Communications in Information and Systems
Vol. 2, No. 4, pp. 325-348, December 2002

A FIXED-LAG SMOOTHING SOLUTION TO OUT-OF-SEQUENCE INFORMATION FUSION PROBLEMS[*]

SUBHASH CHALLA[†], ROBIN J. EVANS[†], XUEZHI WANG[†], AND JONATHAN LEGG[†]

Abstract. Multi-sensor tracking using delayed, out-of-sequence Information (OOSI) is a problem of growing importance due to an increased reliance on networked sensors interconnected via complex communication network architectures. In such systems, it is often the case that information (in the form of raw or processed measurements) is received out-of-time-order at the fusion center. Owing to compatibility with legacy sensors and limited communication bandwidth most practical fusion systems send track information rather than raw measurements to the fusion node. This paper presents a unified Bayesian approach to handling this out-of-sequence information problem and provides implementable sub-optimal algorithms for both cluttered and non-cluttered scenarios involving single and multiple time-delayed measurements/tracks. Such an approach leads to a solution involving the joint probability density of current and past target states. A fixed-lag smoothing framework, developed by John Moore and his students almost 30 years ago, forms the basis of our algorithm. Under linear Gaussian assumptions, the Bayesian solution reduces to an Augmented State Kalman Filter (AS-KF). Computationally efficient versions of the AS-KF are considered in this paper. Simulations are presented to evaluate the performance of these solutions.

Keywords. Target tracking, Networked sensors, Time delayed measurements, Out-of-Sequence Measurements (OOSM), Out-of-Sequence Tracks (OOST), Out-of-Sequence Information (OOSI), Fixed-Lag Smoothing, Smoothing.

1. Introduction.

In a multi-sensor centralized tracking system, sensors produce observations that are sent to a fusion center over communication networks which can introduce random delays. Thus there is no guarantee that data are received in the order they have originated. This problem has appeared in the literature under various names such as the Out Of Sequence Measurements (OOSM) problem [1, 2, 3, 4], the problem of tracking with random sampling and delays [5], [6], [7], and the problem of incorporating random time delayed measurements [8].

Most sensor networks communicate tracks (processed measurements) rather than raw measurements, owing to the prior existence of embedded trackers, and prohibitive communication bandwidths that favor summaries of measurements, rather than the measurements themselves. As in the case of a centralized tracking system, random delays are introduced resulting in reception of tracks out of sequence. This paper addresses this out-of-sequence tracking problem with equivalent measurements.

Equivalent measurements, their use and methods of extraction from track estimates are discussed extensively by Blackman and Popoli [4], Frenkel [9] and Drum-

[*] Received on March 28, 2002; accepted for publication on September 20, 2002. This work is financially supported by CSSIP and DSTO through TDFL.

[†] Center for Sensor Signal and Information Processing (CSSIP), Department of Electrical and Electronic Engineering, University of Melbourne, Victoria, 3010 Australia. E-mail: s.challa@ee.mu.oz.au, r.evans@ee.mu.oz.au, xu.wang@ee.mu.oz.au, Jonathan.Legg@dsto.defence.gov.au

mond [10]. Frenkel [9] demonstrates that the equivalent measurements remove all cross correlations between sensor level tracks and the central fused tracks, enabling them to be treated as normal sensor measurements with white measurement noise. Drummond [10] uses equivalent measurements to deal with the complex cross-correlations involved in track fusion when global tracks are fed back to sensor level tracks. In his paper, he presents two methods that are based on Frenkel's approach. However, Frenkel and Drummond did not extend the use of equivalent measurements beyond the track fusion problem, nor did they provide any Bayesian foundations for their use. In [11], Okello and Challa present the first use of equivalent measurements to the registration problem and exploit their Bayesian foundations in [12].

The basic problem can be formulated using a single sensor with measurements received in incorrect time order [1]. The problem can be stated as follows.

Denoting a "standard" measurement sequence $Y^k = \{y(t_1), y(t_2), \cdots, y(t_k)\}$, the target "standard" tracking problem reduces to the problem of computing the conditional mean estimate of the target state

$$\hat{x}(t_{k|k}) \triangleq E(x(t_k)|Y^k) \tag{1}$$

and its associated error covariance

$$P_{k|k} \triangleq E[(x(t_k) - \hat{x}(t_{k|k}))(x(t_k) - \hat{x}(t_{k|k}))'|Y^k]. \tag{2}$$

Under the assumption that the initial target state x_0 is Gaussian, the conditional mean estimate $\hat{x}(t_{k|k})$ of the target state, which is optimal in the minimum variance sense, can be computed recursively using the Kalman filter (KF). The basic KF algorithm can be easily extended to multi-sensor systems where the data is assumed to arrive at known times and in correct time sequence. A key problem arising when dealing with multiple interconnected sensors with communication links is time delay between the sensor and tracking computer. This problem can be defined as follows: When a measurement corresponding to time, τ, expressed as $y(\tau)$ arrives at time t_k after (1) and (2) have been computed, one faces the problem of updating the state estimate and its covariance with the delayed measurement, i.e., to compute

$$\hat{x}(t_{\tau|k}) \triangleq E(x(t_k)|Y^k, y(\tau)) \tag{3}$$

and

$$P_{\tau|k} \triangleq E[(x(t_k) - \hat{x}(t_{k|k}))(x(t_k) - \hat{x}(t_{k|k}))'|Y^k, y(\tau)]. \tag{4}$$

For the scenario where the delay corresponds to the time between the last two updates ($t_{k-1} \leq \tau < t_k$), an optimal solution has been proposed in the filtering framework by Bar-Shalom[1] where it is also stated that extension to longer time delays involves some kind of non-standard smoothing. Such an approach was later considered by Mallick et al. in [2]. Prior to this, the OOSM problem was solved, approximately, in [6]. The key difference between the algorithms in [1] and [6] is the effect of the process noise w on the filtering scheme. Although both consider the effect of process noise, the optimal solution incorporates the non-zero conditional mean of the process noise into the filter update equations while the approximate filter does not. A solution that completely ignores the process noise appeared in [13] and a numerical comparison between these various approaches was carried out on a simulated example in [1]. Based on these simulations and the complexity issues in tracker implementations, it was concluded in [1] that the sub-optimal solution that assumes the conditional mean of the process noise to be zero provides a good balance between accuracy and computational complexity especially when data association issues (e.g., tracking in clutter) are involved. An improved solution to the OOSI problem in clutter can be found in [14]. The following solutions and algorithms are proposed in this paper:

- A general Bayesian solution to the OOSM and OOST problems;
- An augmented state Kalman Filter (AS-KF) for tracking with OOSM and OOST when there is no clutter;
- Computationally efficient versions of the above algorithms like Iterative AS-KF, Variable dimension AS-KF.

The implementable algorithms like AS-KF and its variants are based on the fixed-lag smoothing framework developed by Moore and Tam almost 30 years ago [15, 16]. We use this framework with a simple modification to the measurement equation to cater for the delayed, out-of-sequence information.

The paper is organized as follows. A general Bayesian solution to the OOSM and OOST problem is considered in Section 2. It is then compared with the existing solutions in the literature in Section 3 . Filtering algorithms involving augmented states are necessarily computationally intensive. Hence, Section 4 is dedicated to understanding the computational issues and presenting alternate, computationally efficient solutions. Simulation results summarizing the best, worst and realistic performance of the OOSI fusion algorithms are presented in Section 5. Finally, conclusions are drawn in Section 6. Most of the detailed derivations are presented in the Appendix.

2. Bayesian Approach to OOSM and OOST Problems. Let $x(t_k)$ be the target state at time t_k, $Y(\tau)$ be the set of delayed sensor measurements, corresponding to time τ, and Y^k be the set of sensor measurement sequence received up to time t_k.

Having processed all the measurements Y^k, the complete information about the target state $x(t_k)$ is described by the probability density function $p(x(t_k)|Y^k)$. The OOSM problem arises as a consequence of receiving a measurement set $Y(\tau)$ at time t_k that corresponds to time $\tau < t_k$. The solution to this OOSM problem seeks to update $p(x(t_k)|Y^k)$ with $Y(\tau)$ to obtain $p(x(t_k)|Y^k, Y(\tau))$.

