix} 1 & 0 & T_{k,k-1} & 0 \\ 0 & 1 & 0 & T_{k,k-1} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

with the time interval (not a constant in the asynchronous case)

$$T_{k,k-1} = t_k - t_{k-1}, \tag{6}$$

and $\mathbf{v}^*$ is the zero-mean process noise for the interval $T_{k,k-1}$. Based on the discretized continuous-time white noise acceleration model [21], its covariance is

$$E[\mathbf{v}^*(\cdot)\mathbf{v}^*(\cdot)'] = \mathbf{Q}(T_{k,k-1}) = \begin{bmatrix} \frac{T_{k,k-1}^3}{3} & 0 & \frac{T_{k,k-1}^2}{2} & 0 \\ 0 & \frac{T_{k,k-1}^3}{3} & 0 & \frac{T_{k,k-1}^2}{2} \\ \frac{T_{k,k-1}^2}{2} & 0 & T_{k,k-1} & 0 \\ 0 & \frac{T_{k,k-1}^2}{2} & 0 & T_{k,k-1} \end{bmatrix} q, \tag{7}$$

where q is the power spectral density (PSD) of the continuous-time (acceleration) process noise (same for x and y, and assumed independent between the coordinates). The measurement vectors are

$$\mathbf{z}^{\mathrm{A}}(t_k) = \begin{cases} b^{\mathrm{A}}(t_k), & \text{when A is emitting at } t_k, \\ [r^{\mathrm{AB}}(t_k)\ \ b^{\mathrm{A}}(t_k)]', & \text{when B is emitting at } t_k, \end{cases} \tag{8}$$

$$\mathbf{z}^{\mathrm{B}}(t_k) = \begin{cases} b^{\mathrm{B}}(t_k), & \text{when B is emitting at } t_k, \\ [r^{\mathrm{BA}}(t_k)\ \ b^{\mathrm{B}}(t_k)]', & \text{when A is emitting at } t_k, \end{cases} \tag{9}$$

$$\mathbf{z}^{\mathrm{D}_i}(t_k) = \begin{cases} [r^{\mathrm{D_iA}}(t_k)\ \ b^{\mathrm{D}_i}(t_k)]', & \text{when A is emitting at } t_k, \\ [r^{\mathrm{D_iB}}(t_k)\ \ b^{\mathrm{D}_i}(t_k)]', & \text{when B is emitting at } t_k, \end{cases} \tag{10}$$

where (see Fig. 1)

$$r^{\mathrm{AB}}(t_k) = |AO| + |AB| - |BO| + w_r^{\mathrm{AB}}(t_k), \tag{11}$$

$$r^{\mathrm{BA}}(t_k) = |BO| + |AB| - |AO| + w_r^{\mathrm{BA}}(t_k), \tag{12}$$

$$r^{\mathrm{D_iA}}(t_k) = |D_iO| + |AD_i| - |AO| + w_r^{\mathrm{D_iA}}(t_k), \tag{13}$$

$$r^{\mathrm{D_iB}}(t_k) = |D_iO| + |BD_i| - |BO| + w_r^{\mathrm{D_iB}}(t_k), \tag{14}$$

with

$$|AO| = \sqrt{x^{\mathrm{A}}(t_k)^2 + y^{\mathrm{A}}(t_k)^2}, \tag{15}$$

$$|BO| = \sqrt{x^{\mathrm{B}}(t_k)^2 + y^{\mathrm{B}}(t_k)^2}, \tag{16}$$

$$|D_iO| = \sqrt{x^{\mathrm{D}_i}(t_k)^2 + y^{\mathrm{D}_i}(t_k)^2}, \tag{17}$$

$$|AB| = \sqrt{[x^{\mathrm{A}}(t_k) - x^{\mathrm{B}}(t_k)]^2 + [y^{\mathrm{A}}(t_k) - y^{\mathrm{B}}(t_k)]^2}, \tag{18}$$

$$|AD_i| = \sqrt{[x^{\mathrm{A}}(t_k) - x^{\mathrm{D}_i}(t_k)]^2 + [y^{\mathrm{A}}(t_k) - y^{\mathrm{D}_i}(t_k)]^2}, \tag{19}$$

$$|BD_i| = \sqrt{[x^{\mathrm{B}}(t_k) - x^{\mathrm{D}_i}(t_k)]^2 + [y^{\mathrm{B}}(t_k) - y^{\mathrm{D}_i}(t_k)]^2}, \tag{20}$$

are the four bistatic ranges, and w_r^{AB}, w_r^{BA}, $w_r^{\mathrm{D_iA}}$, and $w_r^{\mathrm{D_iB}}$ are their measurement noises, assumed to be zero-mean white Gaussian with variance σ_r^2, and

$$b^*(t_k) = \tan^{-1}\left[\frac{x^*(t_k)}{y^*(t_k)}\right] + w_b^*(t_k) \tag{21}$$

are three bearings, where $\boxtimes^*$ represents $\boxtimes^{\mathrm{A}}$, $\boxtimes^{\mathrm{B}}$, or $\boxtimes^{\mathrm{D}_i}$, and w_b^* is the bearing measurement Gaussian error with variance σ_b^2.

III. REVIEW ON THE EXISTING APPROACHES

The three existing estimation approaches that can be applied to our problem are reviewed in this section. They are the optimal, simple, and CI algorithms. The review focuses on the dynamic state estimation. The parameter

estimation used for the track initiation is not included. Interested readers can find the track initiation details in [18].

A. The Optimal Algorithm

The optimal algorithm [18] is straightforward. It estimates the augmented state $\mathbf{x}^{\mathrm{ALL}}$ using the augmented measurement $\mathbf{z}^{\mathrm{ALL}}$, which are defined as

$$\mathbf{x}^{\mathrm{ALL}}(t_k) = [\mathbf{x}^{\mathrm{A}}(t_k)' \ \mathbf{x}^{\mathrm{B}}(t_k)' \ \mathbf{x}^{\mathrm{D}_1}(t_k)' \ \cdots \ \mathbf{x}^{\mathrm{D}_n}(t_k)']', \quad (22)$$

$$\mathbf{z}^{\mathrm{ALL}}(t_k) = [\mathbf{z}^{\mathrm{A}}(t_k)' \ \mathbf{z}^{\mathrm{B}}(t_k)' \ \mathbf{z}^{\mathrm{D}_1}(t_k)' \ \cdots \ \mathbf{z}^{\mathrm{D}_n}(t_k)']', \quad (23)$$

where the sizes of $\mathbf{x}^{\mathrm{ALL}}(t_k)$ and $\mathbf{z}^{\mathrm{ALL}}(t_k)$ are determined by the number of targets and transmitters in the system. The state transition model and measurement model are

$$\mathbf{x}^{\mathrm{ALL}}(t_k) = \mathbf{F}^{\mathrm{ALL}}(T_{k,k-1})\mathbf{x}^{\mathrm{ALL}}(t_{k-1}) + \mathbf{v}^{\mathrm{ALL}}(T_{k,k-1}), \quad (24)$$

$$\mathbf{z}^{\mathrm{ALL}}(t_k) = \mathbf{h}^{\mathrm{ALL}}[\mathbf{x}^{\mathrm{ALL}}(t_k)] + \mathbf{w}^{\mathrm{ALL}}(t_k), \quad (25)$$

where

$$\mathbf{F}^{\mathrm{ALL}}(T_{k,k-1}) = \begin{bmatrix} \mathbf{F}(T_{k,k-1}) & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{F}(T_{k,k-1}) \end{bmatrix}, \quad (26)$$

$\mathbf{h}^{\mathrm{ALL}}[\cdot]$ can be derived from (8)–(21), and $\mathbf{v}^{\mathrm{ALL}}$ and $\mathbf{w}^{\mathrm{ALL}}$ are the process noise and measurement noise, respectively.

The optimal algorithm performs estimation without losing any information. Thus, it can in theory obtain the best estimates. However, when the system is marginally observable, the augmented state is large, and the measurement noise is large, the optimal algorithm is inferior to the new algorithm (as it will be shown in Section 4) because the latter handles small states (each target state is estimated separately). The state estimate error covariance (time t_k is omitted for simplicity) for the optimal algorithm's augmented state is

$$\mathbf{P}^{\mathrm{ALL}} = \begin{bmatrix} \mathbf{P}^{\mathrm{A}} & \mathbf{P}^{\mathrm{AB}} & \mathbf{P}^{\mathrm{AD}_1} & \cdots & \mathbf{P}^{\mathrm{AD}_n} \\ \mathbf{P}^{\mathrm{BA}} & \mathbf{P}^{\mathrm{B}} & \mathbf{P}^{\mathrm{BD}_1} & \cdots & \mathbf{P}^{\mathrm{BD}_n} \\ \mathbf{P}^{\mathrm{D}_1\mathrm{A}} & \mathbf{P}^{\mathrm{D}_1\mathrm{B}} & \mathbf{P}^{\mathrm{D}_1} & \cdots & \mathbf{P}^{\mathrm{D}_1\mathrm{D}_n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}^{\mathrm{D}_n\mathrm{A}} & \mathbf{P}^{\mathrm{D}_n\mathrm{B}} & \mathbf{P}^{\mathrm{D}_n\mathrm{D}_1} & \cdots & \mathbf{P}^{\mathrm{D}_n} \end{bmatrix}. \quad (27)$$

The off-diagonal blocks are the cross-covariances among the transmitter and target state estimate errors. In this problem, the transmitter and target state estimate errors are all cross-correlated. The off-diagonal blocks are therefore nonzero.

