Tracking with Multisensor Out-of-Sequence Measurements with Residual Biases

SHUO ZHANG
YAAKOV BAR-SHALOM
GREGORY WATSON

In multisensor target tracking systems, measurements from different sensors on the same target typically exhibit biases. These biases can be accounted for as fixed random variables by the Schmidt-Kalman filter. Furthermore, measurements from the same target can arrive out of sequence. Recently, a procedure for updating the state with a multistep-lag "out-of-sequence" measurement (OOSM) using the simpler "1-step-lag" algorithm was developed for the situation without measurement biases. The present work presents the solution to the combined problem of handling biases from multiple sensors when their measurements arrive out of sequence. The state update with an OOSM is derived first for a KF tracker. This technique is then extended to the case where the tracker is an IMM estimator.

Manuscript received December 18, 2009; revised May 17, 2010 and November 1, 2010; released for publication November 17, 2010.

Refereeing of this contribution was handled by Stefano Coraluppi.

Research sponsored under ONR N00014-07-1-0131 and ARO W911NF-06-1-0467 grants.

Authors' addresses: S. Zhang and Y. Bar-Shalom, ECE Department, University of Connecticut, Storrs, CT 06269-2157, E-mail: (shuo.zhang@engr.uconn.edu); G. Watson, SPARTA, Arlington, VA 22209, E-mail: (Gregory.Watson@sparta.com).

1. INTRODUCTION

In multisensor target tracking systems, measurements from different sensors on the same target typically exhibit biases. These biases can be accounted for as fixed random variables by the Schmidt-Kalman filter. Furthermore, measurements from the same target can arrive out of sequence. Recently, a procedure for updating the state with a multistep-lag "out-of-sequence" measurement (OOSM) using the simpler "1-step-lag" algorithm was developed for the situation without measurement biases. The present work presents the solution to the combined problem of handling biases from multiple sensors when their measurements arrive out of sequence. The state update with an OOSM is derived first for a KF tracker. This technique is then extended to the case where the tracker is an IMM estimator.

The OOSM problem has been discussed in the literature starting with the initial work of [6] (discussed also in [5]), which presented an approximate solution to the problem of updating the current state of a target with an one-step-lag OOSM, called "algorithm B" in [2]. The optimal solution to the one-step-lag OOSM problem, called "algorithm A," was derived in [2]. It was also shown in [2] that algorithm B is nearly optimal for a one-step-lag OOSM. In [10], the comparison of algorithms A and B is discussed. In the case of receiving more that one OOSM in succession, one needs to modify algorithm A slightly (to preserve the optimality): in addition to updating the state at the current time, one also needs to update the state at the OOSM time using the standard Kalman updating algorithm. Alternatively, one can also stack multiple OOSMs in a single vector and use (the augmented version of) algorithm A to update the state with multiple OOSMs optimally in one step (see [17] for more details). In all these works it was assumed that the OOSM lag is less than a sampling interval. This has been designated as the "one-step-lag OOSM problem," and thus the corresponding algorithms can be called A1 and B1. The first solution to the general l-step-lag OOSM problem, Bl, was presented in [11] in the framework of B1. The algorithm Bl requires the storage of the sequence of filter gains and measurement matrices. The approach presented in [3] obtains the update with an l-step-lag OOSM in a *single step* (a "giant leap"), i.e., it generalized the previous algorithms to an arbitrary l. Furthermore, the resulting algorithms, Al1 and Bl1, have practically the same requirements as those of A1 and B1, respectively, for all $l > 1$. These algorithms have also been shown to perform nearly optimally in [3]. A general optimal solution to the OOSM problem was presented in [19], but it is substantially more complicated than [3].

A particle filter (PF) approach for dealing with OOSMs with arbitrary lags is proposed in [14], which presented a general solution to the nonlinear/non-Gaussian tracking problem in the presence of OOSMs. It was observed in [12] that, accuracywise, PF has no advantage over KF with converted Cartesian measurements or EKF, but takes much more CPU time. It was shown in [13] that the OOSM problem can be posed as a generalized smoothing or retrodiction problem and the Rauch-Tung-Streibel (RTS) smoother was used to obtain (in the linear case) an optimal algorithm for l-step-lag OOSM. Recently, a joint probability density approach called Accumulated State Density (ASD) is introduced in [8] with applications to the OOSM problem. By using ASD, the standard filtering and retrodiction are achieved in a unified manner. Rather than only updating the current state, ASD evaluates the effects of the OOSMs to all the states inside a certain window.

Section 2 presents the formulation of the OOSM problem for biased multiple sensors. Section 3 generalizes the Schmidt-Kalman filter (SKF), originally developed for tracking with a single sensor in the presence of residual biases, to the multisensor case. While the OOSM problem with biases from multiple sensors can be solved by augmenting the target state with all the sensor biases, this would not be practical for real systems. Section 4 derives the modified Joseph form for OOSM, which considers both the cases with and without biases. The combined problem of OOSM with biases from multiple sensors is solved in Section 5 using the Bl1 approach combined with the SKF *without state augmentation* resulting in the SKF/OOSM algorithm. These techniques are also described for the case where the tracker is an IMM estimator in Section 6. Section 7 discusses the heuristic "covariance inflation" approach for biases. The simulation results are given in Section 8. Section 9 presents a discussion of the results.

2. FORMULATION OF THE PROBLEM

The state of the system, x, of dimension n_x, is assumed to evolve from time t_{k-1} to time t_k according to

$$x(k) = F(k,k-1)x(k-1) + v(k,k-1) \tag{1}$$

where, using only the index of the time arguments, $F(k,k-1)$ is the state transition matrix to time t_k from time t_{k-1} and $v(k,k-1)$ is the (cumulative effect of the) process noise for this interval. The order of the arguments in both F and v follows here the convention for the transition matrices. Typically, the process noise has a single argument, but here two arguments will be needed for clarity.

The measurement equation is

$$z^{i(k)}(k) = H_x^{i(k)}(k)x(k) + w^{i(k)}(k) + H_b^{i(k)}(k)b^{i(k)}, \qquad i \in \{1,\ldots,N_S\} \tag{2}$$

where $i(k)$ is the index of the sensor which provided the measurement[1] from time t_k (the "time stamp"), $w^{i(k)}(k)$ is the corresponding measurement noise, modelled to be zero-mean, and $b^{i(k)}$ is the residual bias for this sensor. The dimension of the above measurement is n_{z^i} and the dimension of the bias in this measurement is denoted as n_i. The matrix H_b multiplying the bias has been discussed in [15] for various nonlinear measurements.

It is assumed that bias correction has been done separately (externally to the OOSM problem) following a sensor registration procedure. Consequently, the residual bias, assumed to be a *time-invariant random variable*, is zero-mean

$$E[b^i] = 0, \qquad i \in \{1,\ldots,N_S\} \tag{3}$$

and

$$\text{cov}[b^i,b^j] = E[b^i(b^j)'] = P_{b^i b^j}\delta_{ij}, \qquad i,j \in \{1,\ldots,N_S\} \tag{4}$$

where the shorter superscripts are used.

The noises are assumed zero-mean, white with covariances

$$\begin{aligned} E[v(k,j)v(k,j)'] &= Q(k,j) \\ E[w^{i(k)}(k)w^{i(k)}(k)'] &= R^{i(k)}(k) \end{aligned} \tag{5}$$

and, together with initial state error and the residual biases, mutually uncorrelated.

The time τ, at which the OOSM was made, is assumed to be such that

$$t_{k-l} < \tau < t_{k-l+1}. \tag{6}$$

This will require the evaluation of the effect of the process noise over an arbitrary noninteger number of sampling intervals. Note that $l = 1$ corresponds to the case where the lag is a fraction of a sampling interval; for simplicity this is called the "1-step-lag" problem, even though the lag is really a fraction of a time step.

The relationship between the current state $x(k)$ and the state observed by the OOSM is as follows. Similarly to (1), one has

$$x(k) = F(k,\kappa)x(\kappa) + v(k,\kappa) \tag{7}$$

where κ is the discrete time notation for τ. The above can be rewritten backward as

$$x(\kappa) = F(\kappa,k)[x(k) - v(k,\kappa)]. \tag{8}$$

where $F(\kappa,k) = F(k,\kappa)^{-1}$ is the backward transition matrix.

The problem is as follows: At time $t = t_k$ one has

$$\begin{aligned} \hat{x}(k \mid k) &\triangleq E[x(k) \mid Z^k] \\ P(k \mid k) &\triangleq \text{cov}[x(k) \mid Z^k] \end{aligned} \tag{9}$$

based on the (multisensor) cumulative set of measurements at t_k

$$Z^k \triangleq \{z^{i(\ell)}(\ell)\}_{\ell=1}^{k}. \tag{10}$$

[1]The superscript $i(k)$ will be shortened to i wherever this does not cause confusion.

Subsequently, the earlier measurement from time τ, denoted from now on with discrete time notation as κ,

$$z^{i(\kappa)}(\kappa) \stackrel{\Delta}{=} z^{i(\kappa)}(\kappa) = H_x^{i(\kappa)}(\kappa)x(\kappa) + w^{i(\kappa)}(\kappa) + H_b^{i(\kappa)}(\kappa)b^{i(\kappa)} \tag{11}$$

arrives after the state estimate (9) has been calculated. We want to update this estimate with the earlier measurement (11), namely, to calculate

$$\begin{aligned} \hat{x}(k \mid \kappa) &= E[x(k) \mid Z^{\kappa}] \\ P(k \mid \kappa) &= \text{cov}[x(k) \mid Z^{\kappa}] \end{aligned} \tag{12}$$

where

$$Z^{\kappa} \stackrel{\Delta}{=} \{Z^k, z^{i(\kappa)}(\kappa)\}. \tag{13}$$

This update should be done *without* reordering and reprocessing the measurements according to their time stamps.

3. THE MULTISENSOR SCHMIDT-KALMAN FILTER

This section presents the multisensor Schmidt-Kalman Filter (SKF) for the case of state estimation in the presence of residual biases but without OOSMs. The SKF procedure [16, 7] consists of augmenting the target state vector with the measurement bias vector, calculating the KF gain for this augmented state but then updating only the target state. While the bias is not updated, its covariance stays constant, but the crosscovariance between the bias and the state does change when the state is updated.

In the multisensor case there are, however, as many bias vectors as the number of sensors from which measurements are obtained. Consequently, the straightforward approach would be to augment the target state with all the biases and, while only the target state is updated, the entire updated covariance matrix of such an augmented state has to be calculated, yielding all the updated state-bias crosscovariances. This approach can be, however, very costly because of the possibly high dimension of the augmented state—typically 6 for the target state and with a minimum of 3 bias components from possibly as many as 10 sensors (not an unlikely scenario), one has at least a 36×36-dimensional covariance matrix to be updated. The major problem with this high-dimensional matrix occurs in the update with the OOSM, which requires the inversion of the augmented state covariance matrix (which is a full matrix), and this can be computationally expensive for real time implementation.

In the development below it is shown that one can augment the target state only with the bias of the sensor which provided the measurement to be used for the update and a "generic" other sensor. This allows to obtain the updated crosscovariances of the state with all the biases, block by block, rather than having to update the covariance matrix of the state augmented with all the biases. A similar procedure will be used in the update with the OOSM to avoid the need to invert a very large matrix. Furthermore, in the OOSM case, the inversion will have to be done only for the $(n_x \times n_x)$ state covariance matrix, without any augmentation.

Let the augmented state, of dimension $n_x + n_i + n_j$, be

$$\mathbf{x} \stackrel{\Delta}{=} \begin{bmatrix} x \\ b^i \\ \beta^j \end{bmatrix} \tag{14}$$

where i is the index of the sensor that provided the measurement to be used for the update at time k (the time argument of this index is now dropped for simplicity) and j is the index of a "generic" other sensor. The "generic" sensor bias β^j includes all the sensor biases except that from the current measurement, e.g.,

$$\beta^j = \begin{bmatrix} b^2 \\ b^3 \end{bmatrix}, \qquad b^i = b^1 \tag{15}$$

for the case $N_S = 3$ and the current measurement is from sensor 1. The use of a single notation β^j is just for simplicity. The state equation for this augmented state is

$$\mathbf{x}(k) = \mathbf{F}(k,k-1)\mathbf{x}(k-1) + \mathbf{v}(k,k-1) \tag{16}$$

where

$$\mathbf{F}(k,k-1) \stackrel{\Delta}{=} \begin{bmatrix} F(k,k-1) & 0 & 0 \\ 0 & I_{n_i} & 0 \\ 0 & 0 & I_{n_j} \end{bmatrix} \tag{17}$$

I_{n_i} denotes the $n_i \times n_i$ identity matrix and

$$\mathbf{v}(k,k-1) \stackrel{\Delta}{=} \begin{bmatrix} v(k,k-1) \\ 0 \\ 0 \end{bmatrix} \tag{18}$$

i.e., the biases are assumed constant between their (external) updates. The measurement at time k is

$$z^i(k) = \mathbf{H}^i(k)\mathbf{x}(k) + w^i(k) \tag{19}$$

where

$$\mathbf{H}^i(k) \stackrel{\Delta}{=} [H_x^i(k) \quad H_b^i(k) \quad 0]. \tag{20}$$

Let the prediction covariance of $\mathbf{x}(k)$ be

$$\begin{aligned} &\mathbf{P}(k \mid k-1) \\ &\stackrel{\Delta}{=} \begin{bmatrix} P_{xx}(k \mid k-1) & P_{xb^i}(k \mid k-1) & P_{x\beta^j}(k \mid k-1) \\ P_{xb^i}(k \mid k-1)' & P_{b^ib^i}(k \mid k-1) & 0 \\ P_{x\beta^j}(k \mid k-1)' & 0 & P_{\beta^j\beta^j}(k \mid k-1) \end{bmatrix} \\ &= \begin{bmatrix} P_{xx}(k \mid k-1) & P_{xb^i}(k \mid k-1) & P_{x\beta^j}(k \mid k-1) \\ P_{xb^i}(k \mid k-1)' & P_{b^ib^i} & 0 \\ P_{x\beta^j}(k \mid k-1)' & 0 & P_{\beta^j\beta^j} \end{bmatrix}. \end{aligned} \tag{21}$$

Then the optimal filter gain for updating $\mathbf{x}(k)$ is

$$\mathbf{W}^i(k)^{\mathrm{OPT}} = \mathbf{P}(k \mid k-1)\mathbf{H}^i(k)'S^i(k)^{-1}$$
$$= \mathbf{P}(k \mid k-1)\mathbf{H}^i(k)' \cdot [\mathbf{H}^i(k)\mathbf{P}(k \mid k-1)\mathbf{H}^i(k)' + R^i(k)]^{-1}$$
$$= \begin{bmatrix} W_x^i(k) \\ W_{b^i}^i(k) \\ W_{\beta^j}^i(k) \end{bmatrix} \tag{22}$$

which consists of three blocks.