Invoking Bayes rule,

$$p(x(t_k)|Y^k, Y(\tau)) = \frac{p(Y(\tau)|x(t_k), Y^k)p(x(t_k)|Y^k)}{p(Y(\tau)|Y^k)}. \tag{5}$$

Consider the numerator of (5) and introducing the target state at time τ, $x(\tau)$,

$$\begin{aligned} p(Y(\tau)|x(t_k), Y^k)p(x(t_k)|Y^k) &= \int p(Y(\tau), x(\tau)|x(t_k), Y^k)p(x(t_k)|Y^k)dx(\tau) \\ &= \int p(Y(\tau)|x(\tau), x(t_k), Y^k)p(x(\tau)|x(t_k), Y^k)p(x(t_k)|Y^k)dx(\tau). \end{aligned}$$

Since $p(x(\tau), x(t_k)|Y^k) = p(x(\tau)|x(t_k), Y^k)p(x(t_k)|Y^k)$, we have

$$\begin{aligned} &p(Y(\tau)|x(t_k), Y^k)p(x(t_k)|Y^k) \\ &= \int p(Y(\tau)|x(\tau), x(t_k), Y^k)p(x(\tau), x(t_k)|Y^k)dx(\tau). \end{aligned} \tag{6}$$

Substituting (7) back into (5), yields

$$\begin{aligned} p(x(t_k)|Y^k, Y(\tau)) &= \frac{\int p(Y(\tau)|x(\tau), x(t_k), Y^k)p(x(\tau), x(t_k)|Y^k)dx(\tau)}{p(Y(\tau)|Y^k)} \\ &= \int \frac{p(Y(\tau)|x(\tau), x(t_k), Y^k)p(x(\tau), x(t_k)|Y^k)}{p(Y(\tau)|Y^k)}dx(\tau). \end{aligned}$$

Using the inverse form of Bayes rule

$$p(x(t_k)|Y^k, Y(\tau)) = \int p(x(\tau), x(t_k)|Y^k, Y(\tau))dx(\tau). \tag{7}$$

It is thus clear that solving the OOSM problem involves *consideration of the joint density of the current target state and the target state corresponding to the delayed measurement.*

Generalizing this, the OOSM problem involving multiple delays can be stated as follows: Let the delayed measurements received at time t_k be denoted by $Y(\tau) = \{Y(\tau_1), Y(\tau_2), \cdots, Y(\tau_d)\}$, where $\tau_i < t_k, \forall\ i \in \{1, \cdots, d\}$ and τ_d is the time corresponding to the maximum time delay. Then the solution to the OOSM problem is to determine the density

$$\begin{aligned} p(x(t_k)|Y^k, Y(\tau)) = &\int_{x(\tau_1)} \int_{x(\tau_2)} \cdots \int_{x(\tau_d)} p(x(t_k), x(\tau_1), x(\tau_2), \cdots, x(\tau_d)|Y^k, Y(\tau)) \\ &dx(\tau_1), dx(\tau_2), \cdots, dx(\tau_d) \end{aligned} \tag{8}$$

which indicates that, in general, the solution involves a Bayes recursion for the joint probability density of an augmented state vector $\mathbf{X}_k = [x(t_k), x(\tau_1), \cdots, x(\tau_d)]^T$, i.e.,

$$p(x(t_k), x(\tau_1), x(\tau_2), \cdots, x(\tau_d)|Y^k, Y(\tau)) = p(\mathbf{X}_k|Y^k, Y(\tau)). \tag{9}$$

Consider a discrete time system where

$$\tau_1 = t_{k-1}, \tau_2 = t_{k-2}, \cdots, \tau_d = t_{k-d}$$

Note that, in general, the delayed measurement time τ need not correspond exactly to the times

$$t_{k-1}, t_{k-2}, \cdots, t_{k-d}$$

i.e., τ might fall within the intervals around $t_{k-i},\ i \in [1, d]$. This situation can be solved by approximating $\tau = t_k$. The error caused by this approximation is called the discretization error. Usually these errors are small in most practical systems.

From Bayes rule, equation (9) becomes

$$\begin{aligned} &p(x(t_k), x(t_{k-1}), \cdots, x(t_{k-d})|Y^k, Y(\tau)) \\ &= \frac{1}{\delta} p(Y(\tau)|x(t_k), x(t_{k-1}), \cdots, x(t_{k-d}), Y^k) p(x(t_k), x(t_{k-1}), \cdots, x(t_{k-d})|Y^k) \end{aligned} \tag{10}$$

for a normalizing constant δ.

Denoting $[x(t_k), \cdots, x(t_{k-d})]^T$ as $\mathbf{X}_k^d$, the Bayes recursion for (10) becomes

$$p(\mathbf{X}_k^d|Y^k, Y(\tau)) = \frac{1}{\delta} p(Y(\tau)|\mathbf{X}_k^d, Y^k) p(\mathbf{X}_k^d|Y^k). \tag{11}$$

If the two densities on the right hand side of (11) are Gaussian, then the posterior density on the left hand side of (11) is also Gaussian and the solution reduces to a standard KF with an augmented state [17].

In order to simplify the derivation/implementation of a number of algorithms proposed in this paper, it is desirable to include the current measurement along with the delayed measurements even if it is not delayed, and thus the measurement vector becomes

$$Y_k = [y(t_k), y(t_{k-1}), \cdots, y(t_{k-d})]^T. \tag{12}$$

Hence, by changing notation, (11) can be rewritten as

$$p(\mathbf{X}_k^d|Y^{k-1}, Y_k) = \frac{1}{\delta} p(Y_k|\mathbf{X}_k^d, Y^{k-1}) p(\mathbf{X}_k^d|Y^{k-1}). \tag{13}$$

Equation (13) is a fundamental relationship which leads to the development of the AS-KF algorithm. Now let us consider the realistic scenario of multi-sensor tracking. We illustrate the exposition of the Bayesian solution for the case of two sensors. If $y_i(t_k)$ is the sensor measurement from sensor $i = 1, 2$ at time t_k, then let $Y_i^k = \{y_i(j) : j = 1, ..., k\}$ be the set of sensor measurements up to time t_k generated by sensor i and $Y_k(i)$ be the set of measurements received at time k including the delayed measurements received at time k, thus

$$Y_i(t_k) = [y_i(t_k), y_i(t_{k-1}), \cdots, y_i(t_{k-d})]^T. \tag{14}$$

By invoking Bayes' rule at the central location,

$$\begin{aligned} &p(\mathbf{X}_k^d|Y_1^k, Y_2^k) \\ &= p(\mathbf{X}_k^d|Y_1(t_k), Y_2(t_k), Y_1^{k-1}, Y_2^{k-1}) \\ &= \frac{1}{\delta_{12}} p(Y_1(t_k), Y_2(t_k)|\mathbf{X}_k^d, Y_1^{k-1}, Y_2^{k-1}) \times p(\mathbf{X}_k^d|Y_1^{k-1}, Y_2^{k-1}), \end{aligned} \tag{15}$$

where δ_{12} is a normalising constant.

Owing to legacy systems or existing trackers and bandwidth constrained communication links, measurements are usually not available. However, summary of these measurements (i.e., track estimates) in the form of posterior probability density functions are available as shown in the Figure 1. In such a case track to track fusion at the fusion center can be performed using the augmented state approach as follows. Let the augmented state at time t_k at the fusion center be $\mathbf{X}(t_k) \triangleq [x(t_k), \cdots, x(t_{k-r})]^T$. Thus, the fusion problem reduces to the problem of finding a solution to the joint density $p(x(t_k), \cdots, x(t_{k-r})|Y_1^k, Y_2^{k-r})$. Using Bayes' rule, we have

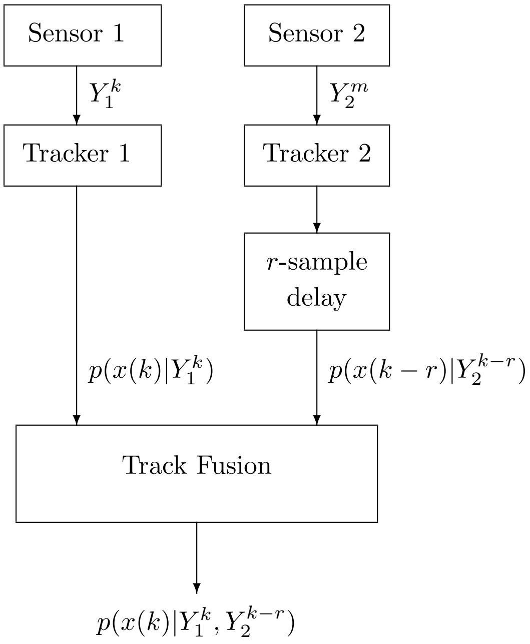

FIG. 1. *A distributed sensor fusion scenario.*

$$\begin{aligned}
&p(x(t_k), \cdots, x(t_{k-r})|Y_1^k, Y_2^{k-r})\\
&= \frac{1}{\delta} p(y_1(t_k), y_2(t_{k-r})|\mathbf{X}_k^d, Y_2^{k-r-1})\\
&\quad \times p(\mathbf{X}_k^d|Y_1^{k-1}, Y_2^{k-r-1})\\
&= \frac{1}{\delta} p(y_1(t_k)|\mathbf{X}_k^d p(y_2(t_{k-r})|\mathbf{X}(t_k))\\
&\quad \times p(\mathbf{X}_k^d|Y_1^{k-1}, Y_2^{k-r-1})\\
&= \frac{1}{\delta^*} p(u_1(t_k)|\mathbf{X}_k^d p(u_2(t_{k-r})|\mathbf{X}_k^d)\\
&\quad \times p(\mathbf{X}_k^d|Y_1^{k-1}, Y_2^{k-r-1})\\
&= \frac{1}{\delta^*} p(u_1(t_k), u_2(t_k)|\mathbf{X}_k^d) p(\mathbf{X}_k^d|Y_1^{k-1}, Y_2^{k-r-1}),
\end{aligned} \tag{16}$$