Although the optimal algorithm can provide optimal solution in theory, it has three drawbacks given in Section 1. Thus, to track the targets individually (without large augmented state size), accounting for the coupling is preferred.

B. The Simple Algorithm

The simple algorithm [18] tracks the transmitters and targets individually in a completely decoupled manner. The state and measurement vectors are $\mathbf{x}^*(t_k)$ in (1)–(3) and $\mathbf{z}^*(t_k)$ in (8)–(10), respectively. The state transition model is as in (4), and the measurement model is (8)–(21). It can be seen that the bistatic ranges in (11)–(14) are functions of the target state and the emitting transmitter position (which can be either transmitter A or transmitter B based on who is emitting at time t_k). At a particular time t_k, the emitting transmitter state must be estimated first, so that the updated transmitter position can be used in the measurement model for estimating the other transmitter and the target states.

This approach ignores the transmitter state estimate errors [namely, $\mathbf{P}^{\mathrm{A}}$ and $\mathbf{P}^{\mathrm{B}}$ in (27)] in the other transmitter and target state estimation. Furthermore, since the transmitters and targets are estimated individually, the cross-covariances among estimate errors in the optimal algorithm [all the off-diagonal blocks in (27)] are ignored.

C. The CI Algorithm

This approach was used in [18] based on the CI method [19]. Similarly to the simple approach, the transmitter states are estimated first, and then the estimated transmitter positions are used to estimate the target states. Unlike the simple algorithm, it includes the transmitter position errors (assumed white, which they are not) in the measurement noise when performing target state estimation. Two CI filters were presented in [18]. One tracks the transmitters with augmented transmitter states, whereas another one tracks the transmitters individually. We will review the first one, as it yields better performance.

For the transmitter state estimation, the augmented transmitter state and measurement vectors are

$$\mathbf{x}^{\mathrm{AB}}(t_k) = [\mathbf{x}^{\mathrm{A}}(t_k)' \ \mathbf{x}^{\mathrm{B}}(t_k)']', \quad (28)$$

$$\mathbf{z}^{\mathrm{AB}}(t_k) = [\mathbf{z}^{\mathrm{A}}(t_k)' \ \mathbf{z}^{\mathrm{B}}(t_k)']', \quad (29)$$

respectively. The state transition model and measurement model are

$$\mathbf{x}^{\mathrm{AB}}(t_k) = \mathbf{F}^{\mathrm{AB}}(T_{k,k-1})\mathbf{x}^{\mathrm{AB}}(t_{k-1}) + \mathbf{v}^{\mathrm{AB}}(T_{k,k-1}), \quad (30)$$

$$\mathbf{z}^{\mathrm{AB}}(t_k) = \mathbf{h}^{\mathrm{AB}}[\mathbf{x}^{\mathrm{AB}}(t_k)] + \mathbf{w}^{\mathrm{AB}}(t_k), \quad (31)$$

where

$$\mathbf{F}^{\mathrm{AB}}(T_{k,k-1}) = \begin{bmatrix} \mathbf{F}(T_{k,k-1}) & \mathbf{0} \\ \mathbf{0} & \mathbf{F}(T_{k,k-1}) \end{bmatrix}. \quad (32)$$

The measurement function $\mathbf{h}^{\mathrm{AB}}[\cdot]$ can be derived from (8)–(21), and $\mathbf{v}^{\mathrm{AB}}$ and $\mathbf{w}^{\mathrm{AB}}$ are the process noise and measurement noise, respectively.

For the target estimation, the state and measurement vectors are $\mathbf{x}^{D_i}$ and $\mathbf{z}^{D_i}$ given in (3) and (10), respectively. The state transition model has been given in (4), and the measurement model can be derived from (10)–(21). The CI filter with the following steps is used for the dynamic target state estimation.

Prediction:

$$\hat{\mathbf{x}}^{D_i}(t_k|t_{k-1}) = \mathbf{F}(T_{k,k-1})\hat{\mathbf{x}}^{D_i}(t_{k-1}|t_{k-1}), \tag{33}$$

$$\mathbf{P}^{D_i}(t_k|t_{k-1}) = \mathbf{F}(T_{k,k-1})\mathbf{P}^{D_i}(t_{k-1}|t_{k-1})\mathbf{F}(T_{k,k-1})' + \mathbf{Q}^{D_i}(T_{k,k-1}). \tag{34}$$

Update:

$$\hat{\mathbf{x}}^{D_i}(t_k|t_k) = \hat{\mathbf{x}}^{D_i}(t_k|t_{k-1}) + \mathbf{K}^{D_i}(t_k)\left[\mathbf{z}^{D_i}(t_k) - \hat{\mathbf{z}}^{D_i}(t_k)\right], \tag{35}$$

$$\mathbf{P}^{D_i}(t_k|t_k) = \mathbf{P}^{D_i}(t_k|t_{k-1}) - \mathbf{K}^{D_i}(t_k)\mathbf{S}^{D_i}(t_k)\mathbf{K}^{D_i}(t_k)', \tag{36}$$

where

$$\hat{\mathbf{z}}^{D_i}(t_k) = \mathbf{h}\left[\hat{\mathbf{x}}^{D_i}(t_k|t_{k-1}), \hat{\mathbf{x}}^{A/B}(t_k)\right], \tag{37}$$

$$\mathbf{K}^{D_i}(t_k) = \mathbf{P}^{D_i}(t_k|t_{k-1})\mathbf{H}^{D_i}(t_k)'\mathbf{S}^{D_i}(t_k)^{-1}, \tag{38}$$

$$\mathbf{S}^{D_i}(t_k) = \mathbf{H}^{D_i}(t_k)\mathbf{P}^{D_i}(t_k|t_{k-1})\mathbf{H}^{D_i}(t_k)' + \mathbf{H}^{A/B}(t_k)\mathbf{P}^{A/B}(t_k)\mathbf{H}^{A/B}(t_k)' + \mathbf{R}^{D_i}, \tag{39}$$

and

$$\mathbf{H}^{D_i}(t_k) = \frac{\partial \mathbf{h}[\mathbf{x}^{D_i}(t_k), \mathbf{x}^{A/B}(t_k)]}{\partial \mathbf{x}(t_k)}, \tag{40}$$

$$\mathbf{H}^{A/B}(t_k) = \frac{\partial \mathbf{h}[\mathbf{x}^{D_i}(t_k), \mathbf{x}^{A/B}(t_k)]}{\partial \mathbf{x}^{A/B}(t_k)}, \tag{41}$$

where $\mathbf{x}^{A/B}$ and $\mathbf{P}^{A/B}$ are the state and error covariance of transmitter A or B based on who is emitting at time t_k.

It can be seen that the CI filter is the same as the Kalman filter except the innovation covariance computation in (39). The measurement error covariance $\mathbf{R}^{D_i}$ in the Kalman filter is replaced by the inflated measurement error covariance

$$\mathbf{H}^{A/B}(t_k)\mathbf{P}^{A/B}(t_k)\mathbf{H}^{A/B}(t_k)' + \mathbf{R}^{D_i} \tag{42}$$

in the CI filter. The transmitter error covariance $\mathbf{P}^{A/B}$ is added to the measurement noise. However, $\mathbf{P}^{A/B}$ is not "white," as the transmitter state estimate error is correlated to its history. This conflicts with the Kalman filter requirement on the measurement noise to be white. As indicated in [19], it is clearly not correct since, although the noise in (42) is zero mean, it is by no means white. It yields inconsistent estimates with covariances that are too small. However, the transmitter state estimation can still yield consistent results, as the augmented state $\mathbf{x}^{AB}$ is used.

Furthermore, the target and transmitter states are estimated separately in the CI algorithm, which ignores the contributions of the target measurements $\mathbf{z}^{D_i}$ to the transmitter state; namely, the cross-covariances [$\mathbf{P}^{AD_i}$ and $\mathbf{P}^{BD_i}$ in (27) in the optimal algorithm] are not handled correctly. Similarly to the simple approach, the individual target state estimation also causes the cross-covariances [$\mathbf{P}^{D_iD_j}$ with $i \neq j$ in (27) in the optimal algorithm] to be not treated correctly. Although the CI approach is better than the simple approach in term of information loss, it still has a gap to the optimal approach.

IV. THE HTF-D ALGORITHM

It is desirable to have an approach to track the targets individually with small state size, and obtain the estimates without losing any information. In this section, we will develop such an approach. The principle, shown in Fig. 2, consists of the following two steps:

(A) To track targets individually with minimal augmented states [defined in (43)]. An augmented state consists of the two transmitter states and one particular target state. It is updated by the corresponding transmitter and target measurements defined in (44). Since the transmitter measurements, $\mathbf{z}^A$ and $\mathbf{z}^B$, are used to update all the individual tracks, the individual track state estimation errors are highly correlated, since they include common information.

(B) To fuse the individual track estimates through a special information fusion algorithm. The common information among the individual tracks is computed first, so that the common information can be decorrelated, and the individual tracks can then be fused without double counting the common information.