The idea of the SKF is to use only the top block from the above, i.e., the actual gain will be

$$\mathbf{W}^i(k) = \begin{bmatrix} W_x^i(k) \\ 0 \\ 0 \end{bmatrix} \tag{23}$$

The expression of this block is

$$W_x^i(k) = [P_{xx}(k \mid k-1)H_x^i(k)' + P_{xb^i}(k \mid k-1)H_b^i(k)']S^i(k)^{-1} \tag{24}$$

where the innovation covariance is

$$S^i(k) = H_x^i(k)P_{xx}(k \mid k-1)H_x^i(k)' + H_x^i(k)P_{xb^i}(k \mid k-1)H_b^i(k)' + H_b^i(k)P_{b^ix}(k \mid k-1)H_x^i(k)' + H_b^i(k)P_{b^ib^i}H_b^i(k)' + R^i(k). \tag{25}$$

Since (23) is a suboptimal gain, the state covariance[2] update equation to be used in this case is the Joseph form (see, e.g., [1], Eq. (5.2.3-18)), which is the only one valid for an arbitrary gain. Thus, we have

$$\mathbf{P}(k \mid k) = [I_{n_x+n_i+n_j} - \mathbf{W}^i(k)\mathbf{H}^i(k)]\mathbf{P}(k \mid k-1) \cdot [I_{n_x+n_i+n_j} - \mathbf{W}^i(k)\mathbf{H}^i(k)]' + \mathbf{W}^i(k)R^i(k)\mathbf{W}^i(k)'. \tag{26}$$

Using (20), (21), and (23), the blocks of (26) are obtained as

$$\begin{aligned} P_{xx}(k \mid k) = {} & [I_{n_x} - W_x^i(k)H_x^i(k)]P_{xx}(k \mid k-1)[I_{n_x} - W_x^i(k)H_x^i(k)]' \\ & - W_x^i(k)H_b^i(k)P_{xb^i}(k \mid k-1)'[I_{n_x} - W_x^i(k)H_x^i(k)]' \\ & - [I_{n_x} - W_x^i(k)H_x^i(k)]P_{xb^i}(k \mid k-1)H_b^i(k)'W_x^i(k)' \\ & + W_x^i(k)H_b^i(k)P_{b^ib^i}H_b^i(k)'W_x^i(k)' + W_x^i(k)R^i(k)W_x^i(k)' \end{aligned} \tag{27}$$

$$P_{xb^i}(k \mid k) = [I_{n_x} - W_x^i(k)H_x^i(k)]P_{xb^i}(k \mid k-1) - W_x^i(k)H_b^i(k)P_{b^ib^i} \tag{28}$$

$$P_{x\beta^j}(k \mid k) = [I_{n_x} - W_x^i(k)H_x^i(k)]P_{x\beta^j}(k \mid k-1), \qquad \forall j \neq i(k) \tag{29}$$

$$P_{b^ib^i}(k) = P_{b^ib^i}(k-1) = P_{b^ib^i} \tag{30}$$

$$P_{\beta^j\beta^j}(k) = P_{\beta^j\beta^j}(k-1) = P_{\beta^j\beta^j} \tag{31}$$

$$P_{b^i\beta^j}(k) = 0. \tag{32}$$

The state update is done, in view of (23), according to

$$\hat{x}(k \mid k) = \hat{x}(k \mid k-1) + W_x^i(k)\nu^i(k) \tag{33}$$

where the innovation corresponding to $z^i(k)$ is

$$\nu^i(k) = z^i(k) - H_x^i(k)\hat{x}(k \mid k-1). \tag{34}$$

The prediction equations, based on the model (16) are the standard ones, namely,

$$\hat{x}(k \mid k-1) = F(k,k-1)\hat{x}(k-1 \mid k-1) \tag{35}$$

and for the covariance

$$\mathbf{P}(k \mid k-1) = \mathbf{F}(k,k-1)\mathbf{P}(k-1 \mid k-1)\mathbf{F}(k,k-1)' + \mathbf{Q}(k,k-1). \tag{36}$$

where

$$\mathbf{Q}(k,k-1) \stackrel{\Delta}{=} \mathrm{diag}[Q(k,k-1), 0_{n_i}, 0_{n_j}]. \tag{37}$$

The blocks of the prediction covariance (36) are calculated as

$$P_{xx}(k \mid k-1) = F(k,k-1)P_{xx}(k-1 \mid k-1)F(k,k-1)' + Q(k,k-1) \tag{38}$$

$$P_{xb^i}(k \mid k-1) = F(k,k-1)P_{xb^i}(k-1 \mid k-1) \tag{39}$$

$$P_{x\beta^j}(k \mid k-1) = F(k,k-1)P_{x\beta^j}(k-1 \mid k-1), \qquad \forall j \neq i(k). \tag{40}$$

Equations (28) and (29) yield the updated crosscovariances of the state with the bias in the measurement used in the update and with each bias in the other measurements, respectively. This procedure avoids having to handle the update of a potentially very large covariance matrix. The crosscovariance of the state with the bias in another sensor's measurement will be needed when that sensor's measurement becomes available for updating the state. Similarly, the predicted crosscovari-

[2]Actually this is not "state covariance" but "state-error covariance," since the state estimate is not the conditional mean any more due to the use of the suboptimal gain. However, for simplicity we still use the term "state covariance."

ances are obtained using (39) and (40) and they are the same for all the biases.

Thus, the above equations show how one can obtain the state estimate of the target accounting for all the biases in a multisensor situation, without resorting to state augmentation as far as the computations are concerned. The augmentation was used only to obtain the covariance matrix block updates.

4. MODIFIED JOSEPH FORM FOR OOSM

As discussed above, since the filter gain in the SKF is not optimal, the Joseph form should be used for covariance update. For an out-of-sequence measurement (OOSM), the time of the OOSM is not at the current time, so the Joseph should be modified accordingly. First, we consider the standard Joseph form. Then, the modified Joseph forms for OOSM are derived for the cases with and without residual biases.

4.1. Standard Joseph Form

The state model and measurement model are given by

$$x(k) = F(k)x(k-1) + v(k-1) \tag{41}$$

$$z(k) = H(k)x(k) + w(k) \tag{42}$$

where $v(k)$ and $w(k)$ are mutually independent white noise sequences with covariance $Q(k)$ and $R(k)$, respectively. The state estimate using a Kalman Filter is given by

$$\hat{x}(k \mid k) = \hat{x}(k \mid k-1) + W(k)\nu(k) \tag{43}$$

where W is the filter gain and ν is the innovation, which is given by

$$\begin{aligned} \nu(k) &= z(k) - H(k)\hat{x}(k \mid k-1) \\ &= H(k)x(k) + w(k) - H(k)\hat{x}(k \mid k-1). \end{aligned} \tag{44}$$

By substituting (44) into (43), the state estimate can be written as

$$\begin{aligned} \hat{x}(k \mid k) = \hat{x}(k \mid k-1) + W(k)H(k)[x(k) - \hat{x}(k \mid k-1)] \\ + W(k)w(k). \end{aligned} \tag{45}$$

Then, using (45) the estimation error at time k is given by

$$\begin{aligned} \tilde{x}(k \mid k) &= x(k) - \hat{x}(k \mid k) \\ &= [I - W(k)H(k)][x(k) - \hat{x}(k \mid k-1)] - W(k)w(k) \\ &= [I - W(k)H(k)]\tilde{x}(k \mid k-1) - W(k)w(k) \end{aligned} \tag{46}$$

where $\tilde{x}(k \mid k-1)$ is the prediction error. Thus, the error covariance $P(k \mid k)$ can be obtained as

$$\begin{aligned} P(k \mid k) &= \text{cov}\{\tilde{x}(k \mid k)\} \\ &= [I - W(k)H(k)]\text{cov}\{\tilde{x}(k \mid k-1)\} \\ &\quad \cdot [I - W(k)H(k)]' + W(k)\text{cov}\{w(k)\}W(k)' \\ &= [I - W(k)H(k)]P(k \mid k-1)[I - W(k)H(k)]' \\ &\quad + W(k)R(k)W(k)' \end{aligned} \tag{47}$$

due to the fact that the prediction error $\tilde{x}(k \mid k-1)$ is independent of the measurement noise $w(k)$. Formula (47) is known as the Joseph form.

4.2. Modified Joseph Form For OOSM Without Residual Biases

Now, we consider an OOSM $z(\kappa)$ $(\kappa < k)$. The most recent state estimate after receiving $z(\kappa)$ is given by [3]

$$\hat{x}(k \mid \kappa) = \hat{x}(k \mid k) + W(k,\kappa)\nu(\kappa) \tag{48}$$

where $\nu(\kappa)$ is the innovation at time κ of the OOSM, that is

$$\nu(\kappa) = z(\kappa) - H(\kappa)\hat{x}(\kappa \mid k). \tag{49}$$

Using the suboptimal technique B [5] (performed within 1% of the optimum when the OOSM has a one-step lag), the state retrodiction $\hat{x}(\kappa \mid k)$ is given by

$$\hat{x}(\kappa \mid k) = F(\kappa,k)\hat{x}(k \mid k) \tag{50}$$

and $\nu(\kappa)$ is obtained as

$$\begin{aligned} \nu(\kappa) &= z(\kappa) - H(\kappa)F(\kappa,k)\hat{x}(k \mid k) \\ &= H(\kappa)x(\kappa) + w(\kappa) - H(\kappa)F(\kappa,k)\hat{x}(k \mid k) \\ &= H(\kappa)F(\kappa,k)[x(k) - v(k,\kappa)] \\ &\quad + w(\kappa) - H(\kappa)F(\kappa,k)\hat{x}(k \mid k) \\ &= H(\kappa)F(\kappa,k)[x(k) - \hat{x}(k \mid k)] \\ &\quad - H(\kappa)F(\kappa,k)v(k,\kappa) + w(\kappa) \end{aligned} \tag{51}$$

which has made use of (8). Substituting (51) into (48), we have

$$\begin{aligned} \hat{x}(k \mid \kappa) = \hat{x}(k \mid k) + W(k,\kappa)H(\kappa)F(\kappa,k)[x(k) - \hat{x}(k \mid k)] \\ - W(k,\kappa)H(\kappa)F(\kappa,k)v(k,\kappa) + W(k,\kappa)w(\kappa). \end{aligned} \tag{52}$$

Thus, the estimation error is

$$\begin{aligned} \tilde{x}(k \mid \kappa) &= x(k) - \hat{x}(k \mid \kappa) \\ &= [I - W(k,\kappa)H(\kappa)F(\kappa,k)][x(k) - \hat{x}(k \mid k)] \\ &\quad + W(k,\kappa)H(\kappa)F(\kappa,k)v(k,\kappa) - W(k,\kappa)w(\kappa) \\ &= [I - W(k,\kappa)H(\kappa)F(\kappa,k)]\tilde{x}(k \mid k) \\ &\quad + W(k,\kappa)H(\kappa)F(\kappa,k)v(k,\kappa) - W(k,\kappa)w(\kappa). \end{aligned} \tag{53}$$

Using (53), the error covariance is given by

$$
\begin{aligned}
P(k \mid \kappa) &= \mathrm{cov}\{\tilde{x}(k \mid \kappa)\} \\
&= [I - W(k,\kappa)H(\kappa)F(\kappa,k)]\mathrm{cov}\{\tilde{x}(k \mid k)\} \\
&\quad \cdot [I - W(k,\kappa)H(\kappa)F(\kappa,k)]' \\
&\quad + W(k,\kappa)H(\kappa)F(\kappa,k)\mathrm{cov}\{v(k,\kappa)\} \\
&\quad \cdot [W(k,\kappa)H(\kappa)F(\kappa,k)]' \\
&\quad + W(k,\kappa)\mathrm{cov}\{w(\kappa)\}W(k,\kappa)' \\
&\quad + [I - W(k,\kappa)H(\kappa)F(\kappa,k)] \\
&\quad \cdot \mathrm{cov}\{\tilde{x}(k \mid k), v(k,\kappa)\} \\
&\quad \cdot [W(k,\kappa)H(\kappa)F(\kappa,k)]' \\
&\quad + W(k,\kappa)H(\kappa)F(\kappa,k)\mathrm{cov}\{v(k,\kappa),\tilde{x}(k \mid k)\} \\
&\quad \cdot [I - W(k,\kappa)H(\kappa)F(\kappa,k)]' \\
&= [I - W(k,\kappa)H(\kappa)F(\kappa,k)]P(k \mid k) \\
&\quad \cdot [I - W(k,\kappa)H(\kappa)F(\kappa,k)]' \\
&\quad + W(k,\kappa)H(\kappa)F(\kappa,k)Q(k,\kappa) \\
&\quad \cdot [W(k,\kappa)H(\kappa)F(\kappa,k)]' \\
&\quad + W(k,\kappa)R(\kappa)W(k,\kappa)' \\
&\quad + [I - W(k,\kappa)H(\kappa)F(\kappa,k)]P_{xv}(k,\kappa \mid k) \\
&\quad \cdot [W(k,\kappa)H(\kappa)F(\kappa,k)]' \\
&\quad + W(k,\kappa)H(\kappa)F(\kappa,k)P_{xv}(k,\kappa \mid k)' \\
&\quad \cdot [I - W(k,\kappa)H(\kappa)F(\kappa,k)]'
\end{aligned} \tag{54}
$$

due to the fact that the measurement noise $w(\kappa)$ of the OOSM is independent of the estimation error $\tilde{x}(k \mid k)$ and the process noise $v(k,\kappa)$. Note that, we have (as in [2] Eq. (22))

$$
\begin{aligned}
P_{xv}(k,\kappa \mid k) &= \mathrm{cov}\{\tilde{x}(k \mid k), v(k,\kappa)\} \\
&= \mathrm{cov}\{x(k), v(k,\kappa) \mid Z^k\}
\end{aligned} \tag{55}
$$

since the covariance is independent of the conditioning Z^k. Therefore, when OOSM is considered and the state estimation is given by the technique B, the Joseph form should be modified as in (54).