where δ^* and δ are normalizing constants. In obtaining the last equation of (16), we use the result in (74) obtained in the Appendix. Under Gaussian assumptions, a solution to (16) can be obtained using the Augmented State Kalman Filter (AS-KF) structure at the fusion node [3]. Let $\mathbf{Y}(t_k) = [u_1(t_k), u_2(t_{k-r})]^T$, be the set of equivalent measurements received from the two sensors in the network and the system dynamics involving the augmented past states. The equivalent measurements are given by

$$\begin{aligned} u_i(t_k) &= A^{-1}(t_k)b(t_k) \\ &= \Big[P_i^{-1}(t_{k|k}) - P_i^{-1}(t_{k|k-1})\Big]^{-1} \\ &\quad \Big[P_i^{-1}(t_{k|k})\hat{x}_i(t_{k|k}) - P_i^{-1}(t_{k|k-1})\hat{x}_i(t_{k|k-1})\Big] \end{aligned} \tag{17}$$

$$U_i(t_k) = A^{-1} = \Big[P_i^{-1}(t_{k|k}) - P_i^{-1}(t_{k|k-1})\Big]^{-1}. \tag{18}$$

Thus, in centralized fusion one uses the combined measurement vector formed by augmenting the measurements into $\mathbf{Y}(t_k)$. On the other hand, in distributed fusion one uses the combined measurement vector formed by augmenting the equivalent measurements into $\mathbf{Y}(t_k)$.

3. Implementable Solutions to OOSM Problems.

3.1. OOSM with Single Delay.

3.1. OOSM with Single Delay. An exact solution for the OOSM problem with single delay, proposed in [1] and referred to as the Y-algorithm, considers the process noise between the time of the delayed measurement and the current time, and the correlation of the current state with the delayed measurement $y(\tau)$. The key contribution of this approach, when compared to other suboptimal approaches such as [4, 6, 2], is consideration of the non-zero mean of the process noise which modifies the expression for the backward predicted state $\hat{x}(\tau)$.

3.1.1. Y–Algorithm. This approach assumes that the measurement delay is less than one sampling period, i.e., $t_{k-1} \le \tau < t_k$. Define a joint Gaussian random variable $z(t_k)$

$$z(t_k) = \begin{bmatrix} x(t_k) \\ y(\tau) \end{bmatrix} \quad \text{with covariance} \quad P_z = \begin{bmatrix} P_{xx} & P_{xy} \\ P_{yx} & P_{yy} \end{bmatrix}, \tag{19}$$

where

$$P_{xx} = E[(x(t_k) - \hat{x}(t_{k|k}))(x(t_k) - \hat{x}(t_{k|k}))^T | Y^k] = P_{k|k}, \tag{20}$$

$$P_{yy} = E[(y(\tau) - \hat{y}(\tau))(y(\tau) - \hat{y}(\tau))^T | Y^k] = S_{\tau|k}, \tag{21}$$

$$P_{xy} = E[(x(t_k) - \hat{x}(t_{k|k}))(y(\tau) - \hat{y}(\tau))^T | Y^k] = P_{yx}^T. \tag{22}$$

The solution to the OOSM problem requires the conditional density $p(x(t_k)|y(\tau), Y^k)$. Using the results in [18], this density is known to be Gaussian with mean

$$\hat{x}(t_{k|\tau,k}) = \hat{x}(t_{k|k}) + P_{xy}P_{yy}^{-1}(y(\tau) - \hat{y}(\tau)) \tag{23}$$

and covariance

$$P(t_{k|\tau,k}) = P_{xx} - P_{xy}P_{yy}^{-1}P_{yx}, \tag{24}$$

where the backward predicted measurement is expressed as

$$\hat{y}(\tau) = H_\tau F_{\tau|k}[\hat{x}(t_{k|k}) - Q_k(\tau)H_\tau^T S_{\tau|k}^{-1}(y(t_k) - \hat{y}(t_{k|k-1}))]. \tag{25}$$

In this expression, H_τ is the observation matrix at time τ, $F_{\tau|k}$ is the system backward transition matrix[1] from t_k to τ, the last term, which is ignored in [4, 6, 2], accounts for the effect of process noise (with covariance $Q_k(\tau)$) on the estimate $\hat{x}(t_{k|k})$.

The cross covariance P_{xy} in (22) is given by

$$P_{xy} = [P_{k|k} - P_{x\tilde{y}}]F_{\tau|k}^T H_\tau^T, \tag{26}$$

where

$$P_{x\tilde{y}} \triangleq Cov\{x(t_k), w_k(\tau)|Y^k\} = Q_k(\tau) - P(t_{k|k-1})H_\tau^T S^{-1}(t_k)H_\tau P(t_{k|k-1}). \tag{27}$$

The Y-algorithm, as pointed out in [1], requires storage of the last innovation and can be interpreted as a type of non-standard smoothing.

3.1.2. Augmented State Kalman Filter. The OOSM problem with single delay assumes that the delayed measurement $y(\tau)$ is received at time t_k in addition to the current measurement $y(t_k)$. Let $[x(t_k), x(\tau)]^T$ be the augmented state. Then we can easily establish the following equations based on system and measurement equations,

$$\begin{bmatrix} x(t_k) \\ x(\tau) \end{bmatrix} = \begin{bmatrix} F_{t_k|\tau} & 0 \\ I & 0 \end{bmatrix} \begin{bmatrix} x(t_{k-1}) \\ x(\tau) \end{bmatrix} + \begin{bmatrix} w(t_{k|\tau}) \\ 0 \end{bmatrix}, \tag{28}$$

$$\begin{bmatrix} y(t_k) \\ y(\tau) \end{bmatrix} = \begin{bmatrix} H_k & 0 \\ 0 & H_\tau \end{bmatrix} \begin{bmatrix} x(t_k) \\ x(\tau) \end{bmatrix} + \begin{bmatrix} v(t_k) \\ v(\tau) \end{bmatrix}, \tag{29}$$

where $F_{t_k|\tau}$ is derived from the system dynamic equation, and $t_{k-1} = \tau$.

The standard KF recursion can then be employed to obtain the augmented state estimate which is updated using both delayed measurement $y(\tau)$ and current measurement $y(t_k)$. When there is no delayed measurement, the measurement equation becomes

$$y(t_k) = H_k x(t_k) + v(t_k). \tag{30}$$

[1] Although τ represents time, it also is used to indicate corresponding time index whenever no confusion arises.

In this approach, the correlation of target state and process noise is implicitly handled as opposed to the explicit handling in the Y-algorithm. In later sections, we show that this method can be easily extended to problems involving multiple delays and clutter.

3.2. OOSM with Multiple Delays. When at time t_k, multiple delayed measurements corresponding to known previous times, are received, the problem of updating the current state using these delayed measurements is called the OOSM problem with multiple delays. Mallick *et al.* [2] addressed this problem as summarized below and Mallick's algorithm is referred to as the M-algorithm.

3.2.1. M–Algorithm. The M-algorithm, proposed in [2, 19], extends the Y-algorithm, in an approximate way, to account for multiple delays. The key idea of this approach is to determine the cross covariance of (22) for each delayed measurement and at each time interval. By expressing the delayed measurement $y(\tau)$ as a function of the current state $x(t_k)$, the multiple lag OOSM problem can be solved by computing the cross covariance P_{xy} in a recursive manner for each time delayed measurement. For example, when the delay time τ is more than n sampling intervals, we have

$$\begin{aligned}P_{xy|n} &= -M_{k-n+1}Q(k-n+1,k;k-n+1,k)\\ &\quad -\sum_{i=1}^{n} M_{k-i+1}Q(k-i+1,k-i;k-i+1,k),\end{aligned} \tag{31}$$

where

$$M_{k-i+1} = \begin{cases} B_k, & i=1,\\ C_kC_{k-1}\cdots C_{k-i+2}B_{k-i+1}, & i=2,\cdots,n,\end{cases} \tag{32}$$

$$B_i = I - K_iH_i, \tag{33}$$

$$C_i = B_iF_{i-1|i} \tag{34}$$

and the covariance of process noise

$$\begin{aligned}&Q(k-i+1,k-i;k-i+1,k)\\ &\triangleq E\{w(k-i+1,k-i;k-i+1)w^T(k-i+1,k-i;k-i+1)\}.\end{aligned} \tag{35}$$

Clearly, in the calculation of the covariance in (32), one needs to evaluate the process noise from the time when measurement delay occurs to the current time and all time steps in between. One also has to evaluate all corresponding filter gains. The AS-KF solution presented below does not need to evaluate the process noise explicitly. All that is needed is an augmented state and the standard KF computing steps.