We designate this approach as the HTF-D. The details of the two steps are given next.

A. Individual Target Tracking

Assume there are n targets in the system. The ith individual target state and measurement vectors are defined as

$$\mathbf{x}^i(t_k) = [\mathbf{x}^A(t_k)' \ \mathbf{x}^B(t_k)' \ \mathbf{x}^{D_i}(t_k)']', \tag{43}$$

$$\mathbf{z}^i(t_k) = [\mathbf{z}^A(t_k)' \ \mathbf{z}^B(t_k)' \ \mathbf{z}^{D_i}(t_k)']', \tag{44}$$

where $i = 1, \ldots, n$, and n is the number of the targets. The NCV state transition model and measurement model are

$$\mathbf{x}^i(t_k) = \mathbf{F}^i(T_{k,k-1})\mathbf{x}^i(t_{k-1}) + \mathbf{v}^i(T_{k,k-1}), \tag{45}$$

$$\mathbf{z}^i(t_k) = \mathbf{h}^i[\mathbf{x}^i(t_k)] + \mathbf{w}^i(t_k), \tag{46}$$

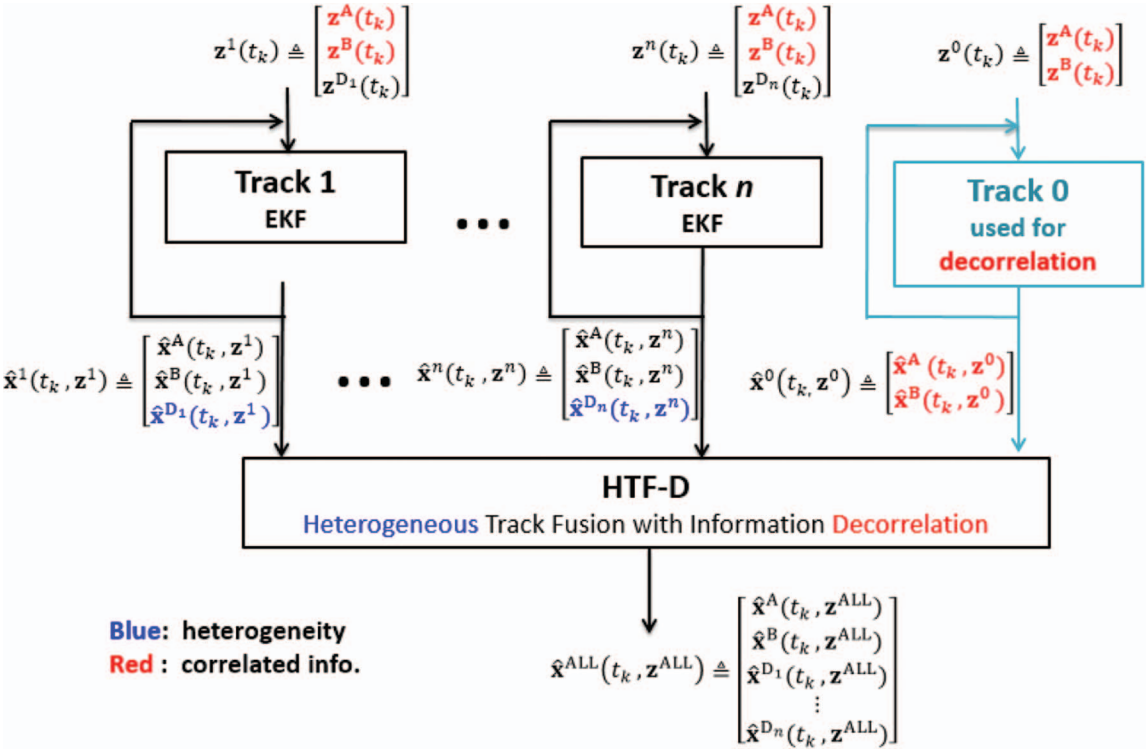

Fig. 2. The flowchart of the HTF-D algorithm. It consists of two steps: 1) Individual target tracking to generate tracks 0–n. These tracks are heterogeneous and correlated. 2) Fusion of these heterogeneous and correlated tracks.

where

$$\mathbf{F}^i(T_{k,k-1}) = \begin{bmatrix} \mathbf{F}(T_{k,k-1}) & \mathbf{0}_{4\times4} & \mathbf{0}_{4\times4} \\ \mathbf{0}_{4\times4} & \mathbf{F}(T_{k,k-1}) & \mathbf{0}_{4\times4} \\ \mathbf{0}_{4\times4} & \mathbf{0}_{4\times4} & \mathbf{F}(T_{k,k-1}) \end{bmatrix} \tag{47}$$

is the state transition matrix, $\mathbf{h}^i[\cdot]$ is the measurement function that follows from (8)–(21), and $\mathbf{v}^i(\cdot)$ and $\mathbf{w}^i(\cdot)$ are the process and measurement noises, respectively. The extended Kalman filter is used for the dynamic estimation. The ith individual track state estimate is

$$\hat{\mathbf{x}}^i(t_k, \mathbf{z}^i) = [\hat{\mathbf{x}}^{\mathrm{A}}(t_k, \mathbf{z}^i)' \ \hat{\mathbf{x}}^{\mathrm{B}}(t_k, \mathbf{z}^i)' \ \hat{\mathbf{x}}^{\mathrm{D}_i}(t_k, \mathbf{z}^i)']', \tag{48}$$

where $\mathbf{z}^i$ indicates that the state is contributed by the measurement $\mathbf{z}^i$ consisting of $\mathbf{z}^{\mathrm{A}}, \mathbf{z}^{\mathrm{B}}$, and $\mathbf{z}^{\mathrm{D}_i}$.

It can be seen that the individual state estimates $\hat{\mathbf{x}}^1(t_k), \ldots, \hat{\mathbf{x}}^n(t_k)$ are heterogeneous and their errors will be highly correlated. The heterogeneity of the individual track states shown in blue in Fig. 2 is due to different $\mathbf{x}^{\mathrm{D}_1}, \ldots, \mathbf{x}^{\mathrm{D}_n}$ in the states. The correlation shown in red in Fig. 2 is caused by the common information from the transmitter measurements $\mathbf{z}^{\mathrm{A}}$ and $\mathbf{z}^{\mathrm{B}}$.

To quantify the correlation among the individual state estimates, we add an additional reference track 0. This reference track is contributed by the transmitter measurements (the common measurements) $\mathbf{z}^{\mathrm{A}}$ and $\mathbf{z}^{\mathrm{B}}$ only. The state and measurement vectors of the reference track are defined as

$$\mathbf{x}^0(t_k) = [\mathbf{x}^{\mathrm{A}}(t_k)' \ \mathbf{x}^{\mathrm{B}}(t_k)']', \tag{49}$$

$$\mathbf{z}^0(t_k) = [\mathbf{z}^{\mathrm{A}}(t_k)' \ \mathbf{z}^{\mathrm{B}}(t_k)']'. \tag{50}$$

The NCV model is applied for the reference track estimation. The estimate state is

$$\hat{\mathbf{x}}^0(t_k, \mathbf{z}^0) = [\hat{\mathbf{x}}^{\mathrm{A}}(t_k, \mathbf{z}^0)' \ \hat{\mathbf{x}}^{\mathrm{B}}(t_k, \mathbf{z}^0)']', \tag{51}$$

where $\mathbf{z}^0$ indicates that the state is contributed by the measurement $\mathbf{z}^0$, namely, $\mathbf{z}^{\mathrm{A}}$ and $\mathbf{z}^{\mathrm{B}}$ only.

The individual track initiation uses the iterated least squares algorithm [18], [19] on a batch of measurements.

B. Individual Heterogeneous Track Fusion

This section shows how to combine the individual track states into one fused state estimate

$$\hat{\mathbf{x}}^{\mathrm{ALL}}(t_k, \mathbf{z}^{\mathrm{ALL}}) = [\hat{\mathbf{x}}^{\mathrm{A}}(t_k, \mathbf{z}^{\mathrm{ALL}})' \ \hat{\mathbf{x}}^{\mathrm{B}}(t_k, \mathbf{z}^{\mathrm{ALL}})' \ \hat{\mathbf{x}}^{\mathrm{D}_1}(t_k, \mathbf{z}^{\mathrm{ALL}})' \ \cdots \ \hat{\mathbf{x}}^{\mathrm{D}_n}(t_k, \mathbf{z}^{\mathrm{ALL}})']', \tag{52}$$

where $\mathbf{z}^{\mathrm{ALL}}$ indicates that the state is contributed by all transmitter and target measurements.