4.3. Modified Joseph Form For OOSM With Residual Biases

Next, we derive the Joseph form by considering both OOSM and residual biases of the sensors, i.e., (54) for the state augmented with the biases. Using (14), the state equation for this augmented state evolving from κ (the time of the OOSM) to the current time k is given by

$$
\mathbf{x}(k) = \mathbf{F}(k,\kappa)\mathbf{x}(\kappa) + \mathbf{v}(k,\kappa) \tag{56}
$$

where

$$
\mathbf{F}(k,\kappa) = \begin{bmatrix} F(k,\kappa) & 0 & 0 \\ 0 & I_{n_i} & 0 \\ 0 & 0 & I_{n_j} \end{bmatrix} \tag{57}
$$

and

$$
\mathbf{v}(k,\kappa) = \begin{bmatrix} v(k,\kappa) \\ 0 \\ 0 \end{bmatrix}. \tag{58}
$$

The corresponding covariance of $\mathbf{v}(k,\kappa)$ is

$$
\mathbf{Q}(k,\kappa) = \begin{bmatrix} Q(k,\kappa) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{59}
$$

The OOSM at time κ obtained from sensor i is

$$
z^i(\kappa) = \mathbf{H}^i(\kappa)\mathbf{x}(\kappa) + w^i(\kappa) \tag{60}
$$

where

$$
\mathbf{H}^i(\kappa) = [H_x^i(\kappa) \quad H_b^i(\kappa) \quad 0]. \tag{61}
$$

Let the updated covariance of $\mathbf{x}(k)$ be

$$
\mathbf{P}(k \mid k) = \begin{bmatrix} P_{xx}(k \mid k) & P_{xb^i}(k \mid k) & P_{x\beta^j}(k \mid k) \\ P_{xb^i}(k \mid k)' & P_{b^ib^i} & 0 \\ P_{x\beta^j}(k \mid k)' & 0 & P_{\beta^j\beta^j} \end{bmatrix} \tag{62}
$$

and the crosscovariance between $\mathbf{x}(k)$ and $\mathbf{v}(k,\kappa)$ be

$$
\mathbf{P}_{xv}(k,\kappa \mid k) = \begin{bmatrix} P_{xv}(k,\kappa \mid k) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{63}
$$

due to the independence between the sensor biases and process noise. The SKF gain using the OOSM $z(\kappa)$ at time k is

$$
\mathbf{W}^i(k,\kappa) = \begin{bmatrix} W_x^i(k,\kappa) \\ 0 \\ 0 \end{bmatrix}. \tag{64}
$$

Then, using the modified Joseph form given in (54), the covariance for the state augmented with residual biases can be written as

$$
\begin{aligned}
\mathbf{P}(k \mid \kappa) &= [I_{n_x+n_i+n_j} - \mathbf{W}^i(k,\kappa)\mathbf{H}^i(\kappa)\mathbf{F}(\kappa,k)]\mathbf{P}(k \mid k) \\
&\quad \cdot [I_{n_x+n_i+n_j} - \mathbf{W}^i(k,\kappa)\mathbf{H}^i(\kappa)\mathbf{F}(\kappa,k)]' \\
&\quad + \mathbf{W}^i(k,\kappa)\mathbf{H}^i(\kappa)\mathbf{F}(\kappa,k) \\
&\quad \cdot \mathbf{Q}(k,\kappa)[\mathbf{W}^i(k,\kappa)\mathbf{H}^i(\kappa)\mathbf{F}(\kappa,k)]' \\
&\quad + \mathbf{W}^i(k,\kappa)R^i(\kappa)\mathbf{W}^i(k,\kappa)' \\
&\quad + [I_{n_x+n_i+n_j} - \mathbf{W}^i(k,\kappa)\mathbf{H}^i(\kappa)\mathbf{F}(\kappa,k)]\mathbf{P}_{xv}(k,\kappa \mid k) \\
&\quad \cdot [\mathbf{W}^i(k,\kappa)\mathbf{H}^i(\kappa)\mathbf{F}(\kappa,k)]' \\
&\quad + \mathbf{W}^i(k,\kappa)\mathbf{H}^i(\kappa)\mathbf{F}(\kappa,k)\mathbf{P}_{xv}(k,\kappa \mid k)' \\
&\quad \cdot [I_{n_x+n_i+n_j} - \mathbf{W}^i(k,\kappa)\mathbf{H}^i(\kappa)\mathbf{F}(\kappa,k)]'.
\end{aligned} \tag{65}
$$

Using (56)–(64), the blocks of (65) are obtained as

$$
\begin{aligned}
P_{xx}(k \mid \kappa) = {}& [I_{n_x} - W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)]P_{xx}(k \mid k) \\
&\cdot [I_{n_x} - W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)]' \\
&- W_x^i(k,\kappa)H_b^i(\kappa)P_{xb^i}(k \mid k)' \\
&\cdot [I_{n_x} - W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)]' \\
&- [I_{n_x} - W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)] \\
&\cdot P_{xb^i}(k \mid k)H_b^i(\kappa)'W_x^i(k,\kappa)' \\
&+ W_x^i(k,\kappa)H_b^i(\kappa)P_{b^ib^i}H_b^i(\kappa)'W_x^i(k,\kappa)' \\
&+ W_x^i(k,\kappa)R^i(\kappa)W_x^i(k,\kappa)' \\
&+ W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)Q(k,\kappa) \\
&\cdot [W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)]' \\
&+ [I_{n_x} - W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)]P_{xv}(k,\kappa \mid k) \\
&\cdot [W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)]' \\
&+ W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)P_{xv}(k,\kappa \mid k)' \\
&\cdot [I_{n_x} - W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)]'.
\end{aligned} \tag{66}
$$

The crosscovariance of the state at k with the bias of the OOSM evolves as

$$
\begin{aligned}
P_{xb^i}(k \mid \kappa) = {}& [I_{n_x} - W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)]P_{xb^i}(k \mid k) \\
&- W_x^i(k,\kappa)H_b^i(\kappa)P_{b^ib^i}.
\end{aligned} \tag{67}
$$

The crosscovariance of the state at k with the other biases evolve as

$$
P_{x\beta^j}(k \mid \kappa) = [I_{n_x} - W_x^i(k,\kappa)H_x^i(\kappa)F(\kappa,k)]P_{x\beta^j}(k \mid k), \qquad \forall j \neq i(k) \tag{68}
$$

and the bias covariances stay unchanged, as in (30)–(32).

5. ONE-STEP ALGORITHM FOR MULTISTEP-LAG OOSM FOR MULTIPLE SENSORS WITH BIASES—SKF/OOSM

Using the approach of [3], one can perform in one step the update with an l-step-lag OOSM. Suitable modifications will be made to account for the fact that the measurements are biased. Two procedures, designated as Al1 and Bl1, were presented in [3] for the situation without biases. Both procedures retrodict the current state to the time of the OOSM, calculate the covariance of the retrodicted state, the retrodicted measurement and its the covariance, the crosscovariance between the current state and the retrodicted measurement and, with these, one can perform the direct update of the current state with the OOSM.

These algorithms are based on the 1-step-lag OOSM algorithms, designated in [2] as A and B, respectively. The difference between algorithms A and B is in the retrodiction of the current state to the time of the OOSM:

A) uses the exact conditional mean, which turns out to be an affine function of the current state estimate, with a second term being a linear transformation of the latest innovation;

B) uses a linear function of the current state estimate, which is the first term from the above.

Algorithm Al1, is similar to A but, using an (approximate) "equivalent measurement" for the measurements in the interval $[k-l+1,k]$, its second term is a linear transformation of the innovation corresponding to the equivalent measurement. Algorithm Bl1 uses, similarly to B, only the first term from Al1 and it does not need the equivalent measurement.

As shown in [3], both algorithms, while suboptimal, performed within 1% of the optimum obtained by reordering and reprocessing the measurements, which would not be practical in real systems. In view of their performance and the fact that, in the presence of biases, the statistical relationship between the "equivalent measurement" and the biases is difficult to quantify, the proposed approach is to modify Bl1 to account for the biases.

The suboptimal technique Bl1 [3] assumes the retrodicted noise to be zero. The retrodiction of the state to κ from k is[3]

$$
\hat{x}(\kappa \mid k) = F(\kappa,k)\hat{x}(k \mid k) \tag{69}
$$

i.e., a linear function of $\hat{x}(k \mid k)$, rather than an affine function. The covariance of this state retrodiction is

$$
\begin{aligned}
P_{xx}(\kappa \mid k) = {}& F(\kappa,k)[P_{xx}(k \mid k) + P_{vv}(k,\kappa \mid k) - P_{xv}(k,\kappa \mid k) \\
&- P_{xv}(k,\kappa \mid k)']F(\kappa,k)'
\end{aligned} \tag{70}
$$

where

$$
P_{vv}(k,\kappa \mid k) = Q(k,\kappa) \tag{71}
$$

$$
P_{xv}(k,\kappa \mid k) = P_{xx}(k \mid k)P_{xx}(k \mid k-l)^{-1}Q(k,\kappa) \tag{72}
$$

are the covariances of the process noise for the retrodiction interval and its crosscovariance with the current state, respectively. Equation (72) above follows by substituting in Equation (37) of [3] its preceding Equations (24) and (18) and simplifying the result.

The covariance of the retrodicted measurement, as given in Equation (39) of [3] for the situation *without biases*, is, assuming the OOSM is from sensor i, given

[3]The superscript ^{B} used in [3] to distinguish between the variables in algorithm versions A and B is dropped, since here we use only algorithm B.

by

$$S^i(\kappa) = H_x^i(\kappa)P(\kappa \mid k)H_x^i(\kappa)' + R^i(\kappa). \tag{73}$$

For the situation of the state augmented with biases, (73) is replaced by

$$\begin{aligned} S^i(\kappa) &= \mathbf{H}^i(\kappa)\mathbf{P}(\kappa \mid k)\mathbf{H}^i(\kappa)' + R^i(\kappa) \\ &= H_x^i(\kappa)P_{xx}(\kappa \mid k)H_x^i(\kappa)' + H_x^i(\kappa)P_{xb^i}(\kappa \mid k)H_b^i(\kappa)' \\ &\quad + H_b^i(\kappa)P_{b^ix}(\kappa \mid k)H_x^i(\kappa)' + H_b^i(\kappa)P_{b^ib^i}H_b^i(\kappa)' + R^i(\kappa) \end{aligned} \tag{74}$$

where, using (8), (69), one has

$$\begin{aligned} P_{xb^i}(\kappa \mid k) &= E\{[x(\kappa) - \hat{x}(\kappa \mid k)][b^i]'\} \\ &= E\{[F(\kappa,k)[x(k) - v(k,\kappa)] - F(\kappa,k)\hat{x}(k \mid k)][b^i]'\} \\ &= F(\kappa,k)P_{xb^i}(k \mid k) \end{aligned} \tag{75}$$

because the residual bias and the process noise are independent.

The crosscovariance between the state at k and the OOSM is, for the case without biases, given by Equation (40) of [3] as

$$P_{xz^i}(k,\kappa \mid k) = [P_{xx}(k \mid k) - P_{xv}(k,\kappa \mid k)]F(\kappa,k)'H_x^i(\kappa)'. \tag{76}$$

In the case with biases one has

$$\begin{aligned} P_{xz^i}(k,\kappa \mid k) &= E\{[x(k) - \hat{x}(k \mid k)][z^i(\kappa) - \hat{z}^i(\kappa \mid k)]'\} \\ &= E\{[x(k) - \hat{x}(k \mid k)][H_x^i(\kappa)x(\kappa) + w^i(\kappa) + H_b^i(\kappa)b^i - H_x^i(\kappa)F(\kappa,k)\hat{x}(k \mid k)]'\} \\ &= E\{[x(k) - \hat{x}(k \mid k)][H_x^i(\kappa)[F(\kappa,k)(x(k) - v(k,\kappa))] + w^i(\kappa) + H_b^i(\kappa)b^i - H_x^i(\kappa)F(\kappa,k)\hat{x}(k \mid k)]'\} \\ &= [P_{xx}(k \mid k) - P_{xv}(k,\kappa \mid k)]F(\kappa,k)'H_x^i(\kappa)' + P_{xb^i}(k \mid k)H_b^i(\kappa)'. \end{aligned} \tag{77}$$

Therefore, the gain for the update of the current state estimate with the OOSM $z^i(\kappa)$ in the presence of biases is (the first block of $\mathbf{W}^i(k,\kappa)^{\text{OPT}} = P_{\mathbf{x}z^i}(k,\kappa \mid k)S^i(\kappa)^{-1}$)

$$W_x^i(k,\kappa) = P_{xz^i}(k,\kappa \mid k)S^i(\kappa)^{-1} \tag{78}$$

with $P_{xz^i}(k,\kappa \mid k)$ given in (77) and $S^i(\kappa)$ given in (74).

The update with the OOSM $z^i(\kappa)$ of the most recent state estimate $\hat{x}(k \mid k)$ is thus

$$\hat{x}(k \mid \kappa) = \hat{x}(k \mid k) + W_x^i(k,\kappa)\nu^i(\kappa) \tag{79}$$

where the innovation corresponding to the OOSM $z^i(\kappa)$ is

$$\nu^i(\kappa) = z^i(\kappa) - \hat{z}^i(\kappa \mid k) \tag{80}$$

and the retrodicted OOSM is

$$\hat{z}^i(\kappa \mid k) = H_x^i(\kappa)\hat{x}(\kappa \mid k) \tag{81}$$

which uses the retrodicted state $\hat{x}(\kappa \mid k)$ given in (69). Using the filter gain given in (78) and the (approximate) crosscovariance in (72), the covariance for the state estimate and the crosscovariances of the state with the biases can be obtained from (66)–(68).

As it can be seen from (72), the need to invert the state covariance and the augmentation of the state with all the sensor biases would make the algorithm prohibitive for real-time implementation. The procedure presented above avoids the need to invert the augmented covariance matrix since it does not use any state augmentation.

6. THE IMM ESTIMATOR IN THE PRESENCE OF MEASUREMENT BIASES

As discussed above, one can carry out target state estimation with biased measurements from multiple sensors without augmenting the state with all the biases. This was shown in the context of Kalman filtering, i.e., when a single target motion model is used. Next, these results are extended to the case where multiple motion models are used and the tracking filter is an IMM estimator [1]. In this case, because of the biases, each of the r modules of the IMM will be an SKF.