3.2.2. Augmented State Kalman Filter for the Multiple Delay OOSM Problem. For multiple delays, the measurement vector has the form (14), and the Bayes' recursion of (13) reduces to the AS-KF with an augmented state $\mathbf{X}_k^d$. The system dynamics model using the augmented state can be constructed based on methods in [20],

$$\begin{aligned} \mathbf{X}_k^d &= \mathbf{F}_k \mathbf{X}_k^d + \mathbf{W}_k, \\ \mathbf{Y}_k &= \mathbf{H}_k \mathbf{X}_k^d + \mathbf{V}_k, \end{aligned} \tag{36}$$

where

$$\mathbf{F}_k = \begin{bmatrix} F_{t_k|\tau} & 0 & \cdots & 0 & 0 \\ I & 0 & \cdots & 0 & 0 \\ 0 & \ddots & 0 & \vdots & \vdots \\ \vdots & 0 & \ddots & 0 & 0 \\ 0 & \cdots & 0 & I & 0 \end{bmatrix}. \tag{37}$$

F is the system transition matrix in discrete form, the observation matrix is

$$\mathbf{H}_k = \begin{bmatrix} H_k & 0 & \cdots & 0 \\ 0 & H_{\tau_1} & 0 & \vdots \\ \vdots & \cdots & \ddots & 0 \\ 0 & \cdots & 0 & H_{\tau_d} \end{bmatrix} \tag{38}$$

and the noise covariance matrix

$$\mathbf{R}_k = \begin{bmatrix} R_k & 0 & \cdots & 0 \\ 0 & R_{\tau_1} & 0 & \vdots \\ \vdots & \cdots & \ddots & 0 \\ 0 & \cdots & 0 & R_{\tau_d} \end{bmatrix}. \tag{39}$$

The predicted density and the likelihood are given by

$$p(\mathbf{X}_k^d | \mathbf{Y}^{k-1}) = \mathcal{N}(\mathbf{X}_k^d; \widehat{\mathbf{X}}_{k|k-1}^d, \mathbf{P}_{k|k-1}^d), \tag{40}$$

$$p(\mathbf{Y}_k | \mathbf{X}_k^d, \mathbf{Y}^{k-1}) = \mathcal{N}(\mathbf{Y}_k; \mathbf{H}_k \widehat{\mathbf{X}}_{k|k-1}^d, \mathbf{S}_k^d), \tag{41}$$

and the updated density [17] is given by

$$p(\mathbf{X}_k^d | \mathbf{Y}^k) = \mathcal{N}(\mathbf{X}_k^d; \widehat{\mathbf{X}}_{k|k}^d, \mathbf{P}_{k|k}^d) \tag{42}$$

with mean and covariance

$$\widehat{\mathbf{X}}_{k|k}^d = \widehat{\mathbf{X}}_{k|k-1}^d + \mathbf{K}_k \widetilde{\mathbf{Y}}_k, \tag{43}$$

$$\mathbf{P}_{k|k}^d = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1}^d, \tag{44}$$

where the innovation is

$$\widetilde{\mathbf{Y}}_k = \mathbf{Y}_k - \mathbf{H}_k \widehat{\mathbf{X}}^d_{k|k-1} \tag{45}$$

with covariance

$$\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k \tag{46}$$

and Kalman gain matrix

$$\mathbf{K}_k = \mathbf{P}^d_{k|k-1} \mathbf{H}_k^T \mathbf{S}_k^{-1}. \tag{47}$$

Clearly, (40)-(47) are the standard Kalman filter equations for an augmented state space model. The Augmented State KF for single delay described previously is a special case. This approach solves the problem of incorporating delayed measurements and also provides smoothed outputs [20] which are not available in the Y-algorithm and M-algorithm.

4. Computational Issues. A naive implementation of the AS-KF is computationally inefficient. Significant computational gains can be achieved by focusing on some of the key properties of these filters. Based on these observations, we now propose efficient versions of the AS-KF. First we propose an iterative version of AS-KF, where the measurements corresponding to different time indices are used one at a time - not as an augmented measurement vector. The iterative AS-KF is identical to AS-KF in its optimality and it is more efficient. Then a variable dimension AS-KF (VDAS-KF) algorithm that augments only the essential past states as opposed to augmenting all the past states up to a maximum delay is presented. Compared to the standard AS-KF we expect VDAS-KF to be less accurate as it trades-off some of the smoothing benefits for reduced computational complexity.

4.1. Iterative AS-KF. Apart from direct computation via the augmented vector and associated matrix form, the AS-KF can also be implemented in a nested form, i.e., iteratively computing (43)-(47) using measurements corresponding to different delays independently. This is because the gain matrix is column independent with respect to the time indices, i.e.,

$$\mathbf{K}_k = \begin{bmatrix} K_k & K_{k-1} & \cdots & K_{k-d} \end{bmatrix}. \tag{48}$$

Then, given the measurement set $\mathbf{Y}_k = [y(t_k), y(t_{k-1}), \cdots, y(t_{k-d})]^T$ received at time t_k, the update equations of the AS-KF state estimate and its covariance are given by (43) and (44). These can be simplified into an iterative form given by

$$\widehat{\mathbf{X}}^d_{k|k} = \widehat{\mathbf{X}}^d_{k|k-1} + \sum_{i=k}^{k-d} \mathbf{K}^i \widetilde{\mathbf{Y}}^i, \tag{49}$$

$$\mathbf{P}^d_{k|k} = \mathbf{P}^d_{k|k-1} - \sum_{i=k}^{k-d} \mathbf{K}^i \mathbf{H}^i \mathbf{P}^d_{k|k-1}, \tag{50}$$

where $\mathbf{H}^i$ is given by (38) with zero everywhere except for the ith partition component. $\widetilde{\mathbf{Y}}^i$ is the same dimension as $\mathbf{Y}_k$ but produced by assuming only the ith component of $\mathbf{Y}_k$ is received and $\mathbf{K}^i$ is the Kalman gain produced using $\mathbf{H}^i$, i.e.,

$$\mathbf{K}^i = \mathbf{P}^d_{k|k-1}\mathbf{H}^{i^T}\mathbf{S}_k^{-1}. \tag{51}$$

The above equations provide an equivalent but efficient implementation of (43) and (44). The equivalence is shown in [14].

4.2. Variable Dimension Augmented State Kalman Filter. The AS-KF presented in Section 3 uses an augmented state involving past states up to the maximum delay. However, augmenting all the past states up to the maximum delay leads to increased computational complexity. A more efficient solution can be obtained by adaptively augmenting only essential past states and removing the states that need not be a part of the augmented state. This variable dimension AS-KF (VDAS-KF) is considered in this section.

The idea of the VDAS-KF is that the augmented state only carries the current state and the past state for which there was a missing measurement. The filter will reduce to a normal Kalman filter if there is no OOSM. The VDAS-KF processes measurements, as shown in Figure 2, in following four cases:

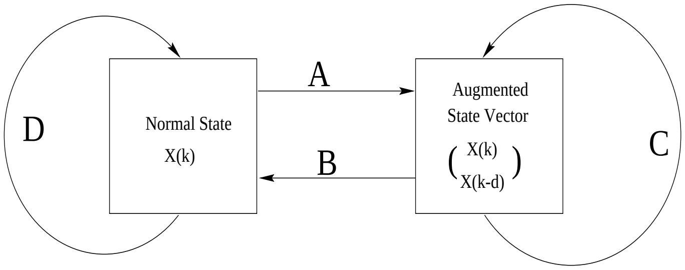

FIG. 2. *The process of an VDAS-KF*

- **Case A:** Target state is augmented if current measurement is delayed.
- **Case B:** The augmented state reduces to the standard state vector if the delayed measurement $y(\tau)$ is received and has been processed.
- **Case C:** The augmented state will keep its dimension unchanged when either
  - the delayed data $y(\tau)$ has not arrived yet; or
  - the delayed data has been received but the current data is delayed.
- **Case D:** No augmented state (standard KF) if there is no measurement delay.

For all the above cases changes to the state dimension will lead to changes in the associated covariance matrix.

The VDAS-KF algorithm is computationally more efficient than the standard AS-KF. However it is more complicated to implement and has comparable tracking performance and computational load comparable with the Y-algorithm.