To fuse these highly correlated and heterogeneous tracks, we need to examine the information content of each individual track estimate. Fig. 3 shows the details for a scenario with three targets. For a particular individual track i, its information vector and information matrix are

$$\hat{\mathbf{y}}^i(t_k, \mathbf{z}^i) = \mathbf{P}^i(t_k, \mathbf{z}^i)^{-1}\hat{\mathbf{x}}^i(t_k, \mathbf{z}^i), \tag{53}$$

$$\mathbf{Y}^i(t_k, \mathbf{z}^i) = \mathbf{P}^i(t_k, \mathbf{z}^i)^{-1}. \tag{54}$$

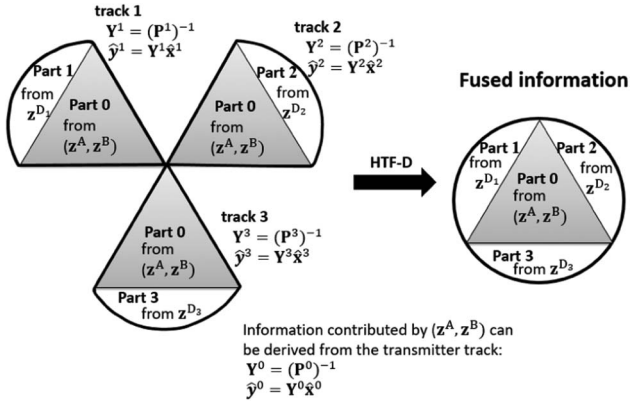

Fig. 3. Information contribution distribution in a three-target scenario.

The above can be separated into two parts. Part **0** (the gray area in Fig. 3) is contributed by the transmitter measurements

$$\mathbf{z}^{\mathrm{A}}(t_1), \ldots, \mathbf{z}^{\mathrm{A}}(t_k), \tag{55}$$

$$\mathbf{z}^{\mathrm{B}}(t_1), \ldots, \mathbf{z}^{\mathrm{B}}(t_k), \tag{56}$$

and Part **i** (the white area in Fig. 3, $\mathbf{i} \in \{\mathbf{1}, \mathbf{2}, \mathbf{3}\}$) is contributed by the particular target measurement

$$\mathbf{z}^{\mathrm{D}_i}(t_1), \ldots, \mathbf{z}^{\mathrm{D}_i}(t_k). \tag{57}$$

The fused information (the full circle shown in Fig. 3) is the sum of Parts **0**, **1**, **2**, and **3**. Part **0** cannot be used more than once as it is from the same source.

The key challenge is to separate the information corresponding to the individual track i into Part **0** and Part **i**. This can be done through the information provided by the reference track 0, which is contributed by the transmitter measurements only, and it is actually equivalent to Part **0** embedded in each individual track. The information vector and matrix of Part **i** in track i can then be decorrelated by

$$\hat{\mathbf{y}}^{\mathbf{i},i}(t_k, \mathbf{z}^{\mathrm{D}_i}) = \hat{\mathbf{y}}^i(t_k, \mathbf{z}^i) \ominus \hat{\mathbf{y}}^0(t_k, \mathbf{z}^0), \tag{58}$$

$$\mathbf{Y}^{\mathbf{i},i}(t_k, \mathbf{z}^{\mathrm{D}_i}) = \mathbf{Y}^i(t_k, \mathbf{z}^i) \ominus \mathbf{Y}^0(t_k, \mathbf{z}^0), \tag{59}$$

where $\ominus$ is heterogeneous "information subtraction." Since $[\hat{\mathbf{y}}^i(\cdot), \hat{\mathbf{Y}}^i(\cdot)]$ with $i = 1, \ldots, n$ and $[\hat{\mathbf{y}}^0(\cdot), \hat{\mathbf{Y}}^0(\cdot)]$ are heterogeneous (of dimensions 12 and 8, respectively), the subtraction needs homogenization. The reference track has no information about $\mathbf{x}^{\mathrm{D}_i}$; 0s are used to pad the missing elements in $[\hat{\mathbf{y}}^0(\cdot), \hat{\mathbf{Y}}^0(\cdot)]$. The information subtraction in (58)–(59) is rewritten as

$$\begin{aligned}\hat{\mathbf{y}}^{\mathbf{i},i}(t_k, \mathbf{z}^{\mathrm{D}_i}) &= \hat{\mathbf{y}}^i(t_k, \mathbf{z}^i) \ominus \hat{\mathbf{y}}^0(t_k, \mathbf{z}^0) \\ &= \begin{bmatrix} \hat{\mathbf{y}}^{\mathrm{A}}(t_k, \mathbf{z}^i) \\ \hat{\mathbf{y}}^{\mathrm{B}}(t_k, \mathbf{z}^i) \\ \hat{\mathbf{y}}^{\mathrm{D}_i}(t_k, \mathbf{z}^i) \end{bmatrix} \ominus \begin{bmatrix} \hat{\mathbf{y}}^{\mathrm{A}}(t_k, \mathbf{z}^0) \\ \hat{\mathbf{y}}^{\mathrm{B}}(t_k, \mathbf{z}^0) \end{bmatrix} \\ &= \begin{bmatrix} \hat{\mathbf{y}}^{\mathrm{A}}(t_k, \mathbf{z}^i) \\ \hat{\mathbf{y}}^{\mathrm{B}}(t_k, \mathbf{z}^i) \\ \hat{\mathbf{y}}^{\mathrm{D}_i}(t_k, \mathbf{z}^i) \end{bmatrix} - \begin{bmatrix} \hat{\mathbf{y}}^{\mathrm{A}}(t_k, \mathbf{z}^0) \\ \hat{\mathbf{y}}^{\mathrm{B}}(t_k, \mathbf{z}^0) \\ \mathbf{0}_{4\times 1} \end{bmatrix}, \end{aligned} \tag{60}$$

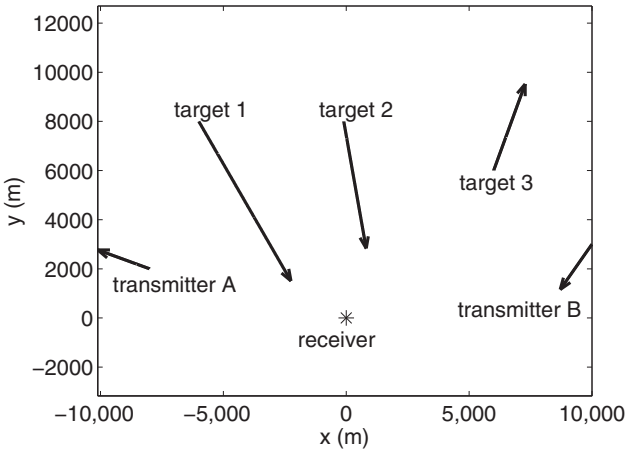

Fig. 4. Scenario 1.

$$\begin{aligned}\mathbf{Y}^{\mathbf{i},i}(t_k, \mathbf{z}^{\mathrm{D}_i}) &= \mathbf{Y}^i(t_k, \mathbf{z}^i) \ominus \mathbf{Y}^0(t_k, \mathbf{z}^0) \\ &= \hat{\mathbf{Y}}^i(t_k, \mathbf{z}^i) - \begin{bmatrix} \hat{\mathbf{Y}}^0(t_k, \mathbf{z}^0) & \mathbf{0}_{8\times 4} \\ \mathbf{0}_{4\times 8} & \mathbf{0}_{4\times 4} \end{bmatrix}. \end{aligned} \tag{61}$$

After all individual track information is decorrelated into two parts, the fused the information vector and matrix are computed as

$$\begin{aligned}\hat{\mathbf{y}}^{\mathrm{ALL}}(t_k, \mathbf{z}^{\mathrm{ALL}}) &= \hat{\mathbf{y}}^0(t_k, \mathbf{z}^0) \oplus \hat{\mathbf{y}}^{\mathbf{1},1}(t_k, \mathbf{z}^{\mathrm{D}_1}) \oplus \cdots \\ &\quad \oplus \hat{\mathbf{y}}^{\mathbf{n},n}(t_k, \mathbf{z}^{\mathrm{D}_n}), \end{aligned} \tag{62}$$

$$\begin{aligned}\hat{\mathbf{Y}}^{\mathrm{ALL}}(t_k, \mathbf{z}^{\mathrm{ALL}}) &= \hat{\mathbf{Y}}^0(t_k, \mathbf{z}^0) \oplus \hat{\mathbf{Y}}^{\mathbf{1},1}(t_k, \mathbf{z}^{\mathrm{D}_1}) \oplus \cdots \\ &\quad \oplus \hat{\mathbf{Y}}^{\mathbf{n},n}(t_k, \mathbf{z}^{\mathrm{D}_n}), \end{aligned} \tag{63}$$

where $\oplus$ is heterogeneous "information addition," which is implemented through information homogenization as done in the heterogeneous "information subtraction" before, namely, to pad with 0s the missing elements.

The fused vector and matrix are then computed as

$$\hat{\mathbf{x}}(t_k, \mathbf{z}^{\mathrm{ALL}}) = \mathbf{Y}(t_k, \mathbf{z}^{\mathrm{ALL}})^{-1}\hat{\mathbf{y}}(t_k, \mathbf{z}^{\mathrm{ALL}}), \tag{64}$$

$$\mathbf{P}(t_k, \mathbf{z}^{\mathrm{ALL}}) = \mathbf{Y}(t_k, \mathbf{z}^{\mathrm{ALL}})^{-1}. \tag{65}$$

The individual transmitter and target state estimates and error covariances can then be obtained as the appropriate parts/blocks from $\hat{\mathbf{x}}(t_k, \mathbf{z}^{\mathrm{ALL}})$ and $\mathbf{P}(t_k, \mathbf{z}^{\mathrm{ALL}})$.

Since the HTF-D does not discard or ignore any information, it provides the optimal estimates. Furthermore, it does not have the drawbacks of the "optimal" algorithm,[3] as it does not estimate a big augmented state.