6.1. Update With A Current Measurement

In the IMM algorithm, the first step is mixing. Using the augmented representation, the mixed state and mixed covariance are given by

$$\hat{\mathbf{x}}_m^0(k-1 \mid k-1) = \sum_{n=1}^{r} \hat{\mathbf{x}}_n(k-1 \mid k-1)\mu_{n|m}(k-1 \mid k-1) \tag{82}$$

$$\begin{aligned} \mathbf{P}_m^0(k-1 \mid k-1) &= \sum_{n=1}^{r} \mu_{n|m}(k-1 \mid k-1) \\ &\quad \cdot \{\mathbf{P}_n(k-1 \mid k-1) \\ &\quad + [\hat{\mathbf{x}}_n(k-1 \mid k-1) - \hat{\mathbf{x}}_m^0(k-1 \mid k-1)] \\ &\quad \cdot [\hat{\mathbf{x}}_n(k-1 \mid k-1) - \hat{\mathbf{x}}_m^0(k-1 \mid k-1)]'\} \end{aligned} \tag{83}$$

for $m = 1,\ldots,r$, where $\mu_{n|m}$ is the mixing probability [1], and the augmented state estimate $\hat{\mathbf{x}}_n(k-1 \mid k-1)$ and state covariance $\mathbf{P}_n(k-1 \mid k-1)$ matched to mode n are

$$\hat{\mathbf{x}}_n(k-1 \mid k-1) = \begin{bmatrix} \hat{x}_n(k-1 \mid k-1) \\ 0 \\ 0 \end{bmatrix} \tag{84}$$

$$\mathbf{P}_n(k-1 \mid k-1) = \begin{bmatrix} P_{x_n x_n}(k-1 \mid k-1) & P_{x_n b^i}(k-1 \mid k-1) & P_{x_n \beta^j}(k-1 \mid k-1) \\ P_{x_n b^i}(k-1 \mid k-1)' & P_{b^i b^i} & 0 \\ P_{x_n \beta^j}(k-1 \mid k-1)' & 0 & P_{\beta^j \beta^j} \end{bmatrix} \tag{85}$$

with the bias terms in (84) being zero in view of (23). Using (82)–(85), the blocks of (82) and (83) are obtained as

$$\hat{x}_m^0(k-1 \mid k-1) = \sum_{n=1}^{r} \hat{x}_n(k-1 \mid k-1)\mu_{n|m}(k-1 \mid k-1) \tag{86}$$

$$\begin{aligned} P^0_{x_m x_m}(k-1 \mid k-1) &= \sum_{n=1}^{r} \mu_{n|m}(k-1 \mid k-1) \\ &\quad \cdot \{P_{x_n x_n}(k-1 \mid k-1) \\ &\quad + [\hat{x}_n(k-1 \mid k-1) - \hat{x}^0_m(k-1 \mid k-1)] \\ &\quad \cdot [\hat{x}_n(k-1 \mid k-1) - \hat{x}^0_m(k-1 \mid k-1)]'\} \end{aligned} \tag{87}$$

$$P^0_{x_m b^i}(k-1 \mid k-1) = \sum_{n=1}^{r} \mu_{n|m}(k-1 \mid k-1) P_{x_n b^i}(k-1 \mid k-1) \tag{88}$$

$$P^0_{x_m \beta^j}(k-1 \mid k-1) = \sum_{n=1}^{r} \mu_{n|m}(k-1 \mid k-1) P_{x_n \beta^j}(k-1 \mid k-1) \quad \forall j \neq i(k). \tag{89}$$

The likelihood of mode m at time k is, assuming the mode-conditioned innovations to be Gaussian distributed (the common assumption [1]) is

$$\Lambda_m(k) = \mathcal{N}[\nu^i_m(k); 0, S^i_m(k)], \qquad m = 1,\ldots,r \tag{90}$$

where, with $\hat{x}_m(k \mid k-1)$ being the mode-m-conditioned predicted state, the innovation corresponding to mode m is, similarly to (34),

$$\nu^i_m(k) = z^i(k) - H^i_x(k)\hat{x}_m(k \mid k-1) \tag{91}$$

and the innovation covariance is, similarly to (25),

$$\begin{aligned} S^i_m(k) &= H^i_x(k) P_{x_m x_m}(k \mid k-1) H^i_x(k)' \\ &\quad + H^i_x(k) P_{x_m b^i}(k \mid k-1) H^i_b(k)' \\ &\quad + H^i_b(k) P_{b^i x_m}(k \mid k-1) H^i_x(k)' \\ &\quad + H^i_b(k) P_{b^i b^i} H^i_b(k)' + R^i(k) \end{aligned} \tag{92}$$

where $P_{x_m x_m}(k \mid k-1)$ and $P_{x_m b^i}(k \mid k-1)$ are the predicted state covariance and state-bias crosscovariance matched to mode m. In the above it is assumed that the measurement equations are the same for all the modes. The values of $\hat{x}_m(k \mid k-1)$, $P_{x_m x_m}(k \mid k-1)$, and $P_{x_m b^i}(k \mid k-1)$ are obtained from (35)–(39) using the mixed state estimate and mixed covariances given in (86)–(89).

Based on the mode likelihoods, the model probabilities at current time k, $\{\mu_m(k \mid k)\}_{m=1,\ldots,r}$, can be obtained [1], which are then used to calculate the combined state estimate and state covariance, namely,

$$\hat{\mathbf{x}}(k \mid k) = \sum_{m=1}^{r} \hat{\mathbf{x}}_m(k \mid k)\mu_m(k \mid k) \tag{93}$$

$$\begin{aligned} \mathbf{P}(k \mid k) &= \sum_{m=1}^{r} \mu_m(k \mid k) \\ &\quad \cdot \{\mathbf{P}_m(k \mid k) + [\hat{\mathbf{x}}_m(k \mid k) - \hat{\mathbf{x}}(k \mid k)] \\ &\quad \cdot [\hat{\mathbf{x}}_m(k \mid k) - \hat{\mathbf{x}}(k \mid k)]'\}. \end{aligned} \tag{94}$$

Similar to the mixing step, the blocks of (93) and (94) are obtained as

$$\hat{x}(k \mid k) = \sum_{m=1}^{r} \hat{x}_m(k \mid k)\mu_m(k \mid k) \tag{95}$$

$$\begin{aligned} P_{xx}(k \mid k) &= \sum_{m=1}^{r} \mu_m(k \mid k) \\ &\quad \cdot \{P_{x_m x_m}(k \mid k) + [\hat{x}_m(k \mid k) - \hat{x}(k \mid k)] \\ &\quad \cdot [\hat{x}_m(k \mid k) - \hat{x}(k \mid k)]'\} \end{aligned} \tag{96}$$

$$P_{xb^i}(k \mid k) = \sum_{m=1}^{r} \mu_m(k \mid k) P_{x_m b^i}(k \mid k) \tag{97}$$

$$P_{x\beta^j}(k \mid k) = \sum_{m=1}^{r} \mu_m(k \mid k) P_{x_m\beta^j}(k \mid k), \qquad \forall j \neq i(k) \tag{98}$$

where the values of $\hat{x}_m(k \mid k)$, $P_{x_m x_m}(k \mid k)$, $P_{x_m b^i}(k \mid k)$, and $P_{x_m\beta^j}(k \mid k)$ are obtained from (27)–(33) for each mode.

6.2. Update With An OOSM

Using the same notations—subscripting with m the mode-m-conditioned state estimates, covariances, innovations—the equations from Section 5 provide the procedure for using an OOSM in each module. The likelihood of each mode based on an OOSM will be, analogously to (90),

$$\Lambda_m(\kappa) = \mathcal{N}[\nu_m^i(\kappa); 0, S_m^i(\kappa)], \qquad m = 1,\ldots,r \tag{99}$$

where, with $\hat{x}_m(\kappa \mid k)$ being the mode-m-conditioned retrodicted state, the innovation corresponding to mode m is, similarly to (80),

$$\nu_m^i(\kappa) = z^i(\kappa) - H_x^i(\kappa)\hat{x}_m(\kappa \mid k) \tag{100}$$

where $\hat{x}_m(\kappa \mid k)$ is given by

$$\hat{x}_m(\kappa \mid k) = F_m(\kappa,k)\hat{x}_m(k \mid k). \tag{101}$$

The innovation covariance is, similarly to (74),

$$S_m^i(\kappa) = H_x^i(\kappa)P_{x_m x_m}(\kappa \mid k)H_x^i(\kappa)' + H_x^i(\kappa)P_{x_m b^i}(\kappa \mid k)H_b^i(\kappa)' + H_b^i(\kappa)P_{b^i x_m}(\kappa \mid k)H_x^i(\kappa)' + H_b^i(\kappa)P_{b^i b^i}H_b^i(\kappa)' + R^i(\kappa). \tag{102}$$

Using the likelihood function (99), the current mode probabilities updated with the OOSM are as in [4], namely,

$$\mu_m(k \mid \kappa) = \frac{1}{c}\left[\sum_{n=1}^{r}\Lambda_n(\kappa)\Pi_{mn}(\kappa,k)\right]\mu_m(k \mid k). \tag{103}$$

where the normalization constant is

$$c = \sum_{m=1}^{r}\sum_{n=1}^{r}\Lambda_n(\kappa)\Pi_{mn}(\kappa,k)\mu_m(k \mid k). \tag{104}$$

The mode transition probability $\Pi_{mn}(k_2,k_1)$ from time k_1 to k_2 is defined as

$$\Pi_{mn}(k_2,k_1) = P\{M(k_2) = n \mid M(k_1) = m\} \tag{105}$$

which is an element (row m, column n) in the transition matrix $\Pi(k_2,k_1)$. For $r = 2$, the transition matrix $\Pi(k_2,k_1)$ according to a continuous-time Markov chain is given by [4]

$$\Pi(k_2,k_1) = \frac{1}{\lambda_1+\lambda_2}\begin{bmatrix} \lambda_2 + \lambda_1 e^{-(\lambda_1+\lambda_2)T} & \lambda_1 - \lambda_1 e^{-(\lambda_1+\lambda_2)T} \\ \lambda_2 - \lambda_2 e^{-(\lambda_1+\lambda_2)T} & \lambda_1 + \lambda_2 e^{-(\lambda_1+\lambda_2)T} \end{bmatrix} \tag{106}$$

where $T = |t_{k_2} - t_{k_1}|$, and $1/\lambda_m$ is the expected sojourn time for mode m.

Similar to (95)–(98), the combined state estimate and covariances with the OOSM are given by

$$\hat{x}(k \mid \kappa) = \sum_{m=1}^{r}\hat{x}_m(k \mid \kappa)\mu_m(k \mid \kappa) \tag{107}$$

$$P_{xx}(k \mid \kappa) = \sum_{m=1}^{r}\mu_m(k \mid \kappa) \cdot \{P_{x_m x_m}(k \mid \kappa) + [\hat{x}_m(k \mid \kappa) - \hat{x}(k \mid \kappa)] \cdot [\hat{x}_m(k \mid \kappa) - \hat{x}(k \mid \kappa)]'\} \tag{108}$$

$$P_{xb^i}(k \mid \kappa) = \sum_{m=1}^{r}\mu_m(k \mid \kappa)P_{x_m b^i}(k \mid \kappa) \tag{109}$$

$$P_{x\beta^j}(k \mid \kappa) = \sum_{m=1}^{r}\mu_m(k \mid \kappa)P_{x_m\beta^j}(k \mid \kappa), \qquad \forall j \neq i(k) \tag{110}$$

where the values of $\hat{x}_m(k \mid \kappa)$, $P_{x_m x_m}(k \mid \kappa)$, $P_{x_m b^i}(k \mid \kappa)$, and $P_{x_m\beta^j}(k \mid \kappa)$ are obtained from (79) and (66)–(68) for each mode. Note that the mixing step is not carried out with the OOSM [4].

7. THE HEURISTIC "COVARIANCE INFLATION" APPROACH FOR BIASES

This approach increases the measurement noise variance by the variance of the biases (assumed to have mean zero). Note that this amounts to treating the biases as if they were an additional zero-mean white noise, which is clearly not correct. Only the SKF correctly treats the biases as fixed random variables. The reason this heuristic approach is considered here is that it has been used due to its simplicity, but as it will be shown, it yields inconsistent estimates.

Consider a measurement model given by

$$z(k) = h(x(k),b) + w(k) \tag{111}$$

which may be a nonlinear function of the target state and residual biases. The superscript i has been dropped for simplicity. Using the first-order Taylor expansion at $x(k) = \hat{x}(k \mid k-1)$ and $b = 0$, the measurement $z(k)$ can be approximated as

$$\begin{aligned} z(k) &\approx h(\hat{x}(k \mid k-1),0) + H_x(k)(x(k) - \hat{x}(k \mid k-1)) \\ &\quad + H_b(k)(b-0) + w(k) \\ &= h(\hat{x}(k \mid k-1),0) + H_x(k)\tilde{x}(k \mid k-1) \\ &\quad + H_b(k)b + w(k) \end{aligned} \tag{112}$$

where

$$H_x(k) = \left.\frac{\partial h(x,b)}{\partial x}\right|_{x(k)=\hat{x}(k|k-1),b=0} \tag{113}$$

$$H_b(k) = \left.\frac{\partial h(x,b)}{\partial b}\right|_{x(k)=\hat{x}(k|k-1),b=0} \tag{114}$$

TABLE I
Sensor Indices and Corresponding Time Stamps

Sensor Index	1	1	2	1	2	1	2	1	2	1	2	1	2	1
Time Stamp (s)	0	5	2.5	10	7.5	15	12.5	20	17.5	25	22.5	30	27.5	35

and $\hat{x}(k \mid k-1)$ is the state prediction at time k using the set of measurements z^{k-1}. The MMSE estimate of $z(k)$ is given by

$$\hat{z}(k \mid k-1) = E\{z(k) \mid z^{k-1}\} \approx h(\hat{x}(k \mid k-1), 0) \tag{115}$$

which has used the Taylor approximation given in (112) (this is actually the estimate using EKF). The corresponding error covariance, which has the same expression as in (25), is

$$\begin{aligned} S(k) &= \mathrm{cov}\{z(k) - \hat{z}(k \mid k-1)\} \\ &\approx \mathrm{cov}\{H_x(k)\tilde{x}(k \mid k-1) + H_b(k)b + w(k)\} \\ &= H_x(k)\mathrm{cov}\{\tilde{x}(k \mid k-1)\}H_x'(k) \\ &\quad + H_b(k)\mathrm{cov}\{b\}H_b'(k) + \mathrm{cov}\{w(k)\} \\ &\quad + H_x(k)\mathrm{cov}\{\tilde{x}(k \mid k-1), b\}H_b'(k) \\ &\quad + H_b(k)\mathrm{cov}\{b, \tilde{x}(k \mid k-1)\}H_x'(k) \end{aligned} \tag{116}$$

due to the fact that the measurement noise $w(k)$ is independent of the prediction error $\tilde{x}(k \mid k-1)$ and residual biases b. If the crosscovariances between the estimation error and residual biases are set to be zero, $S(k)$ can be written as

$$\begin{aligned} S(k) &= H_x(k)\mathrm{cov}\{\tilde{x}(k \mid k-1)\}H_x'(k) + H_b(k)\mathrm{cov}\{b\}H_b'(k) \\ &\quad + \mathrm{cov}\{w(k)\} \end{aligned} \tag{117}$$

which amounts to a covariance inflation with the inflation term $H_b(k)\mathrm{cov}\{b\}H_b'(k)$, similar to that in [18]. The effect of ignoring the crosscovariances will be evaluated in the simulation results.