5. Simulation Results. In this section three numerical examples are presented. Algorithm performance for OOSM and OOST is studied. The tracking performance is characterized by the Root Mean Square (RMS) error over 500 Monte Carlo runs for each specific scenario. In Example 1, it is assumed that the OOSM has only one lag delay. The performance of the Y-algorithm, the VDAS-KF and a two lag AS-KF are compared. In Example 2, a more general scenario is considered where OOSM tracking with multiple delays is allowed and the performance is compared between the M-algorithm and the standard AS-KF. The OOST problem is presented in the final example, where the performance of the proposed equivalent measurement based AS-KF is investigated.

Note that the output of an augmented state filter can be taken either from the first component of the augmented state, or from the last component of the augmented state. The former involves "filtering " only and the corresponding algorithm is denoted using a suffix "F", while the later involves smoothing and is denoted using a suffix "S". For example, AS-KF2-F denotes an AS-KF with 2 lags and its output are taken from the first component of the augmented state.

5.1. Example 1. This numerical example has been extensively used in many target tracking algorithm comparisons such as [18, 1, 2, 19], because it involves the most commonly used motion model in tracking with values of the maneuvering indices that cover the entire motion range of practical interest.

The discrete time system equation is

$$x(t_k) = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x(t_{k-1}) + v(t_k) \tag{52}$$

where $T = 1$ is the sampling interval, $v(t_k)$ is a zero-mean white Gaussian noise with covariance

$$Cov\{v(t_k)\} = Q(t_k) = \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix} q \tag{53}$$

and the observation is given by

$$y(t_k) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t_k) + w(t_k) \tag{54}$$

where $w(t_k)$ is also a zero-mean white Gaussian noise with covariance

$$Cov\{w(t_k)\} = R(t_k) = 1. \tag{55}$$

Based on the above model, we have developed a 2-dimensional target state model which is used for simulation. The maneuvering index is defined (in [21]) by

$$\lambda = \sqrt{\frac{qT^3}{R}}. \tag{56}$$

Table 1

Computational Comparison for Example 1

P_r	Y-Algorithm	VDAS-KF	AS-KF2-F	AS-KF2-S
0	1	1	5.57	5.57
0.25	2.26	2.60	5.57	5.57
0.5	2.30	2.68	5.57	5.57
0.75	4.41	5.47	5.57	5.57

Following [1], two cases (process noise $q = 0.1,$ and 4) corresponding to $\lambda = 0.3,$ and 2 are examined, i.e., the underlying target performs straight line motion, or is highly maneuvering. Data is generated randomly for each run starting with a initial state

$$x(0) = \begin{bmatrix} 200\,Km, & 0.5\,Km/Sec, & 100\,Km, & -0.08\,Km/Sec \end{bmatrix}. \tag{57}$$

A two data point method [18] is used to initialize the filters with

$$P(0|0) = \begin{bmatrix} P_0 & 0 \\ 0 & P_0 \end{bmatrix} \quad \text{where} \quad P_0 = \begin{bmatrix} R & R/T \\ R/T & 2R/T^2 \end{bmatrix} \tag{58}$$

for a priori error covariance or to form the initial error covariance for augmented state.

In this example, we assume that the OOSM can only have a maximum of one lag delay, and the data delay is uniformly distributed within the whole simulation period with a probability P_r that the current measurement is delayed.

Figures 3 and 4 show the simulation results for Example 1 where, the performance of the Y-algorithm, VDAS-KF and AS-KF2 are compared over 500 runs. A computational load comparison for these algorithms is listed in Table 1 in terms of number of floating point operations normalized to that of a standard Kalman filter.

The following observations can be drawn:

1. The Y-algorithm and VDAS-KF have similar RMS error performance within the whole range of the maneuvering indices.
2. The AS-KF2-F always outperforms both the Y-algorithm and VDAS-KF. When the probability of measurement delay (P_r) increases, this performance difference is observed to be greater.
3. The AS-KF2-S (smoothed AS-KF output) is superior to all other methods tested and has the least RMS error.
4. As shown in Table 1, the computational load of the VDAS-KF is comparable to the Y-algorithm and the AS-KF algorithm needs twice the computation.

5.2. Example 2. This example is the same as Example 1, except that multiple delays are allowed for an OOSM. The delayed measurement sequence is generated randomly by assuming they can be delayed by up to a maximum 3 sampling periods, i.e., $t_k - \tau_{max} = 3T$. The distribution of the delayed measurements is assumed to be

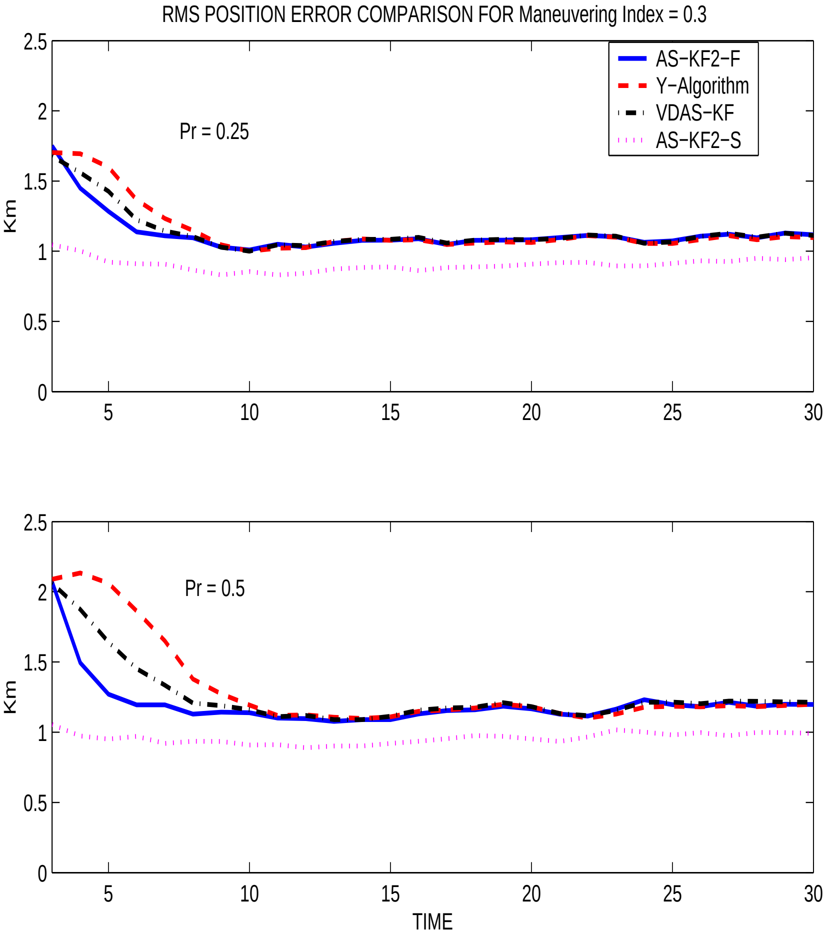

FIG. 3. *RMS performance comparison in the case of a straight line motion target* ($\lambda = 0.3$) *with single delay OOSM* ($P_r = 0.25$ *and* 0.5) *for 1) Y-algorithm, 2) VDAS-KF, 3) AS-KF2.*

uniform in 1, 2, 3 lags with a probability P_r that the measurements at time k will be delayed.

Simulation results are shown in Figures 5 and 6. It is observed that

1. As shown in Figure 5, for none maneuvering target tracking, both AS-KF4-F and M-algorithm have similar RMS performance regardless OOSM. In other words, the OOSM problem is not critical. This can also be seen from Figure 3 in Example 1.
2. AS-KF-F overall outperforms the M-algorithm, while the performance of AS-KF-S is better than both because AS-KF-S corrects all components of its augmented state vector using each delayed measurement rather than M-algorithm which can only make a correction to the current state.
3. For maneuvering target tracking ($\lambda = 2$), the average RMS error of the M-