[3] In the sequel, we shall use the designation "optimal" for the optimal algorithm when it suffers from numerical problem, due its large augmented state.

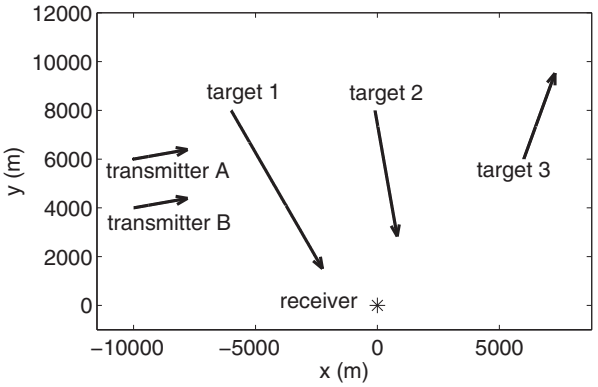

Fig. 5. Scenario 2.

V. SIMULATION RESULTS

A. Test Scenarios

We consider two test scenarios. Scenario 1 is shown in Fig. 4. The receiver stays at position $(0, 0)$. The two transmitters A and B move with speed of 3 m/s. Their headings are -70° and -145° (clockwise from True North), respectively. There are three targets. Targets 1 and 2 are approaching to the receiver with speeds of 10 and 7 m/s, respectively. Target 3 is moving away from the receiver with speed of 5 m/s. The headings of the three targets are 150°, 170°, and 20°, respectively. This scenario has good observability. The two transmitters are well separated,[4] and targets are not located in the blind zone of the multistatic system.[5]

Scenario 2 is shown in Fig. 5. The receiver location and three target trajectories are the same as in Scenario 1, but the transmitters move close to each other. The observability of this scenario is not as good as that of Scenario 1, as the two transmitters are not well separated.

In both scenarios, transmitters A and B emit asynchronous pulse signals with starting times 1 and 10 s, respectively. They have the same pulse interval 30 s. The total simulation duration is 750 s. The receiver receives the pulse signals with the appropriate signal propagation delays (the delays are not shown for simplicity) at

$$\{1\,\text{s}\ \ 31\,\text{s}\ \ \ldots\ \ 721\,\text{s}\}\ \text{from transmitter A}, \tag{66}$$

$$\{10\,\text{s}\ \ 40\,\text{s}\ \ \ldots\ \ 730\,\text{s}\}\ \text{from transmitter B}. \tag{67}$$

The bistatic range measurement error standard deviation is $\sigma_r = 20$ m in both scenarios. The bearing measurement error standard deviation is $\sigma_b = 1^{\circ}$ in Scenario 1, and it increases to $\sigma_b = 1.5^{\circ}$ in Scenario 2 to make estimation more difficult. The probability of detection in both scenarios is 0.8. Two random false alarms were added at each time. A preprocessing with a Kalman filter with 2-D assignment data association is used to track the bistatic range and bearing to filter out the false alarms. The measurements that passed the preprocessing will be used for trajectory estimation. The preprocessing details can be found in [22], where the approach is used to filter false alarms in the measurements of bearing and Doppler frequency.

B. Algorithms Used in the Simulation Test

Four algorithms are tested for comparison. They are as follows:

- *Simple*: This approach estimates the transmitter and target trajectories individually. It estimates the transmitter states first, and then the target states. The drawback is that the transmitter state estimation errors are not taken into consideration in the target estimation. The algorithm details can be found in Section 3.2.
- *CI*: This approach estimates the two transmitters using an augmented state first, and then estimates target states individually. The transmitter estimation errors are taken into consideration in the target state estimation (but only as extra white measurement noise), and some cross-correlations are ignored. The details can be found in Section 3.3.
- *Optimal*: This approach estimates a large augmented state consisting of all transmitter and target states. It can provide the optimal estimates. However, the large augmented state introduces some numerical issues and increases the system sensitivity to the noise. The algorithm details can be found in Section 3.1.
- *HTF-D*: This is the algorithm proposed in the paper. It tracks targets individually (without a large augmented state), and obtains the optimal estimation through heterogeneous track fusion. The details are given in Section 4.

In the simulation tests, the process noise PSD q in (7) is set to 10^{-6} m^2/s^3. The batch measurements for track initiation are taken from time 0 to 100 s.

C. Results of Scenario 1

We conducted 100 Monte Carlo runs on the first scenario using the four algorithms mentioned earlier. The position estimate root mean square errors (RMSEs) of the two transmitters and three targets versus time are presented in Figs. 6–10, respectively.

From the test results, we can observe that the simple algorithm (which ignores the transmitter estimation errors) provides the worst accuracy. The CI algorithm (which considers the transmitter estimation errors as white, and ignores some cross-correlations) is better than

[4]The problem is not observable when the two transmitters are located on the same line from the receiver (see Appendix A).

[5]When a target and a transmitter are located on similar bearings from the receiver, the weak indirect path signal is blocked by the strong direct path signal. This creates a detection blind zone for a target near the line between the transmitter and the receiver. The target observability in the blind zone is also very marginal due to the geometry.

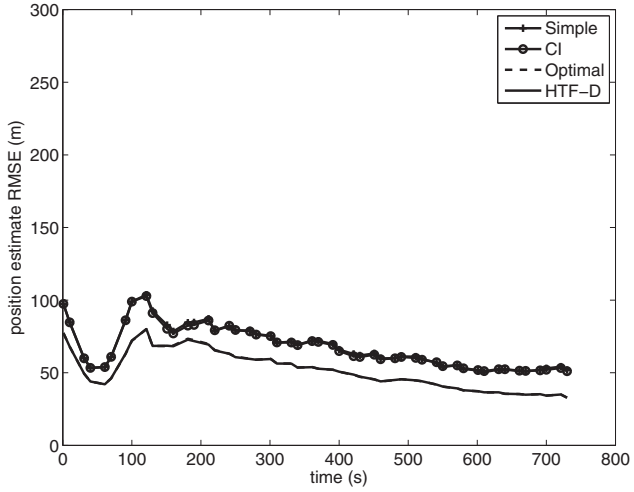

Fig. 6. Position estimate RMSE of transmitter A versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are overlapping each other as they provide the same result.

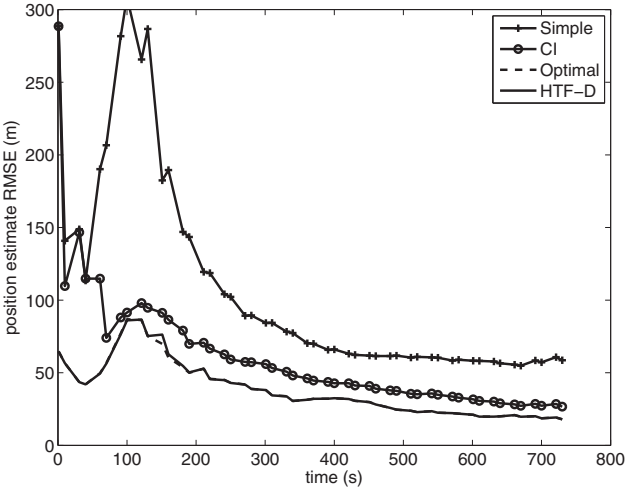

Fig. 8. Position estimate RMSE of target 1 versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are almost overlapping each other as they provide similar results.

the simple algorithm. The HTF-D and the optimal algorithms provide the best estimates and have practically the same results. Their accuracy (solid and dash) lines overlap in Figs. 6–10, except a slight difference in Fig. 8 (around time 150 s), which is due to numerical issues of Matlab computation. So, we can say that the HTF-D achieves the optimal performance in this scenario.

To evaluate the consistency of the algorithms, the average normalized estimation error squared (NEES) of position is evaluated. The average position NEES [19] at time t_k for $N = 100$ Monte Carlo runs is

$$\bar{\epsilon}(t_k) = \frac{1}{Nn} \sum_{i=1}^{N} \tilde{\mathbf{x}}^i_{1:2}(t_k)' \mathbf{P}^i_{1:2,1:2}(t_k)^{-1} \tilde{\mathbf{x}}^i_{1:2}(t_k), \qquad (68)$$

where $n = 2$ is the dimension of the position state, $\mathbf{P}_{1:2,1:2}(t_k)$ is the position estimation error covariance at time t_k, i is the run index,

$$\tilde{\mathbf{x}}_{1:2}(t_k) = \mathbf{x}_{1:2}(t_k) - \hat{\mathbf{x}}_{1:2}(t_k) \qquad (69)$$

is the position estimation error at time t_k, and $\tilde{\mathbf{x}}_{1:2}(t_k)$ and $\mathbf{x}_{1:2}(t_k)$ are the position estimate and ground truth at time t_k, respectively. The two-sided 95% probability region for an $Nn = 200$ degrees of freedom ($N = 100, n = 2$) chi-square random variable is [163, 241]. Dividing by 200, the average NEES should be in the interval [0.815, 1.205]. Figs. 11–15 show the average position NEES versus time of the CI, and optimal and HTF-D algorithms for the two transmitters and three targets, respectively. The NEES of the simple algorithm (around 50–150) is too far away from the boundary, so it is not displayed in the figures. It can be seen that the NEESs of the HTF-D and optimal algorithms are all within the 95% probabil-

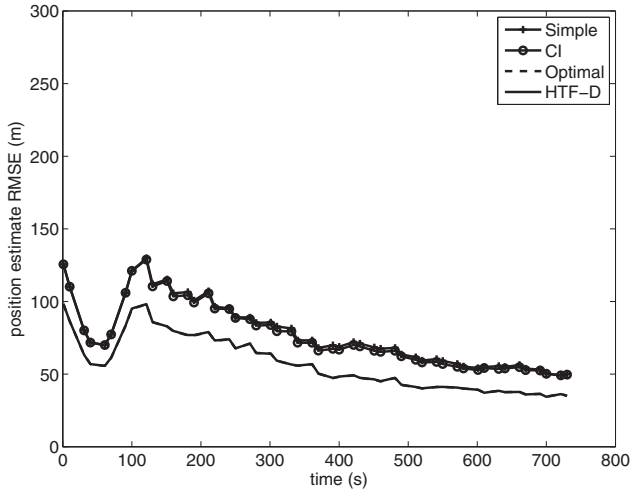

Fig. 7. Position estimate RMSE of transmitter B versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are overlapping each other as they provide the same result.