8. SIMULATION RESULTS

8.1. Example 1: One-Dimensional Motion With Position Measurement Only

The target starts at origin and moves with a constant velocity of 10 m/s along the x-axis. The power spectrum density (PSD) of the process noise is $q = 0.5\ \mathrm{m}^2/\mathrm{s}^3$. Two sensors are used, which are located at $(-50, 0)$ km and $(50, 0)$ km. The (unaugmented) target state is denoted by $\mathbf{x}$ as

$$\mathbf{x} = [x \quad \dot{x}] \tag{118}$$

and the measurement model is

$$z^i = (1 + \alpha^i)[1 \quad 0](\mathbf{x} - \mathbf{x}_p^i) + \Delta^i + w^i, \qquad i = 1, 2 \tag{119}$$

TABLE II
Bias Standard Deviations for Position Measurement (Example 1)

Bias Level	Offset Bias Δ	Scale Bias α
Small	10 m ($= \sigma_w$)	10^{-4}
Large	20 m ($= 2\sigma_w$)	2×10^{-4}

where Δ denotes an offset bias and α denotes a scale (multiplicative) bias. The superscript i is the sensor index and $\mathbf{x}_p^i$ is the state of the ith sensor. The measurement noise s.d. is $\sigma_w = 10$ m for both sensors. The sampling interval for each sensor is 5 s, but they are not synchronized. The times at which the measurements are taken (their "time stamps") and the order of the measurements arriving at the fusion center are shown in Table I, where sensor 2 has all its measurements delayed with 1 step lag.

Two levels of biases are considered with the bias s.d. given in Table II (the same for both sensors).

For each bias level, two options are considered:

1. Reorder the measurements (in-sequence data).
2. Process OOSM.

The option of ignoring OOSMs has been shown in [3] to lead to significant performance loss.

For each scenario, we compare three filters: Kalman filter[4] without covariance inflation (KFwoINF), Kalman filter with covariance inflation (KFwINF), and SKF (Schmidt-KF). Each of these is modified appropriately when processing OOSM. The modified SKF to process OOSM is the SKF/OOSM from Section 5.

The discretized CWNA model (DCWNA) [1] is used as the target's dynamic model. Since the smallest time interval between the time stamps shown in Table I is 2.5 s, we use $T = 2.5$ s as the filter's sampling interval. The results below are based on 1000 Monte Carlo simulations. The two-sided 99% probability region of the NEES (normalized estimation error squared, [1] Section 5) based on the χ^2_{2000} distribution [1] is $[1.84, 2.16]$, marked by two dashed lines in the figures.

All filters were initialized with "one point" initialization (see [1], Section 5) according to which the first position measurement is used as the initial estimate and the initial velocity estimate is set to zero; the standard deviation of the latter is set at half the maximum speed in each coordinate.

The RMSE at the times when OOSMs are processed (at the time stamps of sensor 1 in our case) in Figs. 3

[4]Here, "Kalman filter" may also refer to extended Kalman Filter for nonlinear cases. We use the same acronym "KF" for simplicity.

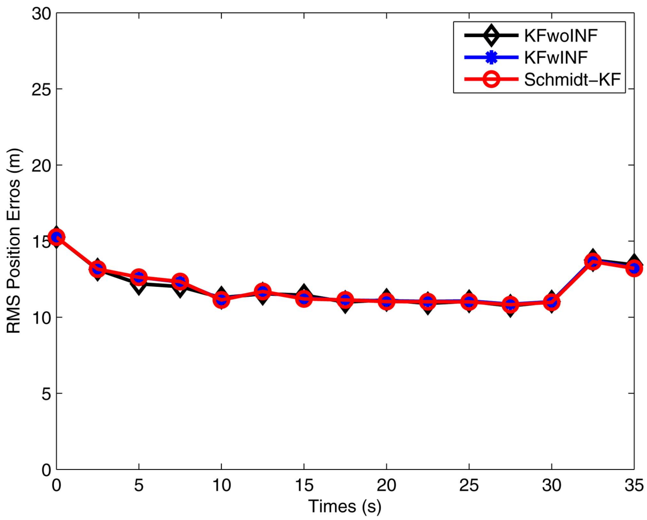


Fig. 1. Position RMSE for target with position measurement only (small bias, reordering measurements).

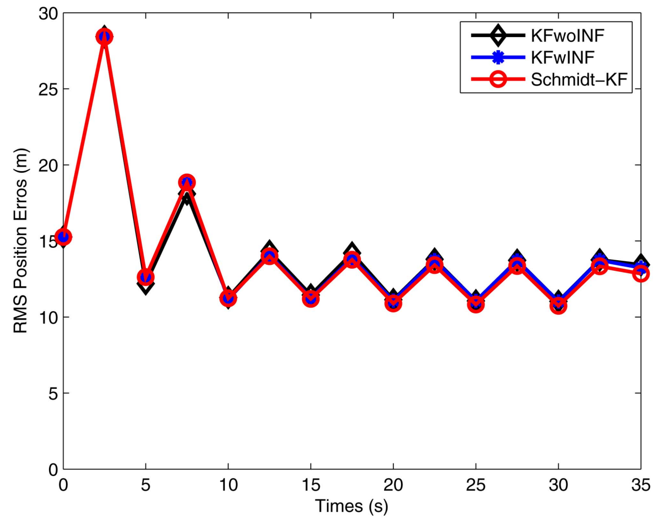


Fig. 3. Position RMSE for target with position measurement only (small bias, OOSM processing).

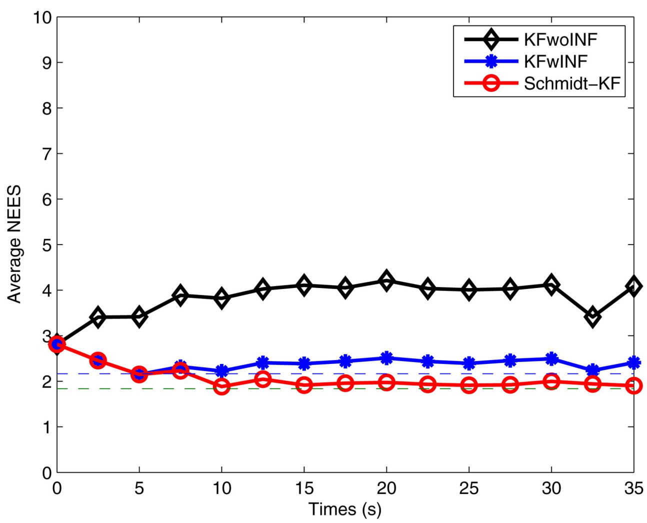


Fig. 2. NEES for target with position measurement only (small bias, reordering measurements).

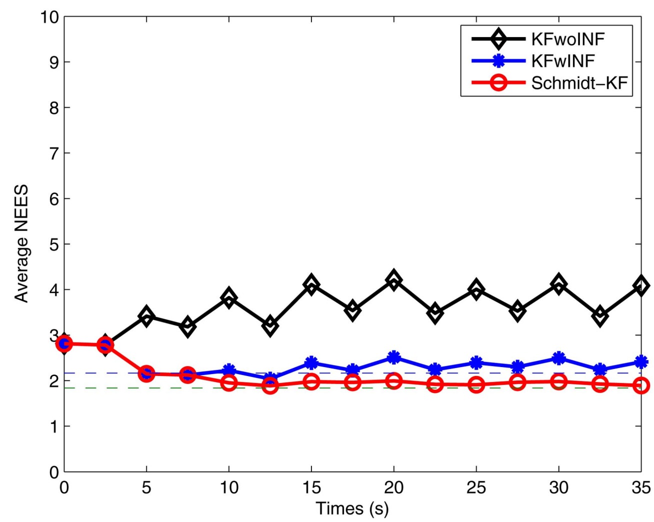


Fig. 4. NEES for target with position measurement only (small bias, OOSM processing).

and 7, is nearly the same as the RMSE from the in-sequence data in Figs. 1 and 5 at the corresponding times. From the NEES in Figs. 2, 4, 6 and 8, we can see that SKF is consistent (i.e., its NEES falls in its probability region [1]) for both in-sequence data and processing OOSMs. KFwoINF is the most inconsistent (overly optimistic) and its inconsistency increases with the bias level. KFwINF improves the filter's consistency but it is still not consistent due to ignoring the correlation between the bias and estimation error (because it assumes the bias to be white noise, as indicated in Section 7). The inconsistency of KFwINF also increases with the bias level. The results also show that KFwINF and SKF do not improve the estimation accuracy in this example with position only measurements. However, in the next example we will see that the SKF does improve the estimation accuracy for the case of GMTI measurements, which include additional range rate measurements.

Tables III and IV show a comparison of the three algorithms for various levels of process noise in the case of small bias for in-sequence data and processing OOSMs, respectively. SKF is consistent in almost all cases.[5] For all process noise PSD levels, KFwoINF is inconsistent (optimistic). For KFwINF, when the PSD is around 10 m^2/s^3 (with the maneuvering index around 1), KFwINF is consistent and for the other cases, KFwINF is inconsistent.

For large bias, the results are shown in Tables V and VI for in-sequence data and processing OOSMs, respectively. The consistency of SKF is the best, though it seems a little "pessimistic" when the process noise PSD is below 0.01 m^2/s^3. For this case of large bias, KFwINF is always inconsistent. Also, from the RMS

[5]Since the multiplicative bias requires linearization, minor inconsistencies (NEES slightly outside the probability region) can occur.

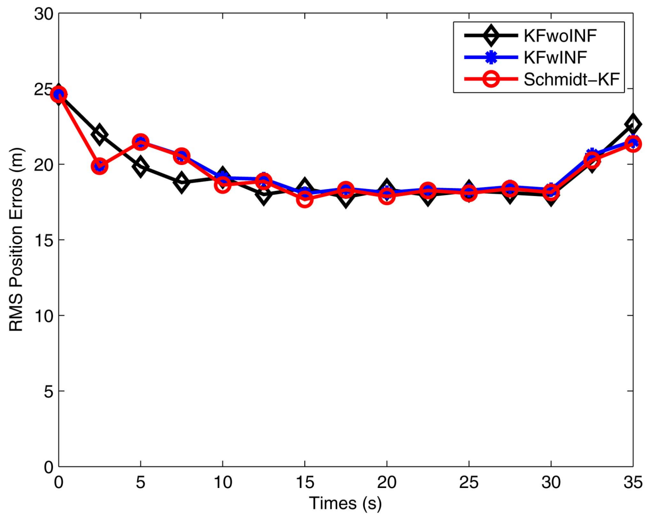


Fig. 5. Position RMSE for target with position measurement only (large bias, reordering measurements).

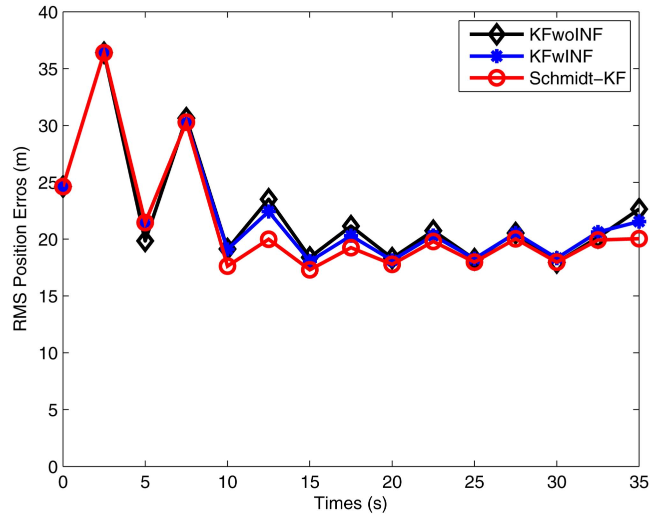


Fig. 7. Position RMSE for target with position measurement only (large bias, OOSM processing).

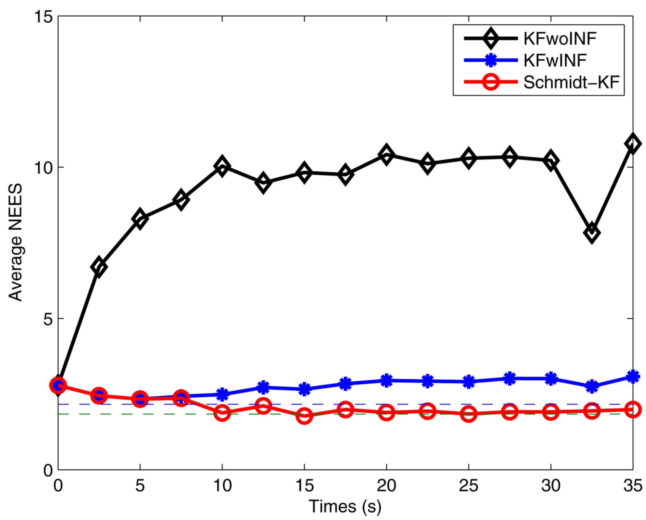


Fig. 6. NEES for target with position measurement only (large bias, reordering measurements).

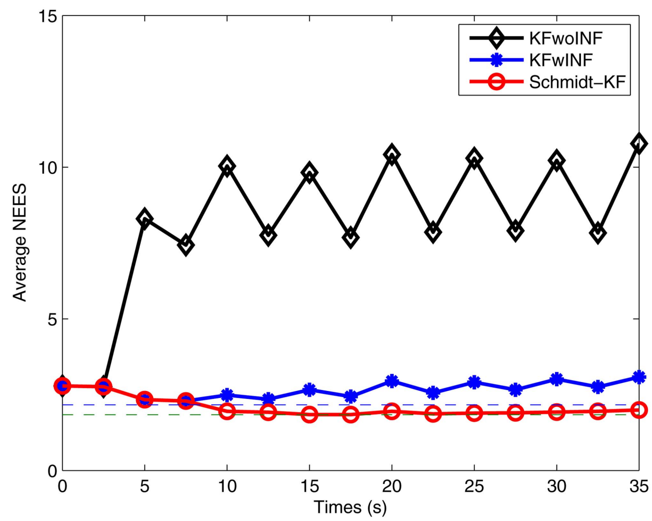


Fig. 8. NEES for target with position measurement only (large bias, OOSM processing).

in Tables III–VI we can notice a small improvement in estimation accuracy using SKF at the final estimate (as in Figs. 1, 3, 5 and 7). However, as shown in Figs. 1, 3, 5 and 7, this improvement is not significant and may not be achieved for every point.

In a realistic scenario, the target is usually tracked in 3D space and a more complicated bias model should be used [15]. However, the results with the use of SKF are similar to the above 1D case and this simple example provides a good illustration of the effect of biases.

