Performance Prediction of Multisensor Tracking Systems for Single Maneuvering Targets

WILLIAM D. BLAIR
PAUL A. MICELI

Studying the performance of multisensor tracking systems against maneuvering targets involves Monte Carlo simulations with the tracking algorithms implemented in a sophisticated computer simulation of the multisensor system. However, a simplified method for predicting the performance of a multisensor tracking system against maneuvering targets is needed for confirmation of the computer simulations, real-time command and control decisions such as multisensor resource allocation, and systems engineering of complex multisensor systems. The challenge of accurate performance prediction arises from the lack of covariance consistency of the Kalman filter when tracking maneuvering targets. In this paper, a method for performance prediction of a nearly constant velocity Kalman filter is extended to tracking a maneuvering target with multiple dispersed sensors on an oblate earth. Given target position and acceleration as a function of time, the tracking performance of each sensor is expressed as a sensor-noise only (SNO) covariance and maneuver lag or filter bias. In the fusion of the data from the multiple sensors, the SNO covariances fuse for a smaller covariance, while the maneuver lags fuse with a gain proportional to the inverse of the covariances for the sensor tracks. This method can also be used to predict the performance of a multisensor system that include one, two, and/or three dimensional sensors. The results of Monte Carlo simulations of multisensor tracking of a maneuvering target are used to illustrate the accuracy of methods for performance prediction.

Manuscript received May 13, 2008; revised April 13, 2009 and July 18, 2011; released for publication July 26, 2011 .

Refereeing of this contribution was handled by Yaakov Bar-Shalom.

Authors' address: Georgia Tech Research Institute, Georgia Institute of Technology, Atlanta, Georgia 30332-8157, E-mail: (dale.blair@gtri.gatech.edu).

1557-6418/12/$17.00 © 2012 JAIF

1. INTRODUCTION

Studying the performance of multisensor tracking systems against maneuvering targets involves Monte Carlo simulations with the tracking algorithms implemented in a sophisticated computer simulation of the multisensor system [2], [1]. However, a simplified method for predicting the performance of a multisensor tracking system against maneuvering targets is needed to confirm the results of computer simulations, real-time command and control decisions such as multisensor resource allocation, and engineering of complex multisensor systems [3].

When reviewing current approaches to performance prediction, four traits are helpful in distinguishing the relative advantages between the approaches listed in the literature survey and the work presented in this paper. The first two traits deal with the ability to predict performance of multisensor or multitarget tracking algorithms. Most of the current approaches to performance prediction typically deal with a single sensor and single target. In addition, while some algorithms deal with either the multisensor or multitarget case, no work has been identified that is capable of handling both the performance prediction of multisensor and multitarget tracking algorithms. The third trait is ability to predict performance of a defined scenario. While some performance prediction techniques express an expected performance based on sensor parameters, none extend well to maneuvering target scenarios since deterministic changes in motion are not considered. Finally, the fourth trait to be considered is algorithm complexity. Engineering of complex systems often involves extensive parametric variability for which, traits three and four are critical. Algorithm scenario dependence and simplicity yield a high level of confidence when comparing against more complicated simulated algorithms. In this work, we present a simple to compute algorithm that accounts for scenario laydowns and can be extended to multiple sensors while assuming measurement sharing on a single maneuvering target.

Cramer-Rao Bound (CRB) techniques are one of the basic tools for estimating performance. However, these techniques were originally designed for estimating deterministic parameters. More appropriate for the performance prediction of tracking systems, the Posterior Cramer-Rao Bound (PCRB) extends the CRB to provide a "measure" of system performance when both measurements and state are assumed to be stochastic processes [5], [4]. The work of [6] extends the PCRB to handle the prediction of multitarget systems under a set of assumptions. In addition, the work of [7] extends the PCRB to handle the prediction of an estimator for a single maneuvering target. However, none of the papers dedicated to an extension of the PCRB include comparisons to Monte Carlo simulations from a realistic tracker. Thus, the usefulness of each algorithm as it pertains to the performance of a multitarget tracker in a given scenario is not known.