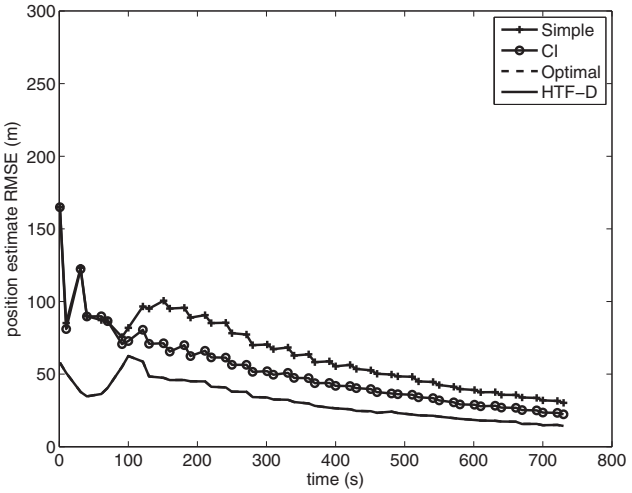

Fig. 9. Position estimate RMSE of target 2 versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are overlapping each other as they provide the same result.

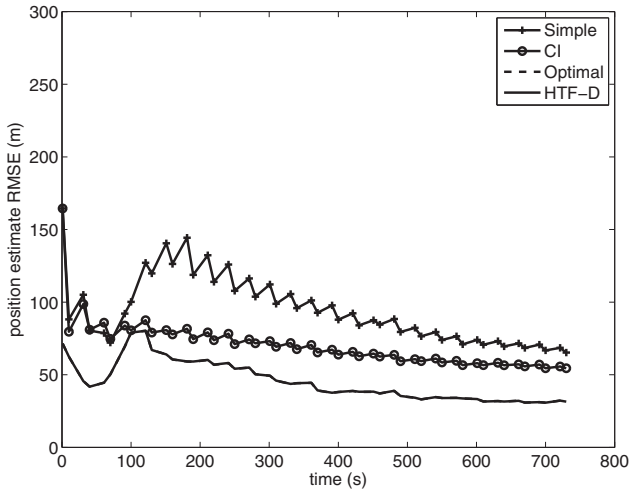

Fig. 10. Position estimate RMSE of target 3 versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are overlapping each other as they provide the same result.

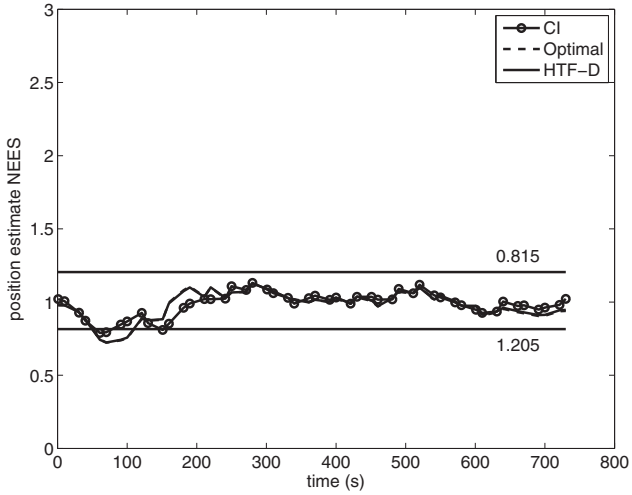

Fig. 11. Position NEES of transmitter A versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are overlapping each other as they provide the same result.

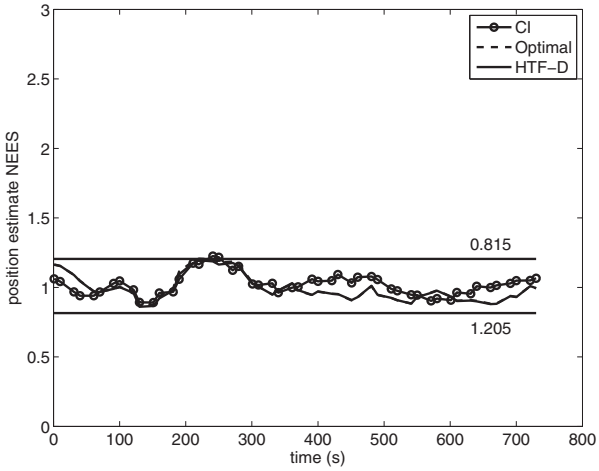

Fig. 12. Position NEES of transmitter B versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are overlapping each other as they provide the same result.

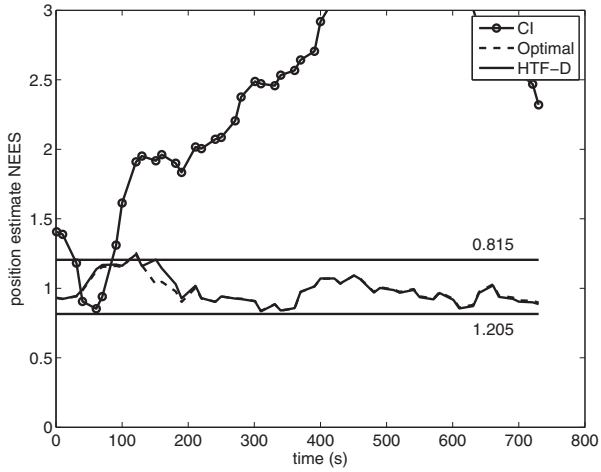

Fig. 13. Position NEES of target 1 versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are almost overlapping each other as they provide similar results.

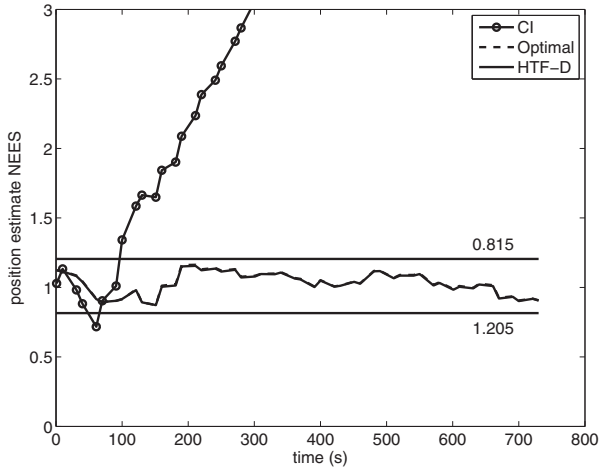

Fig. 14. Position NEES of target 2 versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are overlapping each other as they provide the same result.

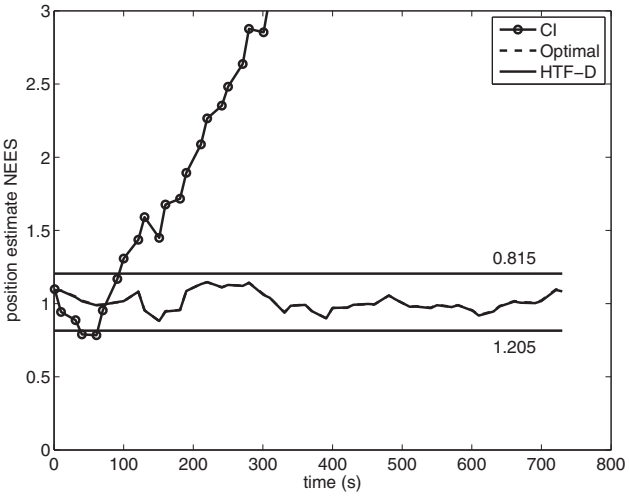

Fig. 15. Position NEES of target 3 versus time from 100 Monte Carlo runs (Scenario 1). The results of the HTF-D (solid line) and optimal (dashed line) algorithms are overlapping each other as they provide the same result.