8.2. Example 2: Target With Low Process Noise And GMTI Measurements

This example considers a target that moves in a 2-dimensional space with a nearly constant velocity. The target state consists of position and velocity along each coordinate (x and y). The initial target state is $[100\ \text{m}, 9\ \text{m/s}, 200\ \text{m}, 5\ \text{m/s}]$. The PSD of the process noise is $q = 0.5\ \text{m}^2/\text{s}^3$ for both x and y coordinate. The motion model considered is DCWNA. Two GMTI radars are located with nearly perpendicular LOS to the target. One is at $(-48, 13)$ km with a slant range around 50 km to the target and the other is at $(-26, -96)$ km with a slant range around 100 km to the target. The measurements are range (r), azimuth (θ) and range rate ($\dot{r}$) with s.d. $\sigma_r = 10$ m, $\sigma_\theta = 1$ mrad and $\sigma_{\dot{r}} = 1$ m/s, respectively, for both sensors. The measurement model is

$$z^i = (I_3 + \Lambda^i)h(\mathbf{x} - \mathbf{x}_p^i) + \Delta^i + w^i, \qquad i = 1,2 \tag{120}$$

where

$$\mathbf{x} = [x \quad \dot{x} \quad y \quad \dot{y}]' \tag{121}$$

denotes here the (unaugmented) target state and and $\mathbf{x}_p^i$ denotes the ith sensor state, the function $h : \mathcal{R}^4 \to \mathcal{R}^3$ is

TABLE III
Comparison of RMS Errors at Final Estimate for Different Maneuvering Index Based on 1000 Monte Carlo Runs
(Small Bias, Reordering Measurements)

Algorithm	Maneuvering Index λ	Process Noise PSD (m^2/s^3)	Position RMS (m)	Velocity RMS (m/s)	NEES
KFwoINF	0.0125	0.001	10.4110	0.3176	9.6063
KFwINF	0.0125	0.001	10.2422	0.2914	4.8272
Schmidt-Kalman	0.0125	0.001	10.1193	0.2832	1.9872
KFwoINF	0.0395	0.01	10.5154	0.4651	6.2039
KFwINF	0.0395	0.01	10.3116	0.4382	3.4659
Schmidt-Kalman	0.0395	0.01	10.2021	0.4299	1.8121
KFwoINF	0.125	0.1	11.7715	1.0856	4.7297
KFwINF	0.125	0.1	11.4237	1.0669	2.8152
Schmidt-Kalman	0.125	0.1	11.3450	1.0640	1.9334
KFwoINF	0.3953	1	14.0554	2.2643	4.0057
KFwINF	0.3953	1	13.9014	2.2850	2.3912
Schmidt-Kalman	0.3953	1	13.8626	2.2634	1.9900
KFwoINF	1.25	10	14.0609	4.8089	3.3235
KFwINF	1.25	10	14.1587	4.9487	2.0359
Schmidt-Kalman	1.25	10	13.9012	4.8435	1.8791

TABLE IV
Comparison of RMS Errors at Final Estimate for Different Maneuvering Index Based on 1000 Monte Carlo Runs
(Small Bias, OOSM Processing)

Algorithm	Maneuvering Index λ	Process Noise PSD (m^2/s^3)	Position RMS (m)	Velocity RMS (m/s)	NEES
KFwoINF	0.0125	0.001	10.4110	0.3176	9.6063
KFwINF	0.0125	0.001	10.2422	0.2914	4.8271
Schmidt-Kalman	0.0125	0.001	10.0202	0.2829	2.0357
KFwoINF	0.0395	0.01	10.5155	0.4651	6.2039
KFwINF	0.0395	0.01	10.3116	0.4382	3.4658
Schmidt-Kalman	0.0395	0.01	10.0704	0.4254	1.8400
KFwoINF	0.125	0.1	11.7723	1.0856	4.7301
KFwINF	0.125	0.1	11.4239	1.0669	2.8153
Schmidt-Kalman	0.125	0.1	11.0296	1.0564	1.9419
KFwoINF	0.3953	1	14.0557	2.2639	4.0057
KFwINF	0.3953	1	13.9006	2.2848	2.3909
Schmidt-Kalman	0.3953	1	13.5495	2.2835	1.9692
KFwoINF	1.25	10	14.0624	4.8086	3.3233
KFwINF	1.25	10	14.1590	4.9481	2.0351
Schmidt-Kalman	1.25	10	13.8653	4.8579	1.8629

given by

$$h(\mathbf{x}) = \begin{bmatrix} r \\ \theta \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \sqrt{x^2 + y^2} \\ \tan^{-1}\dfrac{y}{x} \\ \dot{x}\cos\theta + \dot{y}\sin\theta \end{bmatrix}. \qquad (122)$$

I_3 denotes the 3×3 identity matrix, Λ represents the scale bias, having the form

$$\Lambda = \begin{bmatrix} \alpha_r & 0 & 0 \\ 0 & \alpha_\theta & 0 \\ 0 & 0 & \alpha_{\dot{r}} \end{bmatrix} \qquad (123)$$

with the bias terms on its main diagonal and Δ denotes the offset bias, that is,

$$\Delta = [\Delta_r \quad \Delta_\theta \quad \Delta_{\dot{r}}]'. \qquad (124)$$

The extended Kalman filter (EKF) is used with the Jacobian terms given in Appendix A. The order of the measurements arriving at the fusion center is shown in Table I. Two bias levels are considered, with the bias s.d. given in Table VII. The results are based on 500 Monte Carlo simulations. The two-sided 99% probability region of the NEES is [3.68, 4.33] based on the χ^2_{2000} distribution.

TABLE V
Comparison of RMS Errors at Final Estimate for Different Maneuvering Index Based on 1000 Monte Carlo Runs
(Large Bias, Reordering Measurements)

Algorithm	Maneuvering Index λ	Process Noise PSD (m^2/s^3)	Position RMS (m)	Velocity RMS (m/s)	NEES
KFwoINF	0.0125	0.001	18.0346	0.4642	29.3834
KFwINF	0.0125	0.001	17.3345	0.3283	5.9046
Schmidt-Kalman	0.0125	0.001	16.8580	0.2902	1.7161
KFwoINF	0.0395	0.01	18.9462	0.6461	18.9844
KFwINF	0.0395	0.01	17.7781	0.4978	4.8489
Schmidt-Kalman	0.0395	0.01	17.2558	0.4684	1.7761
KFwoINF	0.125	0.1	20.6240	1.3104	13.7247
KFwINF	0.125	0.1	19.4403	1.1497	3.7531
Schmidt-Kalman	0.125	0.1	19.1942	1.1357	1.9113
KFwoINF	0.3953	1	23.4515	2.7203	9.7744
KFwINF	0.3953	1	22.6597	2.6533	2.7582
Schmidt-Kalman	0.3953	1	22.5325	2.6332	1.9727
KFwoINF	1.25	10	25.3687	4.9322	8.5757
KFwINF	1.25	10	25.4763	5.4990	2.3465
Schmidt-Kalman	1.25	10	24.7708	5.1617	1.9618

TABLE VI
Comparison of RMS Errors at Final Estimate for Different Maneuvering Index Based on 1000 Monte Carlo Runs
(Large Bias, OOSM Processing)

Algorithm	Maneuvering Index λ	Process Noise PSD (m^2/s^3)	Position RMS (m)	Velocity RMS (m/s)NEES	
KFwoINF	0.0125	0.001	18.0346	0.4642	29.3835
KFwINF	0.0125	0.001	17.3345	0.3283	5.9043
Schmidt-Kalman	0.0125	0.001	16.5600	0.2960	1.7797
KFwoINF	0.0395	0.01	18.9465	0.6461	18.9847
KFwINF	0.0395	0.01	17.7781	0.4978	4.8488
Schmidt-Kalman	0.0395	0.01	16.7486	0.4635	1.8212
KFwoINF	0.125	0.1	20.6255	1.3103	13.7260
KFwINF	0.125	0.1	19.4405	1.1497	3.7531
Schmidt-Kalman	0.125	0.1	18.5341	1.0975	1.9393
KFwoINF	0.3953	1	23.4517	2.7192	9.7738
KFwINF	0.3953	1	22.6590	2.6530	2.7580
Schmidt-Kalman	0.3953	1	21.1264	2.6219	1.9659
KFwoINF	1.25	10	25.3720	4.9349	8.5752
KFwINF	1.25	10	25.4715	5.4991	2.3454
Schmidt-Kalman	1.25	10	24.3123	5.3158	1.9246

TABLE VII
Bias Standard Deviations for GMTI Measurements

Bias Level	Offset Bias Δ_r	Offset Bias Δ_θ	Offset Bias $\Delta_{\dot{r}}$	Scale Bias α_r	Scale Bias α_θ	Scale Bias $\alpha_{\dot{r}}$
Small	10 m ($1 \times \sigma_r$)	1 mrad ($1 \times \sigma_\theta$)	1 m/s ($1 \times \sigma_{\dot{r}}$)	1×10^{-4}	1×10^{-4}	1×10^{-4}
Large	20 m ($2 \times \sigma_r$)	2 mrad ($2 \times \sigma_\theta$)	2 m/s ($2 \times \sigma_{\dot{r}}$)	2×10^{-4}	2×10^{-4}	2×10^{-4}

From the RMSE in Figs. 9 and 11 we can see that SKF improves estimation accuracy compared to KFwoINF and KFwINF for in-sequence data as well as in case of processing OOSM. KFwINF is even worse than KFwoINF in this case. With the bias level increasing, the improvement in RMSE using SKF (shown in Figs. 13 and 15) becomes more significant. For consistency, from the NEES shown in Figs. 10 and 12 we can see that SKF takes some time to become consistent since the initial crosscovariance between the estimation

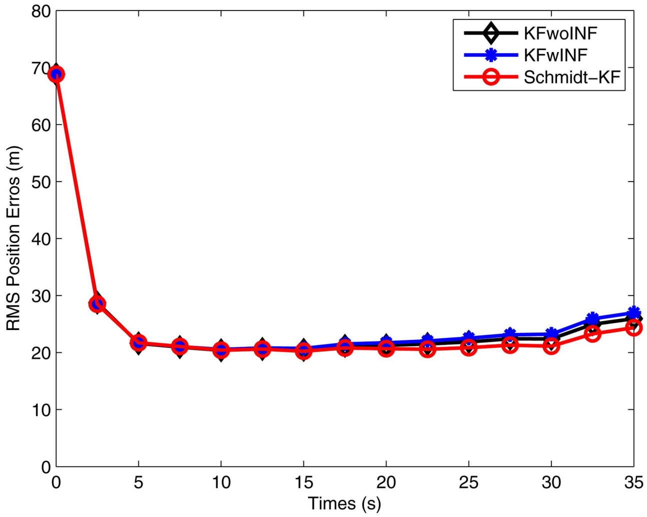


Fig. 9. Position RMSE for target with GMTI measurement (small bias, reordering measurements).

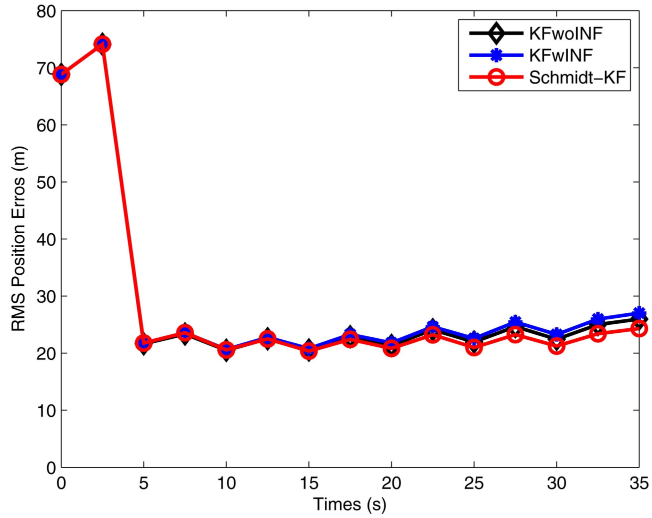


Fig. 11. Position RMSE for target with GMTI measurement (small bias, OOSM processing).

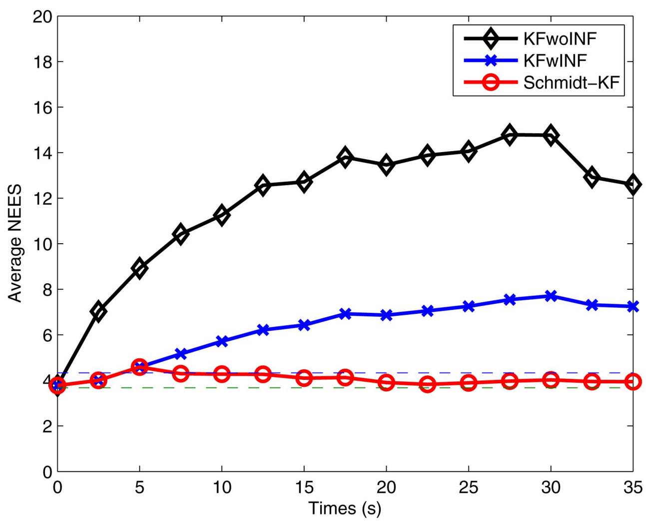


Fig. 10. NEES for target with GMTI measurement (small bias, reordering measurements).

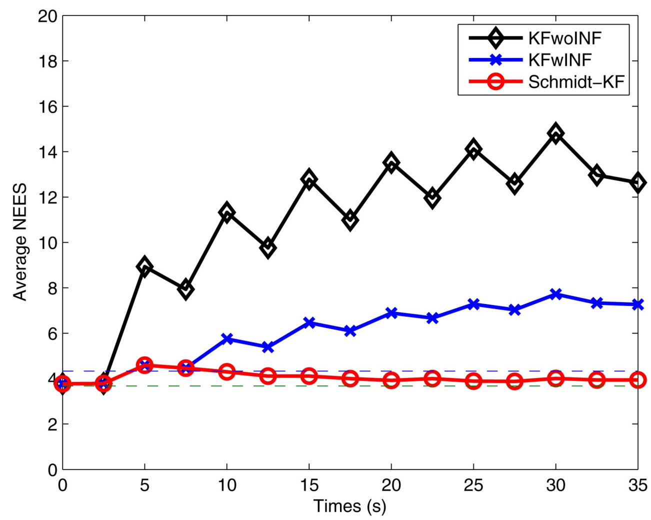


Fig. 12. NEES for target with GMTI measurement (small bias, OOSM processing).

error and the bias is set to be zero[6] and the SKF needs several updates to obtain the correct crosscovariance.

When the bias level increases, SKF needs more updates for the covariance to become consistent (as shown in Figs. 14 and 16). KFwoINF is the most inconsistent (with NEES around 40 in Fig. 14 for the case of large bias). KFwINF improves the consistency but is still not consistent (with NEES around 10 in Fig. 14 for the case of large bias).