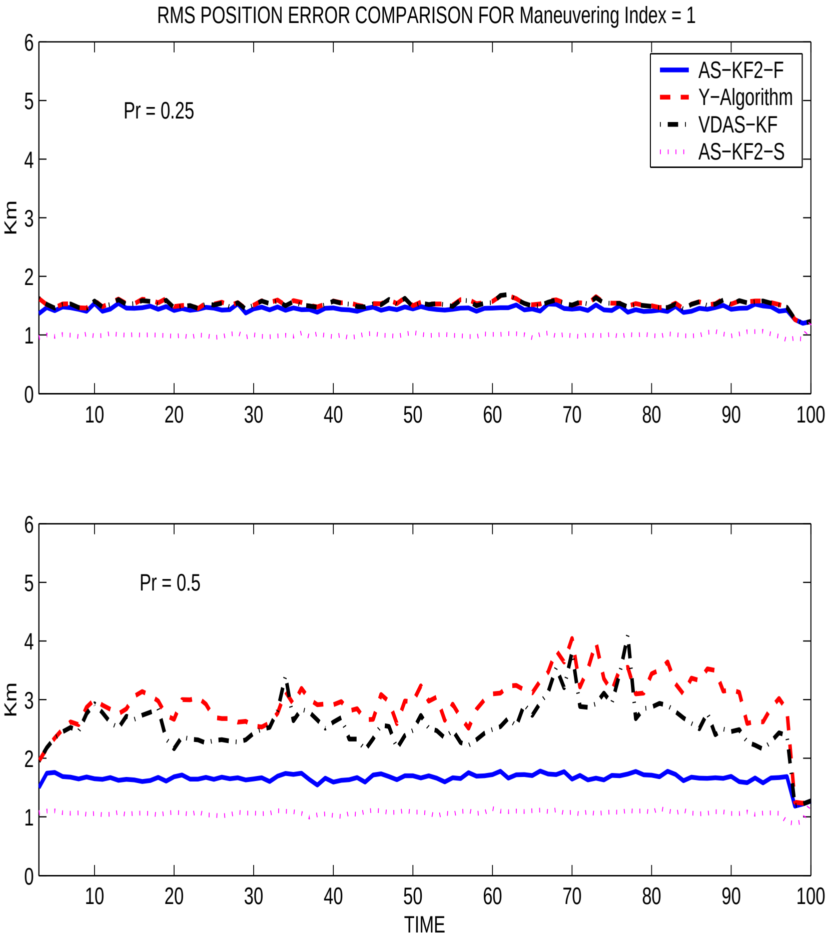

FIG. 4. *RMS performance comparison in the case of a highly maneuvering target ($\lambda = 2$) with single delay OOSM ($P_r = 0.25$ and 0.5) for 1) Y-algorithm, 2) VDAS-KF, 3) AS-KF2.*

algorithm is larger than AS-KF4. Such performance difference is large when the data delay probability (P_r) increases as shown in Figure 6.

4. The computational load of standard 4 lag AS-KF is about 11 times that of the M-algorithm in the case of $P_r = 0.25$. While the AS-KF remembers past states, the M-algorithm needs to compute past gain sequences and non-standard process noises in order to make a correction to the current state.

The performance of the M-algorithm as presented in [2], is expected to be improved if the conditional mean of the process noise is appropriately calculated.

5.3. Example 3. In our simulation, the scenario involves two local track nodes and a central fusion node. The central fusion node always receives track updates from Tracker 1 in real time. The root mean squared (RMS) error is used to compare the fusion performance in the following cases:

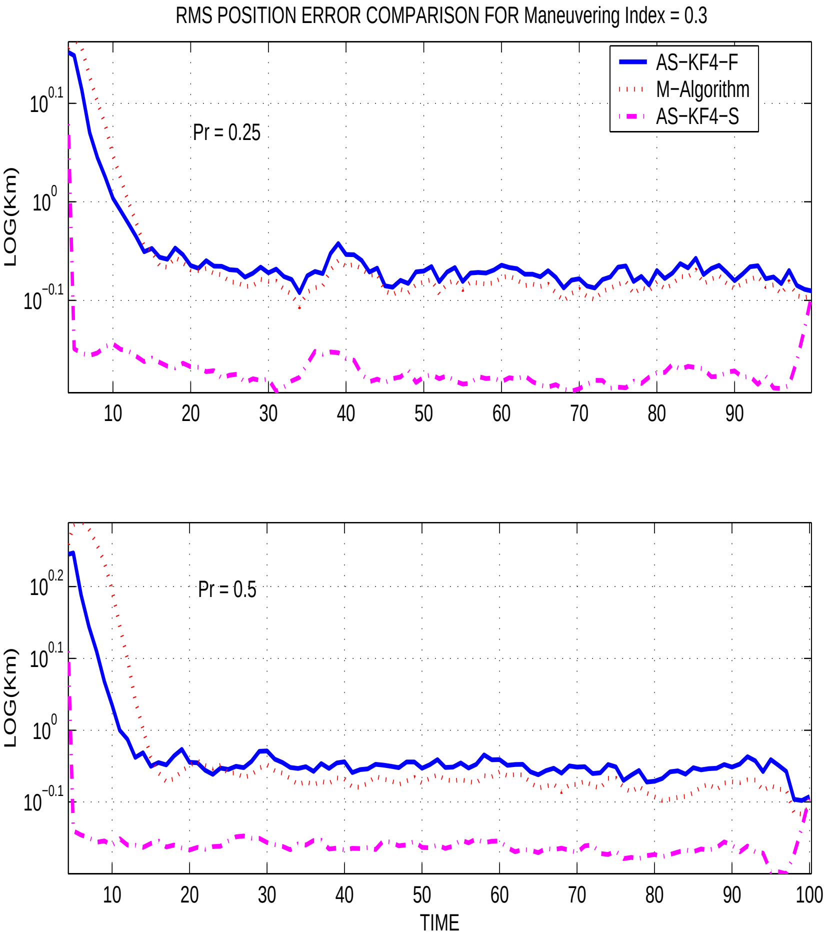

FIG. 5. *RMS performance comparison in the case of a straight line motion target ($\lambda = 0.3$) with multiple delay OOSM ($P_r = 0.25$ and 0.5) for 1) AS-KF4-F, 2) AS-KF4-S, 3) M-algorithm.*

1. The fusion node receives tracks from both trackers without delay.
2. The fusion node does not receive any tracks from Tracker 2, i.e., it treats the delayed tracks as missed information.
3. The fusion node receives delayed tracks from Tracker 2, with appropriate delay in receiving the measurements.

The basic target model is described by a discrete time system

$$x(t_k) = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x(t_{k-1}) + v(t_k) \tag{59}$$

where $T = 1$ is the sampling interval, $v(t_k)$ is a zero-mean white Gaussian process noise with covariance

$$Cov\{v(t_k)\} = Q(t_k) = \begin{bmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{bmatrix} q, \tag{60}$$

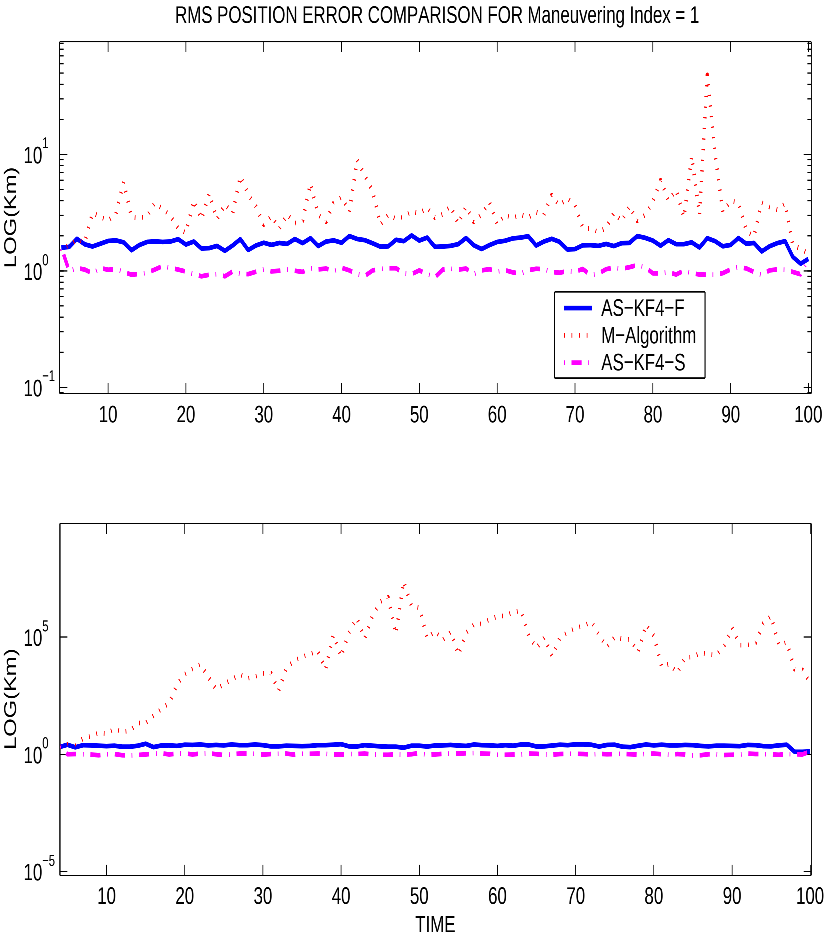

FIG. 6. *RMS performance comparison in the case of a highly maneuvering target ($\lambda = 2$) with multiple delay OOSM ($P_r = 0.25$ and 0.5) for 1) AS-KF4-F, 2) AS-KF4-S, 3) M-algorithm.*

with $q = 0.01$. The observation model for the sensors is given by

$$y_i(t_k) = Hx(t_k) + w_i(t_k), \qquad i \in 1, 2, \tag{61}$$

where w_i is a zero-mean white Gaussian process with covariance $Cov\{w_i(t_k)\} = R_i =$ 1

$$Cov\{w_i(t_k)\} = R_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad i \in 1, 2. \tag{62}$$

An initial state

$$x(0) = [200\,\text{km}, 0.5\,\text{km/s}] \tag{63}$$

is assumed for the tracker of sensor 1 and an initial state of

$$x(0) = [100\,\mathrm{km}, -0.08\,\mathrm{km/s}] \tag{64}$$

is assumed for the tracker of sensor 2. The two sensors are separated by about 100 km.