A particular method of performance prediction that compares well to Monte Carlo simulations is the hybrid conditional averaging (HYCA) technique [8]. The unique feature of this algorithm is that it allows for the performance prediction of algorithms where the uncertainties involved are both continuous and discrete. Such is the case for the interacting multiple model (IMM) algorithm, where a HYCA technique specifically tailored for performance prediction of an IMM estimator [9]. While methods based on HYCA compare well to Monte Carlo simulations, the calculations are quite complex. In addition, the existing methods typically only apply to single sensor and single target scenarios.

One extension of the HYCA technique to multisensor multitarget applications is given in [10] where, the Joint Probabilistic Data Association (JPDA) algorithm is of interest. Specifically, two implementations of the algorithm, sequential and parallel, are compared through the use of performance prediction. Though the JPDA algorithm is specifically tailored for multitarget applications, the performance prediction technique in [10] was not extended to multiple targets. Finally, the work in [11] can be extended to multiple sensors but not for multiple targets [12].

Most of the current research in performance prediction with respect to target tracking does not take into account multisensor or multi-target tracking scenarios. PCRB methods do not take into account specific scenarios (they average over an ensemble), and therefore cannot be compared to Monte Carlo simulations. Some techniques such as HYCA compare very well to Monte Carlo simulations, but the calculations are quite complex and have not been extended to multiple targets. In this paper, a multisensor performance prediction technique for a single maneuvering target that is easy to compute and compares very well with Monte Carlo simulations is developed. Therefore, even though the multitarget case is currently not considered, the method presented in this paper has unique advantages in the aggregate when compared to other methods of performance prediction. Future work will include extensions to multisensor and multitarget scenarios.

The challenge of accurate performance prediction arises from the lack of covariance consistency of the Kalman filter when tracking maneuvering targets. For single sensor tracking of maneuvering targets, the lack of covariance consistency is addressed through the use of steady-state filter analysis to decompose the performance characterization into a sensor noise only (SNO) covariance and maneuver lag or filter bias [15], [20], [16]. Thus, given the sensor location, the position and acceleration as a function of time, the parameters for the sensor measurement errors, and the process noise covariance assumed by the filter, the root mean square error (RMSE) in both position and velocity as a function of time can be predicted for a nearly constant velocity Kalman filter. Since the typical variances of the measurement errors in each coordinate are constant or vary slowly, and an alpha-beta filter characterizes Kalman filtering under steady state conditions, the alpha beta filter equations can be used to approximate the tracking performance of each coordinate of the sensor. The SNO covariance and the maneuver lags are generated in each of the sensor coordinates separately as the measurement errors tend to be independent between coordinates [16]. The maneuver lag represents the component of the state estimate error due to a deterministic target maneuver and is assumed constant for a certain time [20]. The SNO covariance represents the stochastic part of the state estimate error due to measurement noise. For each coordinate, typical values of the measurement variance in the corresponding Cartesian coordinate, measurement rate, and the process noise variance of the track filter are used to select the filter gains for an alpha-beta filter, and the SNO covariance and maneuver lag are computed from the gains. Given the SNO covariances and the maneuver lags for all three coordinates, the full track covariance matrix and bias vector can be generated in sensor coordinates, and those can be transformed to any coordinate system.

This method for performance prediction can be extended to nearly constant velocity track filtering with multiple dispersed sensors on an oblate earth. A similar method was utilized in [17] without a rigorous presentation. As in the case of single sensor tracking, the challenge of accurate performance prediction arises from the lack of covariance consistency in the nearly constant velocity Kalman filter. However, multiple sensor frames exist in this case and the technique in [17] must be extended to address this difference. This paper presents a rigorous and detailed development of a method for predicting the performance of nearly constant velocity tracking of a single maneuvering target with multiple sensors in the absence of data association errors. Thus, the predicted performance represents a lower bound for the track errors for nearly constant velocity tracking. The computed SNO covariance and maneuver lags for each sensor used in the equations for the fusion of multiple independent tracks to characterize the fused SNO covariance and maneuver lag. The prediction methods implemented by these equations constitute the significant contribution of this work. In the fusion of the multisensor data, the SNO covariances fuse for a smaller covariance, while the maneuver lags fuse with a gain proportional to the inverse of the covariances for the bias in the sensor tracks. This method can also be used to predict the performance of a multisensor system that include one, two, and/or three dimensional sensors as long as a combination of the sensors provide observability of all three coordinates. The predicted performance represents that of a multisensor tracking system that shares associated measurement reports between sensors and uses nearly constant velocity Kalman filtering for state estimation.