8.3. Example 3: Maneuvering Target With GMTI Radar Measurements

This example considers a target with realistic maneuvers. The initial state of the target is [100 m, $10/\sqrt{2}$ m/s, 200 m, $10/\sqrt{2}$ m/s]. The target moves with a constant velocity, 10 m/s during $t \in [0, 15\text{ s}]$. Then, it makes a left turn with a constant speed $V = 10$ m/s and a constant turn rate $w = 5^\circ/\text{s} \approx 0.09$ rad/s during $t \in [15\text{ s}, 35\text{ s}]$. In addition to the maneuver it subjects to process noise with PSD of $q_1 = 0.01\ \text{m}^2/\text{s}^3$ for the entire period. The maneuver corresponds, over the sampling interval $T = 2.5$ s, to a velocity change of (approximately) $\Delta V = wVT = 2.25$ m/s. Equating this to the RMS velocity change due to the process noise over interval T, which is given by $\sqrt{q_2 T}$ [1], yields for this case $q_2 = (wVT)^2/T \approx 2\ \text{m}^2/\text{s}^3$.

Two GMTI radars are used in this scenario as Example 2. An IMM estimator is used to track the maneuvering target with two nearly constant velocity (NCV) [1] models. One has a low process noise PSD $q_1 = 0.01\ \text{m}^2/\text{s}^3$ and the other has a high process noise PSD $q_2 = 2\ \text{m}^2/\text{s}^3$. The mode transition matrix is (106) with

[6]The initial crosscovariance between the estimation error and the bias is not available exactly since one needs the true state x to evaluate this crosscovariance due to the scale bias. Using the initial state estimate in the crosscovariance yields the same minor initial inconsistency as when the initial crosscovariance is set to be zero.

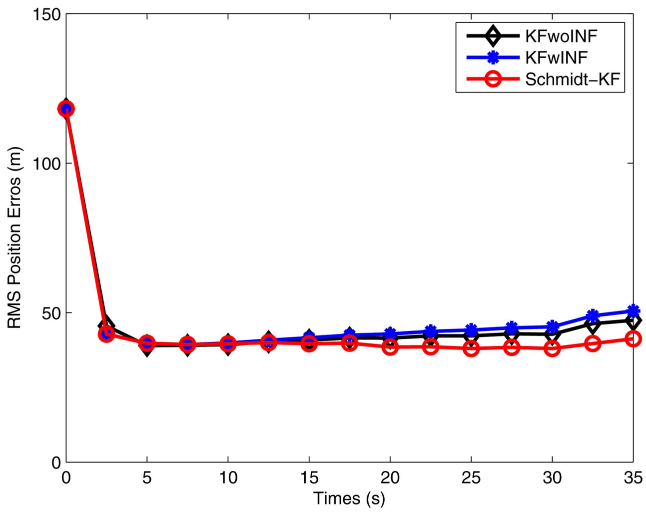


Fig. 13. Position RMSE for target with GMTI measurement (large bias, reordering measurements).

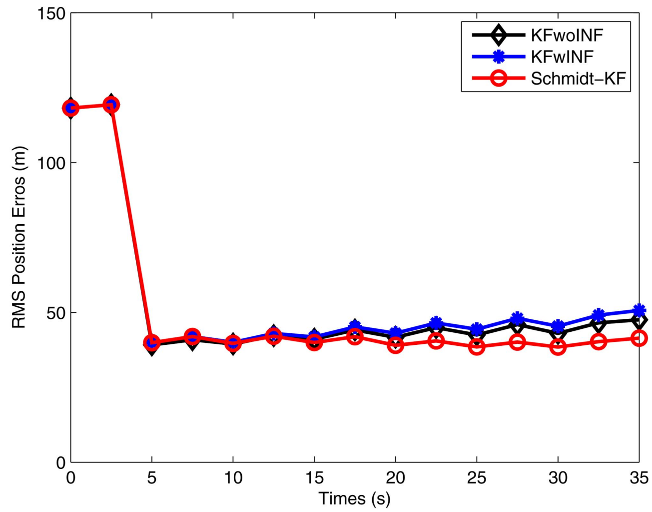


Fig. 15. Position RMSE for target with GMTI measurement (large bias, OOSM processing).

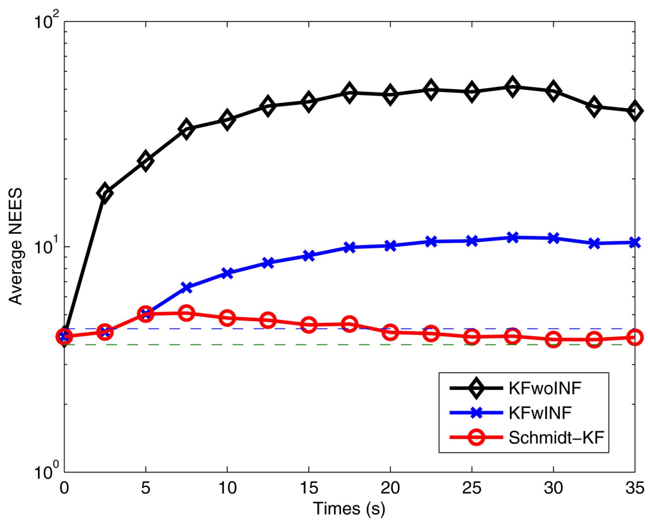


Fig. 14. NEES for target with GMTI measurement (large bias, reordering measurements).

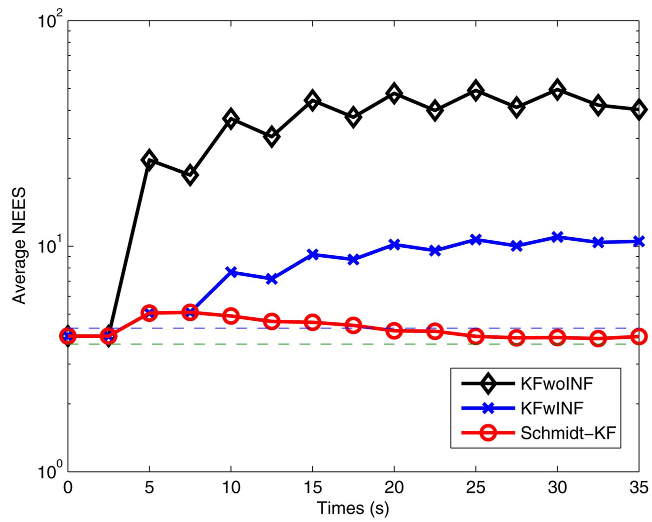


Fig. 16. NEES for target with GMTI measurement (large bias, OOSM processing).

the sojourn times: $\lambda_1^{-1} = 15$ s, $\lambda_2^{-1} = 20$ s. The motion model used is DCWNA. The measurement sequence received at the fusion center is as shown in Table I. The bias s.d. is as given in Table VII. The simulation results below are from 500 Monte Carlo runs.

From the NEES in Figs. 18, 20, 22, and 24 we can see that, due to the use of IMM filter, SKF is not consistent anymore,[7] especially during the mode transition period. However, compared to KFwoINF and KFwINF, the consistency is still improved significantly. At the times when the OOSMs are processed (at the time stamps of sensor 1 in our case), the RMSE for OOSM processing (as shown in Figs. 19 and 23) and the RMSE for in-sequence data (in Figs. 17 and 21) are almost the same. As in Example 2, SKF improves estimation accuracy compared to KFwoINF and KFwINF, even though not significantly in the case of small bias.

[7]No IMM estimation can be perfectly consistent because the inconsistency of a model drives it "soft switching".

9. SUMMARY AND CONCLUSIONS

The single sensor algorithm Bl1, which updates the current state of a target with an OOSM from a single sensor without bias has been extended to the multisensor situation where each sensor exhibits a residual bias. This has been accomplished using the proposed algorithm SKF/OOSM, without having to use an augmented state consisting of the target state and the sensor biases, which can become prohibitive for real-time implementation. This method was presented in the context of a Kalman filter and has also been extended to an IMM estimator. The SKF/OOSM algorithm was compared with the plain Kalman filter without compensation and the (heuristic) Kalman filter with covariance inflation, in the presence of residual biases. The simulation re-

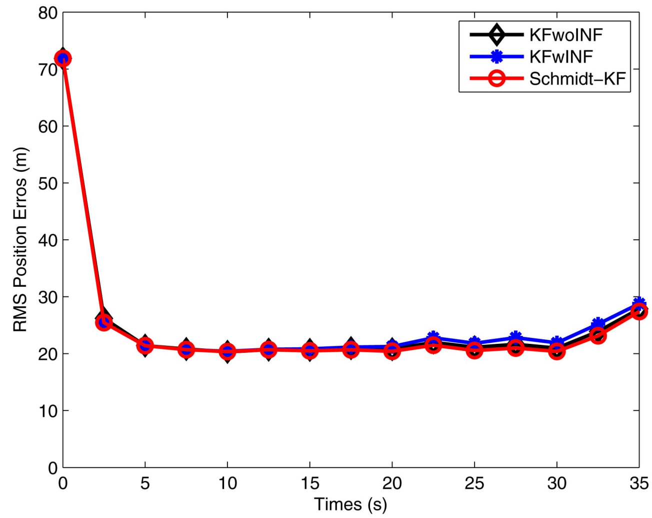


Fig. 17. Position RMSE for maneuvering target with GMTI measurement (small bias, reordering measurements).

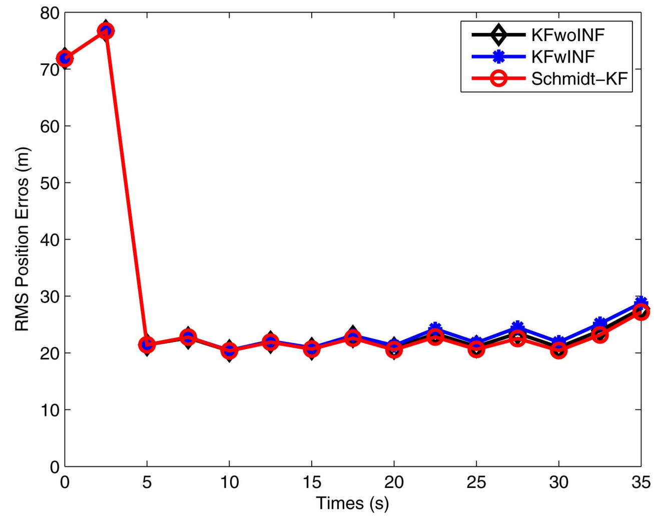


Fig. 19. Position RMSE for maneuvering target with GMTI measurement (small bias, OOSM processing).

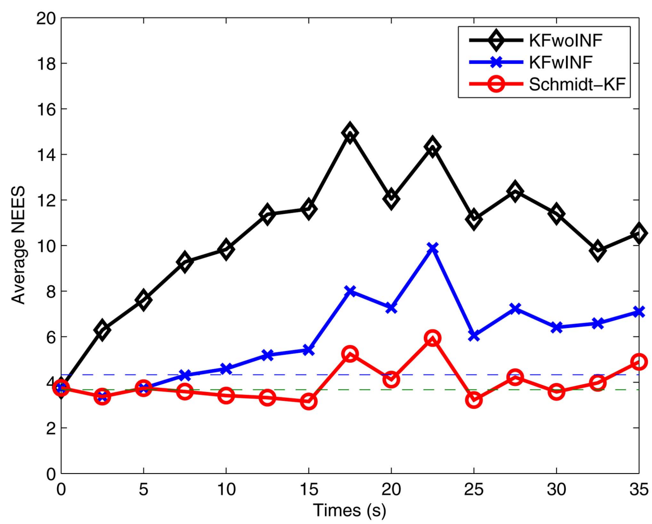


Fig. 18. NEES for maneuvering target with GMTI measurement (small bias, reordering measurements).

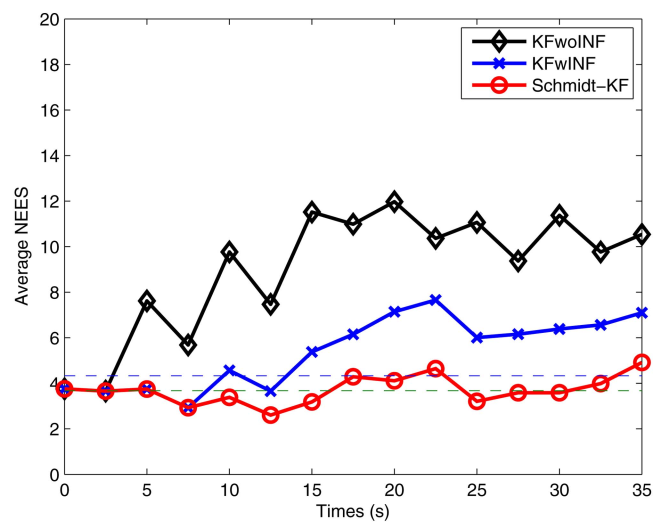


Fig. 20. NEES for maneuvering target with GMTI measurement (small bias, OOSM processing).

sults show that, compared to the other two methods, the major benefit of the SKF/OOSM algorithm is the significant improvement in filter consistency for both in-sequence data and processing OOSMs. For the estimation error, in the case using position only measurements, neither the SKF/OOSM algorithm nor the covariance-inflation method provide improvement in estimation accuracy over the plain Kalman filter without compensation. However, when GMTI measurements are used, which include additional range rate measurements, the SKF/OOSM algorithm outperforms the other two methods in both estimation accuracy and filter consistency.

APPENDIX A. DERIVATIONS OF JACOBIAN FOR GMTI MEASUREMENTS

As in (120), the GMTI measurement model is

$$z = (I_3 + \Lambda)h(\mathbf{x} - \mathbf{x}_p) + \Delta + w \tag{125}$$

where the superscript is dropped for simplicity and h is as in (122)

$$h(\mathbf{x}) = \begin{bmatrix} r \\ \theta \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \sqrt{x^2 + y^2} \\ \tan^{-1}\dfrac{y}{x} \\ \dot{x}\cos\theta + \dot{y}\sin\theta \end{bmatrix}. \tag{126}$$

In the sequel, the bias vector $\mathbf{b}$, which consists of the elements of Δ (the offset biases) as well as the diagonal elements of Λ (the multiplicative biases), is defined as

$$\mathbf{b} = [\mathbf{b}_\Delta' \quad \mathbf{b}_\Lambda']' \tag{127}$$

where

$$\mathbf{b}_\Delta = \Delta, \qquad \mathbf{b}_\Lambda = [\alpha_r \quad \alpha_\theta \quad \alpha_{\dot{r}}]'. \tag{128}$$

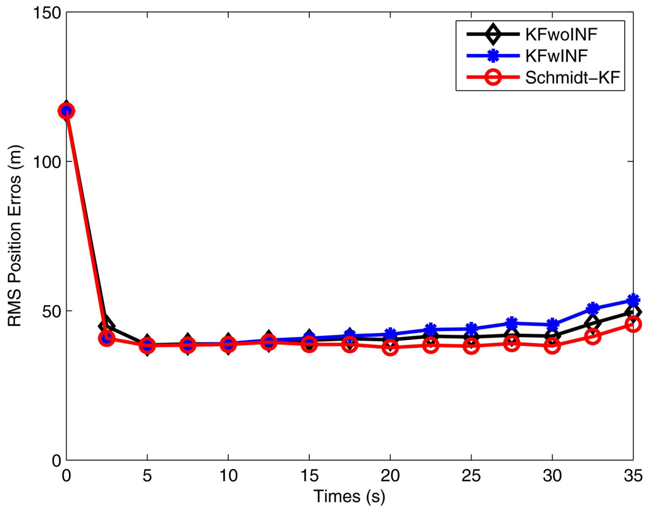


Fig. 21. Position RMSE for maneuvering target with GMTI measurement (large bias, reordering measurements).