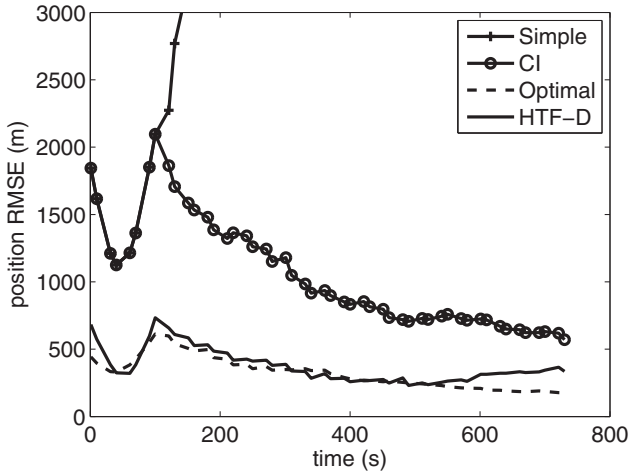

Fig. 16. Position estimate RMSE of transmitter A versus time from 100 Monte Carlo runs (Scenario 2).

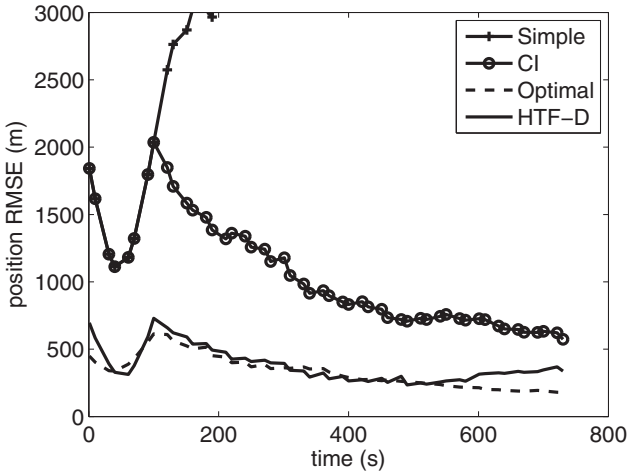

Fig. 17. Position estimate RMSE of transmitter B versus time from 100 Monte Carlo runs (Scenario 2).

ity region [0.815, 1.205]. This shows that both the HTF-D and optimal algorithms yield consistent estimation results.

For the CI algorithm, the transmitter position NEESs are within the 95% probability region (see Figs. 11 and 12). However, the target position NEESs are much larger than the upper bound after track initiation (see Figs. 13–15, $t > 100$ s). This is because the transmitters are estimated using the augmented state given in (28). The estimation error cross-covariance between the transmitters is taken into consideration. The transmitter estimates are therefore yield consistent estimates. However, the target state estimation treats the transmitter position error as white noise, but this is actually not white. The cross-correlations with the historical transmitter estimation errors are ignored. This leads to the target state estimation error covariances smaller than their actual values. The NEESs are therefore above the upper bound.

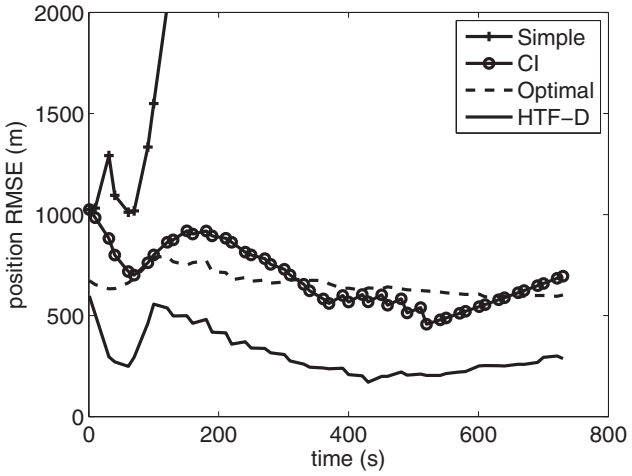

Fig. 18. Position estimate RMSE of target 1 versus time from 100 Monte Carlo runs (Scenario 2).

D. Results of Scenario 2

We also conducted 100 Monte Carlo runs for Scenario 2, which has marginal observability due to its geometry and larger measurement error ($\sigma_b = 1.5^{\circ}$) than Scenario 1. The position estimation RMSEs of the two transmitters and three targets versus time are presented in Figs. 16–20, respectively.

It can be seen that the simple algorithm diverges for all the transmitter and target estimates. The CI approach still shows noticeable gaps compared to the HTF-D.

An interesting observation is that the "optimal" algorithm performs worse than the HTF-D for the target 1 estimation shown in Fig. 18. This is because the "optimal" algorithm has a state vector with large size ($5 \times 4 = 20$, for the 2 transmitters plus 3 targets, each of them has 4 state elements). This creates more numerical problems than a system with a smaller state, and causes the "optimal" algorithm to yield larger errors than the HTF-D algorithm, especially when the measurement noise is large.

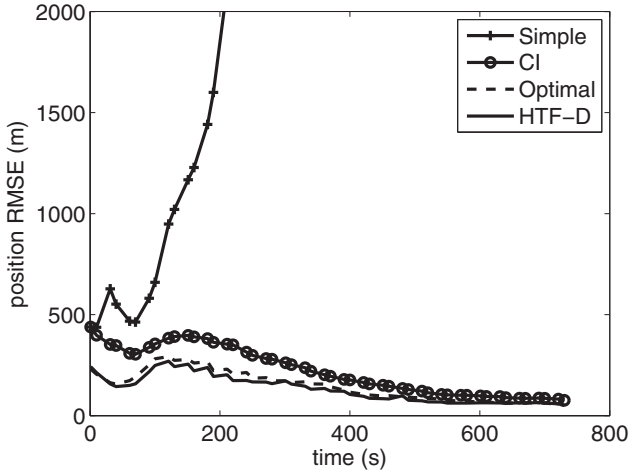

Fig. 19. Position estimate RMSE of target 2 versus time from 100 Monte Carlo runs (Scenario 2).

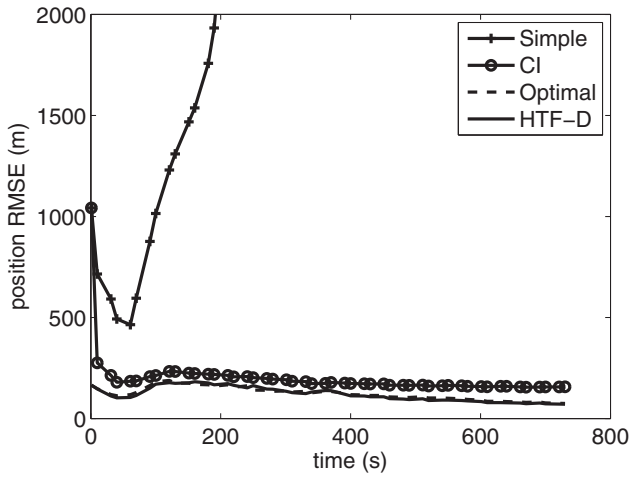

Fig. 20. Position estimate RMSE of target 3 versus time from 100 Monte Carlo runs (Scenario 2).

These results show that the "optimal" algorithm is not robust when the measurement error is large and observability is not good. The number of targets in the system also affects its estimation quality.

VI. CONCLUSIONS

This paper deals with target tracking using a new configuration of a non-cooperative multistatic system where the positions of moving transmitters are unknown. The new HTF-D algorithm was developed to track each target individually, and fuse these correlated and heterogeneous tracks using a novel decorrelation approach. Since it does not lose any information, it can achieve the accuracy of the optimal algorithm. As the HTF-D tracks each target individually, its state size is fixed and small. The drawbacks of the optimal algorithm (such as sensitivity to higher noise and difficulty to implement in real systems) are avoided.

Simulation results showed that the HTF-D indeed obtained the accuracy of the optimal algorithm. The NEES results showed that the HTF-D is consistent statistically as well. We also observed that the "optimal" algorithm yielded larger estimation error than the HTF-D in Scenario 2, where the measured bearing error is larger and observability is not good. This showed that the HTF-D is more robust than the "optimal" algorithm.

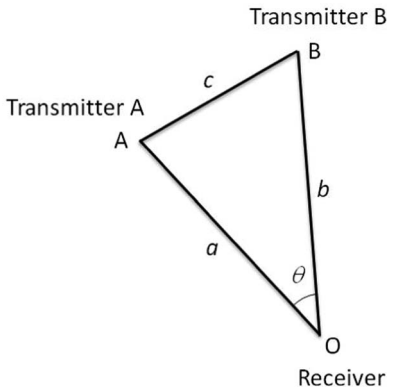

Fig. A1. The triangle formed by the receiver and two transmitters.

APPENDIX A OBSERVABILITY

It is known that a multistatic tracking system is observable when the positions of transmitters and receiver are known. If both transmitters' positions can be solved from the measurements r^{AB}, r^{BA}, b^{A}, and b^{B} [defined in (11), (12), and (21), respectively] uniquely, the problem defined in Section 2 is observable.