This paper is organized as follows. Background information is given in Section 2 to familiarize the reader with the notation and techniques that serve as the foundation for the methods developed. Least squares estimation, Kalman filtering, alpha-beta filtering, performance prediction of alpha-beta filters, and track fusion are reviewed briefly. In Section 3, Monte Carlo simulation results involving the nearly constant velocity tracking of a maneuvering target are given to illustrate the performance prediction for a single sensor tracking a maneuvering target. In Section 4, the techniques are expanded to the performance prediction of multisensor tracking of a single maneuvering target. The results of Monte Carlo simulations for multisensor tracking of a maneuvering target are shown in Section 5 to verify the methods for performance prediction. Concluding remarks are given in Section 6.

2. BACKGROUND

2.1. Least Squares Estimation

Linear least squares estimation is a mathematical technique for parameter estimation that attempts to find the best linear fit to a set of data, where the best is the parameter value that minimizes the sum of the errors squared. The parameters are assumed to be unknown and time-invariant. Consider the linear observation modeled as

$$Y_j = H_j X + W_j \tag{1}$$

where Y_j is the jth observation j, H_j is known for observation j, X is fixed and unknown, and $W_j \sim (0, R_j)$ is the error in the jth observation. For the expression in (1), an estimate of X may be obtained by minimizing the cost function for a least squares estimator (LSE) that is defined for independent observations as

$$C(X) = (Y - HX)^T R^{-1} (Y - HX)$$

where

$$H = [H_1 \quad H_2 \cdots H_N]^T \tag{2}$$

$$Y = [Y_1 \quad Y_2 \cdots Y_N]^T \tag{3}$$

$$W = [W_1 \quad W_2 \cdots W_N]^T \tag{4}$$

$$R = E[WW^T] = \begin{bmatrix} R_1 & 0 & \cdots & 0 \\ 0 & R_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & R_N \end{bmatrix}. \tag{5}$$

The cost is minimized by taking the partial derivative with respect X and setting the result equal to zero. Rearranging terms to solve for the estimate of X gives

$$\hat{X} = (H^T R^{-1} H)^{-1} H^T R^{-1} Y \tag{6}$$

and the covariance of $\hat{X}$ is given by [21]

$$\mathrm{COV}(\hat{X}) = (H^T R^{-1} H)^{-1}. \tag{7}$$

For Gaussian errors W_j, $\hat{X}$ is the maximum likelihood and minimum mean squared error (MMSE) estimate of X.

2.2. Kalman Filter

A Kalman filter is often employed to filter the kinematic measurements for estimating the position, velocity, and acceleration of a target [1]. The kinematic model commonly assumed for a target in track is given by

$$X_{k+1} = F_k X_k + G_k \nu_k \tag{8}$$

where $\nu_k \sim N(0, Q_k)$ is the process noise that models the unknown target acceleration and F_k defines the linear dynamics. The target state vector X_k contains the position, velocity, and possibly acceleration of the target at time t_k, as well as other variables used to model the time-varying acceleration. For this paper, X_k will include position and velocity. The linear measurement model is given by

$$Y_k = H_k X_k + w_k \tag{9}$$

where Y_k is typically the measurement of the position of the target and $w_k \sim N(0, R_k)$ is the observation error. Both w_k and ν_k are assumed to be independent "white" noise processes. When designing the Kalman filter, Q_k is selected such that the 65% to 95% confidence region about zero contains the maximum acceleration level of the target. However, when targets maneuver, the acceleration changes in a deterministic manner. Thus, the white noise assumption associated with ν_k is often violated and the filter develops a bias in the state estimates. If a larger Q_k is chosen, the bias in the state estimates is less during a maneuver, but then Q_k characterizes poorly the target motion when the target is not maneuvering and the filter performance is far from optimal. Furthermore, the error in modeling the two modes of motion (i.e., nonmaneuvering and maneuvering) with a single model and the error in the white noise assumption for the process noise during maneuvers result in an inaccurate state error covariance that cannot be used reliably for performance prediction. While an Interacting Multiple Model (IMM) estimator [1] can be used to address this conflict, the primary focus of this work is on the quick prediction of average tracking performance and that performance serves as an optimistic guide for track filtering performance of targets maneuvering in the presence of data association errors. Given the mode switching of an IMM estimator is not considered in this work, the performance prediction algorithms presented here can be viewed as an estimate of the upper bound of the expected errors for an IMM estimator.