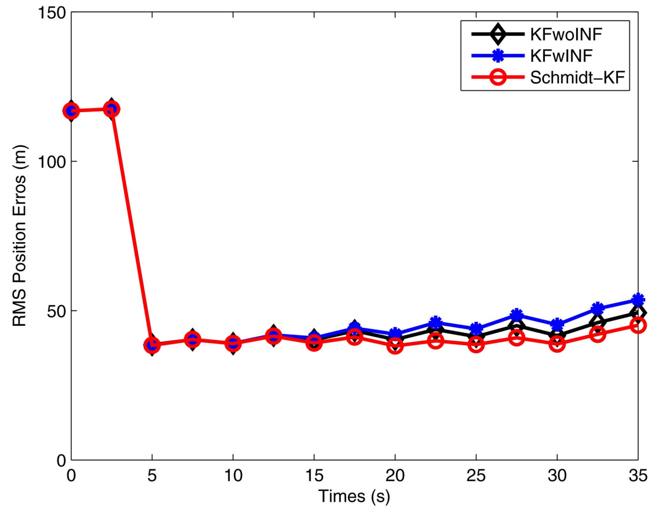


Fig. 23. Position RMSE for maneuvering target with GMTI measurement (large bias, OOSM processing).

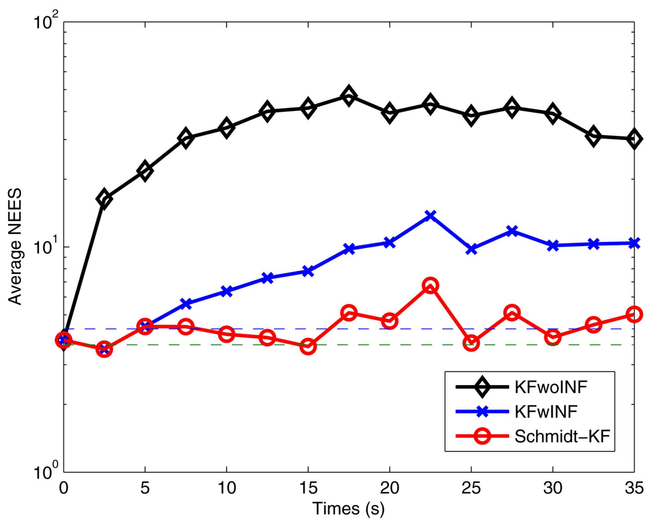


Fig. 22. NEES for maneuvering target with GMTI measurement (large bias, reordering measurements).

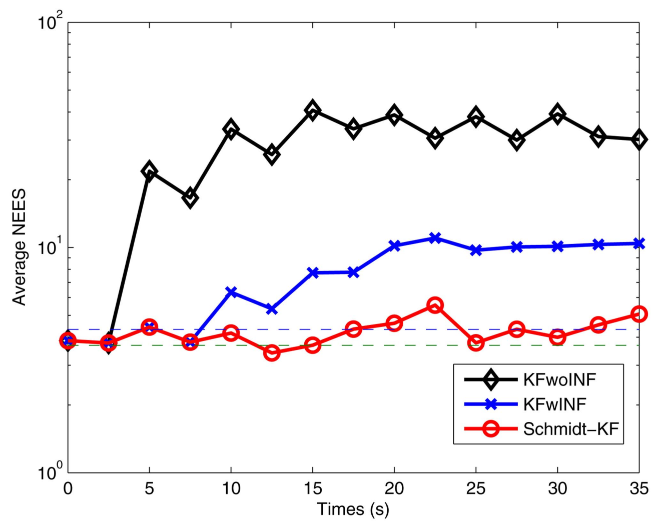


Fig. 24. NEES for maneuvering target with GMTI measurement (large bias, OOSM processing).

The Jacobian with respect to the target state, $H_{\mathbf{x}}$, is

$$\begin{aligned} H_{\mathbf{x}} &= \frac{\partial z}{\partial \mathbf{x}}\bigg|_{\mathbf{x}=\hat{\mathbf{x}},\mathbf{b}=\mathbf{0}} \\ &= (I_3+\Lambda)\frac{\partial h(\mathbf{x}-\mathbf{x}_p)}{\partial(\mathbf{x}-\mathbf{x}_p)}\frac{\partial(\mathbf{x}-\mathbf{x}_p)}{\partial \mathbf{x}}\bigg|_{\mathbf{x}=\hat{\mathbf{x}},\mathbf{b}=\mathbf{0}} \\ &= H(\hat{\mathbf{x}}-\mathbf{x}_p), \end{aligned} \tag{129}$$

where $H(\mathbf{x})$ is

$$\begin{aligned} H(\mathbf{x}) &= \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \\ &= \begin{bmatrix} \frac{x}{r} & 0 & \frac{y}{r} & 0 \\ -\frac{y}{r^2} & 0 & \frac{x}{r^2} & 0 \\ l\sin\theta & \cos\theta & -l\cos\theta & \sin\theta \end{bmatrix} \end{aligned} \tag{130}$$

and

$$l = \frac{\dot{x}\sin\theta - \dot{y}\cos\theta}{r}. \tag{131}$$

The value of $\hat{\mathbf{x}}$ is $\hat{\mathbf{x}}(k \mid k-1)$ for a normal update and is $\hat{\mathbf{x}}(\kappa \mid k)$ for OOSM.

The Jacobian with respect to the bias, $H_{\mathbf{b}}$ can be divided into two parts, that is,

$$H_{\mathbf{b}} = [H_{\mathbf{b}_\Delta} \quad H_{\mathbf{b}_\Lambda}] \tag{132}$$

where $H_{\mathbf{b}_\Delta}$ is the Jacobian with respect to the offset bias $\mathbf{b}_\Delta$ and $H_{\mathbf{b}_\Lambda}$ is the Jacobian with respect to the scale bias $\mathbf{b}_\Lambda$, which are given by

$$\begin{aligned} H_{\mathbf{b}_\Delta} &= \frac{\partial z}{\partial \mathbf{b}_\Delta}\bigg|_{\mathbf{x}=\hat{\mathbf{x}},\mathbf{b}=\mathbf{0}} \\ &= \frac{\partial \Delta}{\partial \mathbf{b}_\Delta}\bigg|_{\mathbf{x}=\hat{\mathbf{x}},\mathbf{b}=\mathbf{0}} \\ &= I_3 \end{aligned} \tag{133}$$

$$H_{\mathbf{b}_\Lambda} = \left.\frac{\partial z}{\partial \mathbf{b}_\Lambda}\right|_{\mathbf{x}=\hat{\mathbf{x}},\mathbf{b}=\mathbf{0}} = \left.\frac{\partial \Lambda h(\mathbf{x}-\mathbf{x}_p)}{\partial \mathbf{b}_\Lambda}\right|_{\mathbf{x}=\hat{\mathbf{x}},\mathbf{b}=\mathbf{0}} = \mathrm{diag}[h(\hat{\mathbf{x}}-\mathbf{x}_p)] \quad (134)$$

where diag[**a**] is a square matrix with the elements of the vector **a** on its main diagonal and the other elements zero.

REFERENCES

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

[2] Y. Bar-Shalom
Update with out-of-sequence measurements in tracking: Exact solution.
IEEE Transactions on Aerospace and Electronic Systems, **38**, 3 (July 2002), 769–778.

[3] Y. Bar-Shalom, H. Chen, and M. Mallick
One-step solution for the multistep out-of-sequence measurement problem in tracking.
IEEE Transactions on Aerospace and Electronic Systems, **AES-40**, 1 (Jan. 2004), 27–37.

[4] Y. Bar-Shalom and H. Chen
IMM estimator with out-of-sequence measurements.
IEEE Transactions on Aerospace and Electronic Systems, **AES-41**, 1 (Jan. 2005), 90–98.

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

[6] R. D. Hilton, D. A. Martin, and W. D. Blair
Tracking with Time-Delayed Data in Multisensor Systems.
NSWCDD/TR-93/351, Aug. 1993, Dahlgren, VA.

[7] A. H. Jazwinski
Stochastic Processes and Filtering Theory.
Academic Press, 1970.

[8] W. Koch
On Accumulatd State Densities with Applications to out-of-sequence measurement processing.
Proceedings of the 12th International Conference on Information Fusion, Seattle, WA, July 2009.

[9] X. D. Lin, Y. Bar-Shalom, and T. Kirubarajan
Exact multisensor dynamic bias estimation with local tracks.
IEEE Transactions on Aerospace and Electronic Systems, **AES-40**, 2 (Apr. 2004), 576–590.

[10] M. Mallick, S. Coraluppi, and Y. Bar-Shalom
Comparison of out-of-sequence measurements algorithms in multi-platform target tracking.
Proceedings of 2001 International Conference on Information Fusion, Montreal, Canada, Aug. 2001.

[11] M. Mallick, S. Coraluppi, and C. Carthel
Advances in asynchronous and decentralized estimation.
Proceedings of 2001 IEEE Aerospace Conference, Big Sky, MT, Mar. 2001.

[12] M. Mallick and A. Marrs
Comparison of the KF and particle filter based out-of-sequence measurement filtering algorithms.
Proceedings of 2003 International Conference on Information Fusion, Cairns, Australia, July 8–11, 2003.

[13] M. Mallick and K. Zhang
Optimal multiple-lag out-of-sequence measurement algorithm based on generalized smoothing framework.
Proceedings of Signal and Data Processing of Small Targets, San Diego, CA, July 31–Aug. 4, 2005.

[14] M. Orton and A. Marrs
Particle filters for tracking with out-of-sequence measurements.
IEEE Transactions on Aerospace and Electronic Systems, **41**, 2 (2005), 693–702.

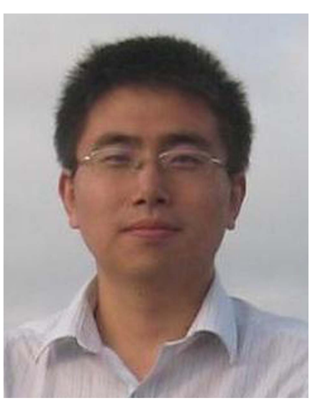
[15] R. Paffenroth, et al.
Mitigation of biases using the Schmidt-Kalman filter.
Proceedings of SPIE Conference on Signal and Data Processing of Small Targets, vol. 6699, San Diego, CA, Aug. 2007.

[16] S. Schmidt
Applications of state space methods to navigation problems.
In C. T. Leondes (Ed.), *Advances in Control Systems*, vol. 3, Academic Press, 1966, 293–340.

[17] X. Shen, Y. Zhu, E. Song, and Y. Luo
Optimal centralized update with multiple local out-of-sequence measurements.
IEEE Transactions on Signal Processing, **57**, 4 (Apr. 2009), 1551–1562.

[18] G. Watson, D. H. McCabe, and T. Rice
Multisensor-multisite composite tracking in the presence of sensor residual bias.
Proceedings of SPIE Conference on Signal and Data Processing of Small Targets, vol. 3809, Denver, CO, July 1999.

[19] K. Zhang, X. R. Li, and Y. Zhu
Optimal update with out-of-sequence measurements for distributed filtering.
IEEE Transactions on Signal Processing, **53**, 6 (June 2005), 1992–2004.



Shuo Zhang was born in 1982. He received the B.E. degree in 2005 and M.E. degree in 2007, both from the Department of Electrical Engineering, Harbin Institute of Technology, China.

He is now a Ph.D. candidate in the Department of Electrical and Computer Engineering, University of Connecticut. His research interests include statistical signal processing, tracking algorithms, and information fusion.

Yaakov Bar-Shalom (S'63—M'66—SM'80—F'84) was born on May 11, 1941. He received the B.S. and M.S. degrees from the Technion, Israel Institute of Technology, in 1963 and 1967 and the Ph.D. degree from Princeton University, Princeton, NJ, in 1970, all in electrical engineering.

From 1970 to 1976 he was with Systems Control, Inc., Palo Alto, CA. Currently he is Board of Trustees Distinguished Professor in the Department of Electrical and Computer Engineering and Marianne E. Klewin Professor in Engineering. He is also director of the ESP Lab (Estimation and Signal Processing) at the University of Connecticut. His research interests are in estimation theory and stochastic adaptive control and he has published over 360 papers and book chapters in these areas. In view of the causality principle between the given name of a person (in this case, "(he) will track," in the modern version of the original language of the Bible) and the profession of this person, his interests have focused on tracking.

He coauthored the monograph *Tracking and Data Association* (Academic Press, 1988), the graduate text *Estimation with Applications to Tracking and Navigation* (Wiley, 2001), the text *Multitarget-Multisensor Tracking: Principles and Techniques* (YBS Publishing, 1995), 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 originated the series of Multitarget Tracking and Multisensor Data Fusion short courses offered at Government Laboratories, private companies, and overseas.

During 1976 and 1977 he served as associate editor of the *IEEE Transactions on Automatic Control* and from 1978 to 1981 as associate editor of *Automatica*. He was program chairman of the 1982 American Control Conference, general chairman of the 1985 ACC, and cochairman of the 1989 IEEE International Conference on Control and Applications. During 1983–1987 he served as chairman of the Conference Activities Board of the IEEE Control Systems Society and during 1987–1989 was a member of the Board of Governors of the IEEE CSS. Currently he is a member of the Board of Directors of the International Society of Information Fusion and served as its Y2K and Y2K2 President. In 1987 he received the IEEE CSS distinguished Member Award. Since 1995 he is a distinguished lecturer of the IEEE AESS. He is corecipient of the M. Barry Carlton Awards for the best paper in the *IEEE Transactions on Aerospace and Electronic Systems* in 1995 and 2000, and received the 1998 University of Connecticut AAUP Excellence Award for Research, the 2002 J. Mignona Data Fusion Award from the DoD JDL Data Fusion Group, and the 2008 IEEE D. J. Picard Medal for Radar Technologies and Applications.

Gregory Watson biography not available.