Without loss of generality, the problem can be simplified as to determine the triangle $\triangle$OAB shown in Fig. A1 from r^{AB}, r^{BA}, b^{A}, and b^{B}. This can be done in one time cycle. The orientations of $\overline{OA}$ and $\overline{OB}$ are fixed by b^{A} and b^{B}, and

$$\theta = b^{A} - b^{B} \tag{A.1}$$

is known. The problem is then of solving for a and b from r^{AB} and r^{BA} using the nonlinear model

$$\begin{bmatrix} r^{AB} \\ r^{BA} \end{bmatrix} = \mathbf{h}_0 \left(\begin{bmatrix} a \\ b \end{bmatrix} \right) + \mathbf{w}_0, \tag{A.2}$$

where $\mathbf{w}_0$ is the error of $[r^{AB}\ r^{BA}]'$,

$$h_{0,1}(\cdot) = c + a - b, \tag{A.3}$$

$$h_{0,2}(\cdot) = c + b - a, \tag{A.4}$$

and

$$c = \sqrt{a^2 + b^2 - 2ab\cos\theta}. \tag{A.5}$$

This problem is observable when

$$\det(\mathbf{H}_0) \neq 0, \tag{A.6}$$

where

$$\mathbf{H}_0 = [\nabla_{\mathbf{y}} \mathbf{h}_0(\mathbf{y})']' \tag{A.7}$$

with $\mathbf{y} = [a\ b]'$. $\mathbf{H}_0$ is derived as

$$\mathbf{H}_0 = \begin{bmatrix} \dfrac{a - b\cos\theta + c}{c} & \dfrac{b - a\cos\theta - c}{c} \\ \dfrac{a - b\cos\theta - c}{c} & \dfrac{b - a\cos\theta + c}{c} \end{bmatrix} \tag{A.8}$$

and its determinant is

$$\det(\mathbf{H}_0) = \frac{2(a+b)(1-\cos\theta)}{c}. \tag{A.9}$$

Since $(a + b) > 0$, $\det(\mathbf{H}_0)$ is 0 when $(1 - \cos\theta)/c$ is 0. This latter condition occurs only when θ is 0^{o} (or 180^{o}). Thus, the problem is observable when the two transmitters are not located on the same line of sight from the receiver.

REFERENCES

[1] V. J. Aidala
"Kalman filter behavior in bearings-only tracking applications,"

IEEE Trans. Aerosp. Electron. Syst., vol. 1, no. 1, pp. 29–39, Jul. 1978.

[2] H. A. J. Huang, Y. Bar-Shalom, R. Yang, and G. W. Ng
"Tracking a maneuvering target using two heterogeneous passive sensors on a single stationary platform with IMM estimation,"
J. Adv. Inf. Fusion, vol. 12, no. 1, pp. 110–124, Jun. 2017.

[3] C. Jauffret and Y. Bar-Shalom
"Track formation with bearing and frequency measurements in clutter,"
IEEE Trans. Aerosp. Electron. Syst., vol. 26, no. 6, pp. 999–1009, Nov. 1990.

[4] C. Jauffret and A.-C. Pignol
"Target motion analysis by inverse triangulation,"
in *Proc. 18th Int. Conf. Inf. Fusion*, Washington, DC, USA, Jul. 2015.

[5] A. G. Lindgren and K. F. Gong
"Position and velocity estimation via bearing observations,"
IEEE Trans. Aerosp. Electron. Syst., vol. 14, no. 4, pp. 564–577, Jul. 1978.

[6] J. M. Passerieux, D. Pillon, P. Blanc-Benon, and C. Jauffret
"Target motion analysis with bearing and frequencies measurements,"
in *Proc. 22nd Asilomar Conf.*, Pacific Grove, CA, USA, Nov. 1988.

[7] R. Yang, Y. Bar-Shalom, and G. W. Ng
Bearings-only Tracking with Fusion from Heterogenous Passive Sensors: ESM/EO and Acoustic.
In *Proceedings of the 18th International Conference on Information Fusion*, Washington, DC, July 2015.

[8] R. Yang, Y. Bar-Shalom, and G. W. Ng
"Bearings-only tracking with tusion from heterogenous passive sensors: ESM/EO and acoustic,"
J. Adv. Inf. Fusion, vol. 12, no. 3, pp. 3–17, Jul. 2017.

[9] D. Grimmett, S. Coraluppi, R. B. La Cour, C. G. Hempel, T. Lang, P. A. M. de Theije, and P. Willett
"MSTWG multistatic tracker evaluation using simulated scenario data sets,"
in *Proc. 11th Int. Conf. Inf. Fusion*, Cologne, Germany, Jul. 2008.

[10] S. Benen, D. Maiwald, and H. Schmidt-Schierhorn
"Low frequency towed active sonar (LFTAS) in multistatic applications,"
GI Jahrestagung, LNI, vol. 154, pp. 2413–2421, 2009.

[11] M. Broetje and K. Pikora
"Parameter estimation for multistatic active sonar using extended fixed points,"
in *Proc. 18th Int. Conf. Inf. Fusion*, Washington, DC, USA, Jul. 2015.

[12] M. Michaelis, M. Broetje, and F. Ehlers
"Case study: limitations of parameter estimation for non-cooperative multistatic active sonar by association ambiguities,"
in *Proc. Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, Bonn, Germany, Oct. 2014, pp. 1–6.

[13] V. Algeier, B. Demissie, W. Koch, and R. Thoma
"State space initiation for blind mobile terminal position tracking,"
EURASIP J. Adv. Signal Process., vol. 1, pp. 1–14, 2008.

[14] R. Mendrzik, F. Meyer, G. Bauch, and M. Win
"Enabling situational awareness in millimeter wave massive MIMO systems,"
IEEE J. Sel. Topics Signal Process., vol. 13, no. 5, pp. 1196–1211, Sep. 2019.

[15] W. Xu, F. Quitin, M. Leng, W. P. Tay, and S. G. Razul
"Distributed localization of a RF target in NLOS environments,"
IEEE J. Sel. Areas Commun., vol. 33, no. 7, pp. 1317–1330, Jul. 2015.

[16] M. Michaelis
"Bistatic simultaneous transmitter localization and mapping,"
in *Proc. Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, Bonn, Germany, Oct. 2015, pp. 1–6.

[17] R. Yang and Y. Bar-Shalom
"Adaptive target tracking using multistatic sensor with unknown moving transmitter positions,"
in *Proc. 7th IEEE Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Curacao, Dec. 2017.

[18] R. Yang, Y. Bar-Shalom, and G. W. Ng
"Target tracking using an asynchronous multistatic sensor system with unknown transmitter positions,"
in *Proc. 21st Int. Conf. Inf. Fusion*, Cambridge, U.K., Jul. 2018.

[19] Y. Bar-Shalom, P. Willett, and X. Tian
Tracking and Data Fusion: A Handbook of Algorithms. Storrs, CT, USA: YBS Publishing, 2011.

[20] P. Willett and S. Coraluppi
"Application of the MLPDA to bistatic sonar,"
in *Proc. 2005 IEEE Aerosp. Conf.*, Big Sky, MT, USA, 2005, pp. 2063–2073.

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

[22] R. Yang, Y. Bar-Shalom, and G. W. Ng
"Tracking/fusion and deghosting with Doppler frequency from two passive acoustic sensors,"
in *Proc. 16th Int. Conf. Inf. Fusion*, Istanbul, Turkey, Jul. 2013.

Rong Yang received the B.E. degree in information and control from Xi'an Jiao Tong University, China, in 1986, the M.Sc. degree in electrical engineering from the National University of Singapore in 2000, and the Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore, in 2012. She is currently a Principal Member of Technical Staff at the DSO National Laboratories, Singapore. Her research interests include passive tracking, low observable target tracking, GMTI tracking, hybrid dynamic estimation, and data fusion. She was Publicity and Publication Chair of FUSION 2012 and the recipient of the FUSION 2014 Best Paper Award (first runner-up).

Yaakov Bar-Shalom was born on May 11, 1941. He received the B.S. and M.S. degrees in electrical engineering from the Technion, Israel Institute of Technology, Haifa, Israel, in 1963 and 1967, respectively, and the Ph.D. degree in electrical engineering from Princeton University, Princeton, NJ, USA, in 1970. From 1970 to 1976, he was with the Systems Control, Inc., Palo Alto, CA, USA. He is currently Board of Trustees Distinguished Professor in the Department of Electrical and Computer Engineering and Marianne E. Klewin Professor in Engineering with the University of Connecticut, Storrs, CT, USA. His current research interests include estimation theory, target tracking, and data fusion. He has authored or coauthored more than 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; translated into Russian, MGTU Bauman, Moscow, Russia, 2011), *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 multitarget 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 was an Associate Editor for the IEEE Transactions on Automatic Control and from 1978 to 1981 as an Associate Editor for *Automatica*. He was General Chairman of the 1985 ACC and Co-Chairman of the 1989 IEEE International Conference on Control and Applications. During 1987–1989, he was a member of the Board of Governors of the IEEE CSS. He was General Chairman of FUSION 2000, President of ISIF in 2000 and 2002, and Vice President for Publications in 2004–2013. In 1987, he was the recipient of the IEEE CSS Distinguished Member Award. Since 1995, he has been a Distinguished Lecturer of the IEEE AESS and has given numerous keynote addresses at major national and international conferences. He was the corecipient 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 was the recipient of 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 the recipient of the IEEE Dennis J. Picard Medal for Radar Technologies and Applications, and in 2012, the Connecticut Medal of Technology. In 2015, he was the recipient of the Lifetime of Excellence in Information Fusion award from the International Society of Information Fusion, renamed in 2016 as "ISIF YaakovBar-Shalom Award for Lifetime of Excellence in Information Fusion". 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.