2.3. Alpha-Beta Filter

The alpha-beta filter is based on the assumption that the target is moving with constant velocity plus zero-mean, white Gaussian acceleration errors. In order to simplify the example and permit analytical predictions of the filter performance, the motion of the target is defined in a single coordinate and the measurements are the positions of the target (i.e., a linear function of the state). For the alpha-beta filter and nearly constant velocity Kalman filter, the state and measurement equations of (8) and (9) are defined by

$$X_k = [x_k \quad \dot{x}_k]^T \tag{10}$$

$$F_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}^T \tag{11}$$

$$G_k = \begin{bmatrix} \frac{1}{2\sqrt{3}} T^{3/2} & \frac{1}{2} T^{3/2} \\ 0 & T^{1/2} \end{bmatrix} \tag{12}$$

$$H_k = [1 \quad 0] \tag{13}$$

where $R_k = \sigma_w^2$ is the variance of the measurement errors in m^2, T is the time interval between measurements, $Q_k = \sigma_\nu^2 I_{2\times 2}$ is the process noise covariance matrix with $\sigma_\nu^2 T^{-1}$ denoting the variance of the "acceleration" errors in $\mathrm{m}^2/\mathrm{s}^4$, and σ_ν^2 denotes the power spectral density (PSD). This process noise model corresponds to the continuous white noise acceleration (CWNA) model [1].

The steady-state form of the constant velocity filter is used for analytical predictions of filter performance. For a filter to achieve these steady-state conditions, the error processes ν_k and w_k must be stationary and the data rate must be constant. While these conditions are seldom satisfied in practice, the steady-state form of the filter can be used to predict average or expected tracking performance. The alpha-beta filter is equivalent to the Kalman filter in steady-state for this motion model. For the alpha-beta filter, the steady-state gains that occur after the transients associated with filter initialization diminish are given by

$$K_k = \begin{bmatrix} \alpha & \frac{\beta}{T} \end{bmatrix}^T \tag{14}$$

where α and β are the optimal gains for the CWNA model. As given in [1], the gains are computed by

$$\Gamma^2 = \frac{\sigma_\nu^2 T^3}{\sigma_w^2} \tag{15}$$

$$\mu = \frac{1}{3} + \sqrt{\frac{1}{12} + \frac{4}{\Gamma^2}} \tag{16}$$

$$\alpha = \beta\sqrt{\mu} \tag{17}$$

$$\beta = \frac{12}{6\left(\mu + \sqrt{\mu}\right) + 1} \tag{18}$$

where Γ is known as the tracking maneuver index. The steady-state error covariance of the alpha-beta filter [1], [13] is given by

$$P_{k|k}^{\alpha\beta} = \sigma_w^2 \begin{bmatrix} \alpha & \frac{\beta}{T} \\ \frac{\beta}{T} & \frac{\beta(2\alpha - \beta)}{2(1-\alpha)T^2} \end{bmatrix}. \tag{19}$$

A simple gain scheduling procedure for approximating the performance of a Kalman filter during initialization from [13] is given by ($k = 0$ for the first measurement)

$$\alpha_k = \max\left\{ \frac{2(2k+1)}{(k+1)(k+2)}, \alpha \right\} \tag{20}$$

$$\beta_k = \max\left\{ \frac{6}{(k+1)(k+2)}, \beta \right\} \tag{21}$$

where α and β are the steady-state values and the initial conditions are given by

$$X_{0|-1} = \begin{bmatrix} x_{0|-1} \\ \dot{x}_{0|-1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{22}$$

2.4. Performance Prediction for Single Sensor Tracking of a Maneuvering Target

The covariance of the state estimate $X_{k|k}$ is given by

$$P_{k|k} = E[(X_{k|k} - \bar{X}_{k|k})(X_{k|k} - \bar{X}_{k|k})^T \mid X_k] \tag{23}$$

where $E[\cdot]$ denotes the expected value operator, and $\bar{X}_{k|k} = E[X_{k|k}]$. When the estimator is unbiased and $E[X_{k|k}] = X_k$, the true value, the covariance is a good predictor of performance. However, when the estimator is biased, the covariance is a poor predictor of performance. When a target undergoes a deterministic maneuver (i.e., a constant acceleration), the estimates are biased and the covariance matrix tends to be an optimistic estimate of track filter performance since it does not reflect the bias. When a target undergoes no maneuver (i.e., a zero acceleration), the covariance matrix tends to also be a biased estimate of track filter performance, because process noise is included in the filter for maneuver response. Thus, in order to address both conditions of the performance prediction, the mean-squared error will be written in terms of a SNO covariance for no maneuver and a bias or maneuver lag for the constant acceleration maneuver.