An ASKF is formed at the central fusion node with r states, where $r = 4$ and allows for a fixed time delay of 3 steps for the equivalent measurements of sensor 2, while it assumes no delay for sensor 1. The target state estimate $\hat{x}_i(t_{k|k})$ and its error covariance $P_i(t_{k|k})$ are obtained by each tracking node. These are then converted into the equivalent measurements $u_i(t_k)$ with variance $U_i(t_k)$ using (71) and (72) at the local nodes and are sent to the central fusion node (the ASKF). The equivalent measurement sequences from the tracking nodes consist of both position and velocity components. Thus, the measurement model required by the centralized fusion node (13) is

$$\begin{bmatrix} u_1(t_k) \\ u_2(t_{k-r}) \end{bmatrix} = \begin{bmatrix} \mathcal{H}_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathcal{H}_2 \end{bmatrix} \mathbf{X}_k^d + \begin{bmatrix} n_1(t_k) \\ n_2(t_{k-r}) \end{bmatrix} \tag{65}$$

where

$$\mathcal{H}_1 = \mathcal{H}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and n_1 and n_2 are independent, zero mean, white noise processes with covariances $U_1(t_k)$ and $U_2(t_{k-r})$ respectively. The system transition matrix at fusion node is given by (37).

The results from averaging 1000 Monte Carlo runs is shown in Figure 7. Through this simulation, the following observations can be made.

1. There is a significant performance gain when tracks are fused, compared to the one obtained from only one sensor and treating the delayed tracks as lost tracks.
2. The optimal fusion of a track that has been delayed by three time steps with an up-to-date track gives a result that is close to that obtained when there is zero delay.

6. Conclusion. In this paper, we formulate and solve the OOSM and OOST problems in the Bayesian framework. We establish that the solution involves the joint probability density of current and past states. Based on this, AS-KF is proposed as the key building block for solutions to these problems in the linear Gaussian case. Computationally efficient algorithms, like VDAS-KF are also presented in this paper. Simulation results are used to demonstrate the effectiveness of these algorithms. The

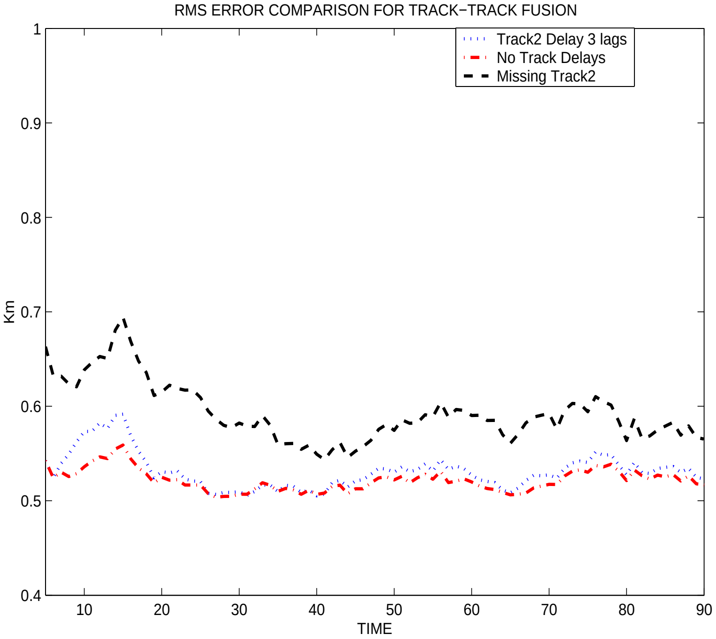

FIG. 7. *RMS error comparison for an ASKF fusion of two tracks when (1) there are no delays; (2) track 2 is missing; (3) track 2 has a delay of 3 lags.*

performance of the algorithms for the OOSM problem is compared with existing solutions in terms of RMS errors and computational complexity. There are no existing solutions to the OOST problem, hence the algorithm performance is evaluated under varying real-life conditions. The results show that the proposed OOSM solutions are computationally expensive when compared to some of the existing solutions but are straightforward to implement and yield significant performance improvements.

7. Appendix.

7.1. Track-to-Track Fusion and Equivalent Measurements. Let $x(t_k)$ be the target state at time t_k for a target that is visible to two non-collocated sensors 1 and 2. If $y_i(t_k)$ is the sensor measurement from sensor $i = 1, 2$ at time t_k, then let $Y_i^k = \{y_i(j) : j = 1, ..., k\}$ be the set of sensor measurements up to time t_k generated by sensor i. At the central location,

$$\begin{aligned} p(x(t_k)|Y_1^k, Y_2^k) &= p(x(t_k)|y_1(t_k), y_2(t_k), Y_1^{k-1}, Y_2^{k-1}) \\ &= \frac{1}{\delta_{12}} p(y_1(t_k)|x(t_k), Y_1^{k-1})\, p(y_2(t_k)|x(t_k), Y_2^{k-1}) \\ &\quad \times p(x(t_k)|Y_1^{k-1}, Y_2^{k-1}) \qquad (66) \\ &= \frac{1}{\delta_{12}} p(y_1(t_k)|x(t_k))\, p(y_2(t_k)|x(t_k)) \\ &\quad \times p(x(t_k)|Y_1^{k-1}, Y_2^{k-1}) \qquad (67) \end{aligned}$$

where δ_{12} is a normalising constant and (67) follows from (66) under the assumption of white measurement noise. If the network has insufficient bandwidth such that only track outputs $p(x(t_k)|Y_i^k)$, $i = 1, 2$, rather that sensor outputs $p(y_i(t_k)|x(t_k))$, $i = 1, 2$, can be transmitted to the fusion centre at each processing stage t_k, then the fusion algorithm must carry out track-to-track fusion based only on $p(x(t_k)|Y_i^k)$, $i = 1, 2$. Since registration is not considered to be a problem here, the target state variable employed by the fusion algorithm can be identical to that employed by the peripheral trackers. Thus, at sensor i,

$$\begin{aligned} p(x(t_k)|Y_i^k)) &= p(x(t_k)|y_i(t_k), Y_i^{k-1}) \\ &= \frac{1}{\delta_i} p(y_i(t_k)|x(t_k))\, p(x(t_k)|Y_i^{k-1}),\ i = 1, 2, \end{aligned} \tag{68}$$

where δ_i is a normalising constant. This equation together with equation (67) yields

$$\begin{aligned} p(x(t_k)|Y_1^k, Y_2^k) = \frac{\delta_1\delta_2}{\delta_{12}} & \frac{p(x(t_k)|Y_1^k)}{p(x(t_k)|Y_1^{k-1})} \\ & \times \frac{p(x(t_k)|Y_2^k)}{p(x(t_k)|Y_2^{k-1})}\, p(x(t_k)|Y_1^{k-1}, Y_2^{k-1}). \end{aligned} \tag{69}$$

Thus, at each time t_k, the fusion center requires the state estimate and state prediction from each sensor. Assuming Gaussian distributions,

$$p(x(t_k)|Y_i^{k-1}) = N[x(t_k); \hat{x}_i(t_{k|k-1}), P_i(t_{k|k-1})]$$

and

$$p(x(t_k)|Y_i^k) = N[x(t_k); \hat{x}_i(t_{k|k}), P_i(t_{k|k})]$$

at sensor $i = 1, 2$. The quotient of probability density functions from sensor i, for $i = 1, 2$, therefore takes the form

$$\begin{aligned} \frac{p(x(t_k)|Y_i^k)}{p(x(t_k)|Y_i^{k-1})} &= K \exp\Bigg\{-\frac{1}{2}\Bigg[\Big(x(t_k) - \hat{x}_i(t_{k|k})\Big)^T P_i^{-1}(t_{k|k})\Big(x(t_k) - \hat{x}_i(t_{k|k})\Big) \\ &\quad - \Big(x(t_k) - \hat{x}_i(t_{k|k-1})\Big)^T P_i^{-1}(t_{k|k-1})\Big(x(t_k) - \hat{x}_i(t_{k|k-1})\Big)\Bigg]\Bigg\} \\ &= K \exp\Bigg\{-\frac{1}{2}\Bigg[x^T(t_k)\Big(P_i^{-1}(t_{k|k}) - P_i^{-1}(t_{k|k-1})\Big)x(t_k) \\ &\quad - 2x^T(t_k)\Big(P_i^{-1}(t_{k|k})\hat{x}_i(t_{k|k}) - P_i^{-1}(t_{k|k-1})\hat{x}_i(t_{k|k-1})\Big) \\ &\quad + \hat{x}_i^T(t_{k|k})P_i^{-1}(t_{k|k})\hat{x}_i(t_{k|k}) - \hat{x}_i^T(t_{k|k-1})P_i^{-1}(t_{k|k-1})\hat{x}_i(t_{k|k-1})\Bigg]\Bigg\} \end{aligned}$$

$$
\begin{aligned}
&= K\exp\left\{-\frac{1}{2}\left[x^T(t_k)A(t_k)x(t_k) - 2x^T(t_k)b(t_k) + c(t_k)\right]\right\} \\
&= K\exp\left\{-\frac{1}{2}\left[\Big(x(t_k) - A^{-1}(t_k)b(t_k)\Big)^T A(t_k)\Big(x(t_k) - A^{-1}b(t_k)\Big)\right.\right. \\
&\qquad \left.\left. - b^T(t_k)A^{-1}(t_k)b(t_k) + c(t_k)\right]\right\}, \qquad (70)
\end{aligned}
$$

where

$$K = \frac{|2\pi P_i(t_{k|k-1})|^{\frac{1}{2}}}{|2\pi P_i(t_{k|k})|^{\frac{1}{2}}}.$$

Denoting

$$
\begin{aligned}
u_i(t_k) &= A^{-1}(t_k)b(t_k) \\
&= \Big[P_i^{-1}(t_{k|k}) - P_i^{-1}(t_{k|k-1})\Big]^{-1} \\
&\qquad \Big[P_i^{-1}(t_{k|k})\hat{x}_i(t_{k|k}) - P_i^{-1}(t_{k|k-1})\hat{x}_i(t_{k|k-1})\Big], \qquad (71) \\
U_i(t_k) &= A^{-1} = \Big[P_i^{-1}(t_{k|k}) - P_i^{-1}(t_{k|k-1})\Big]^{-1}, \qquad (72)
\end{aligned}
$$

(70) can be rewritten in the form

$$
\begin{aligned}
\frac{p(x(t_k)|Y_i^k)}{p(x(t_k)|Y_i^{k-1})} &= K\exp\left\{-\frac{1}{2}\left[\Big(x(t_k) - u_i(t_k)\Big)^T U_i^{-1}(t_k)\Big(x(t_k) - u_i(t_k)\Big)\right.\right. \\
&\qquad \left.\left. - b^T(t_k)A^{-1}(t_k)b(t_k) + c(t_k)\right]\right\} \\
&= K_1\exp\left\{-\frac{1}{2}\left[\Big(x(t_k) - u_i(t_k)\Big)^T U_i^{-1}(t_k)\Big(x(t_k) - u_i(t_k)\Big)\right]\right\} \\
&= \mathcal{N}\big(u_i(t_k); x(t_k), U_i(t_k)\big) \\
&= p(u_i(t_k)|x(t_k)). \qquad (73)
\end{aligned}
$$

Hence,

$$p(y_i(t_k)|x(t_k)) \propto \frac{p(x(t_k)|Y_i^k)}{p(x(t_k)|Y_i^{k-1})} \propto p(u_i(t_k)|x(t_k)). \qquad (74)$$

The variable $u_i(t_k)$ in equation (71) is the equivalent measurement vector from sensor i at time t_k and $U_i(t_k)$ in (72) is its covariance matrix. These expressions are identical to the equivalent measurements derived by inverting the Kalman filter equations when $H = I$, *i.e.*, when the equivalent measurements are expressed in state space variables [9]. It may be noted that these equivalent measurements are no longer correlated with each other and the target state at the fusion node. Hence they may be treated as conventional measurements in a standard KF. Substituting equation (73) into (69), the probability density function at the fusion centre takes the form

$$p(x(t_k)|Y_1^k, Y_2^k) = \frac{\delta_1 \delta_2}{\delta_{12}} \Big[\prod_{1=1}^{2} p(u_i(t_k)|x(t_k))\Big] p(x(t_k)|Y_1^{k-1}, Y_2^{k-1}). \tag{75}$$

This is the same as a centralised Kalman filter with equivalent measurements expressed in state space variables, *i.e.*, $u(t_k) = \left[u_1^T(t_k), u_2^T(t_k)\right]^T$ is the combined equivalent measurement vector,

$$U(t_k) = \begin{bmatrix} U_1(t_k) & 0 \\ 0 & U_2(t_k) \end{bmatrix}$$

is the associated covariance matrix, and $H = [I, I]^T$ is the measurement matrix, where I is the identity matrix with dimension equal to that of the state vector. In other words, $u_i(t_k)$ can be expressed in terms of the standard measurements equation

$$u_i(t_k) = Ix(t_k) + \eta_i(t_k). \tag{76}$$

This is equivalent to

$$u_i(t_k) = [I \;\; 0 \cdots 0\,]\mathbf{X}_k^d + [\eta_i(t_k) \;\; 0 \cdots 0\,]^T. \tag{77}$$

Thus

$$p(u_i(t_k)|x(t_k)) = p(u_i(t_k)|\mathbf{X}_k^d). \tag{78}$$

REFERENCES

[1] Y. Bar-Shalom, *Update with out-of-sequence measurements in tracking: Exact solution*, Signal and Data Processing of Small Targets 2000, Oliver E. Drummond, Ed., 2000, Proceedings of SPIE, Vol. 4080, pp. 541–556.

[2] M. Mallick, S. Coraluppi, and C. Carthel, *Advances in Asynchronous and Decentralized Estimation*, in: Proceedings of the 2001 IEEE Aerospace Conference, Big Sky MT, March 2001.

[3] S. Challa, R. Evans, and X. Wang, *Target Tracking in Clutter Using Time-Delayed Out-of-Sequence Measurements*, in: Proceedings of Defence Applications of Signal Processing (DASP), Sep., 2001.

[4] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, MA, 1999.

[5] G. D Marcus, *Tracking with measurements of uncertain origin and random arrival times*, M. S. Thesis, Dept. of Elec. Eng. & Comput. Sci., University of Connecticut, Storrs, USA, Sep. 1979.

[6] R. D. Hilton, D. A. Martin, and W. D. Blair, *Tracking with time delayed data in multi-sensor system*, Tech. Rep., NSWCDD/TR-93/351, Dahlgren, VA, USA, 1993.

[7] S. C. A. Thomopoulos and L. Zhang, *Decentralized filtering with random sampling and delay*, Information Sciences, 81(1994), pp. 117–131.

[8] O. Ravn, T. D. Larson, N. A. Andersen, and N. K. Poulsen, *Incorporation of time delayed measurements in a discrete time Kalman filter*, in: Conference on Decision and Control, Tampa, Florida, USA, 1998, Proceedings of the 37th IEEE, pp. 3972–3977, National Bureau of Standards.

[9] G. Frankel, *Multisensor tracking of balistic targets*, in: Proc. of SPIE, Vol. 2561, pp. 337–346, July 1995.

[10] O. Drummond, *Feedback in track fusion without process noise*, in: Proc. of SPIE, Vol. 2561, No. 6, pp. 369–383, July 1995.

[11] N. Okello and S. Challa, *Simultaneous registration and track fusion for networked trackers*, in: Proc. 4th International Conf. on Information Fusion, Montreal, Canada, Aug. 2001.

[12] N. Okello and S. Challa, *Joint sensor registration and track-to-track fusion for distributed trackers*, Submitted to IEEE Trans. AES, 2002.

[13] Y. Bar-Shalom and X.-R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*, ISSN 0895-9110, YBS Publishing, 1995.

[14] S. Challa, R. Evans, and X. Wang, *A Bayesian Solution to the OOSM Problem*, Accepted by Information and Fusion Journal, 2002.

[15] J. B. Moore, *Fixed-lag smoothing results for linear dynamical systems*, Australian Telecommunications Research, 7:2(1973), pp. 16–21.

[16] J. B. Moore and P. K. Tam, *Fixed-lag smoothing of nonlinear systems with discrete measurements*, Information Sciences, 6(1973), pp. 151–160.

[17] Y. C. Ho and R. C. K. Lee, *A Bayesian Approch to Problems in Stochastic Estimation and Control*, IEEE Trans. on Automatic Control, pp. 333–339, 1964.

[18] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*, Academic Press, 1988.

[19] M. Mallick, S. Coraluppi, and Y. Bar-Shalom, *Comparison of Out-of-sequence Measurement Algorithms in Multi-platform Target Tracking*, in: Proceedings of 4th Annual Conference on Information Fusion, Vol. II, pp. ThB1-11–18, Montreal, Quebec, Canada, August 2001.

[20] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, Prentice-Hall, NJ, 1979.

[21] Y. Bar-Shalom and X.-R. Li, *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, MA, 1993.