Let

$$B_{k|k} = E[X_{k|k} \mid X_k] - X_k = \bar{X}_{k|k} - X_k \tag{24}$$

where $B_{k|k}$ denotes the filter bias. Thus, mean squared error (MSE) is given by

$$\begin{aligned} \mathrm{MSE} &= E[(X_{k|k} - X_k)(X_{k|k} - X_k)^T \mid X_k] \\ &= E[(X_{k|k} - \bar{X}_{k|k})(X_{k|k} - \bar{X}_{k|k})^T \\ &\quad + 2(X_{k|k} - \bar{X}_{k|k})(\bar{X}_{k|k} - X_k)^T \\ &\quad + (\bar{X}_{k|k} - X_k)(\bar{X}_{k|k} - X_k)^T \mid X_k] \\ &= P_{k|k} + B_{k|k}B_{k|k}^T. \end{aligned} \tag{25}$$

Consider the case of deterministic maneuvers of either zero acceleration or constant acceleration, the filter covariance is given by the SNO covariance when the acceleration is zero and the SNO covariance plus the bias error squared when the acceleration is a nonzero constant. Letting $S^{\alpha\beta}_{k|k}$ denote the SNO covariance of the alpha-beta filter and $B^{\alpha\beta}_{k|k}$ denote the bias due to an acceleration gives

$$E[(X_{k|k} - X_k)(X_{k|k} - X_k)^T \mid X_k] = S^{\alpha\beta}_{k|k} + B^{\alpha\beta}_{k|k}(B^{\alpha\beta}_{k|k})^T. \quad (26)$$

The SNO covariance and the bias are computed by representing the alpha-beta filter as a linear, time-invariant system with an input that can be expressed as a deterministic signal (i.e., a constant acceleration rather than zero-mean white process noise) with white noise measurement errors. The input-output relationships between the measurements Y_k and state estimate $X_{k|k}$ can be expressed as a linear system that is given by

$$X_{k|k} = \bar{F}_k X_{k-1|k-1} + \bar{G}_k Y_k$$

where

$$\bar{F}_k = \begin{bmatrix} 1-\alpha & (1-\alpha)T \\ -\dfrac{\beta}{T} & 1-\beta \end{bmatrix} \quad (27)$$

$$\bar{G}_k = \begin{bmatrix} \alpha & \dfrac{\beta}{T} \end{bmatrix}^T. \quad (28)$$

The error covariance of $X_{k|k}$ that results from the SNO (i.e., no target acceleration to produce bias) is given in [13], [14], and [18] for arbitrary α and β to be

$$S^{\alpha\beta}_{k|k} = \frac{\sigma_w^2}{\alpha(4-2\alpha-\beta)} \begin{bmatrix} 2\alpha^2 + \beta(2-3\alpha) & \dfrac{\beta}{T}(2\alpha-\beta) \\ \dfrac{\beta}{T}(2\alpha-\beta) & \dfrac{2\beta^2}{T^2} \end{bmatrix} \quad (29)$$

where T is the time period between consecutive measurements and σ_w^2 is the variance of measurement errors. Since (29) includes only the sensor measurement errors, it is referred to as the SNO covariance matrix. For a maneuvering target, the bias or lag in the state estimate for arbitrary α and β is given by

$$B^{\alpha\beta}_{k|k} = \begin{bmatrix} (1-\alpha)\dfrac{T^2}{\beta} \\ \left(\dfrac{\alpha}{\beta} - 0.5\right)T \end{bmatrix} A_k \quad (30)$$

where A_k is the acceleration of the target in the coordinate of interest at time t_k. This is the steady-state bias that results after the transient response of the filter has decayed (i.e., typically three or four measurements) [20].

For an m-step (i.e., measurement intervals) ahead prediction, the error covariance of the state estimate that results from the measurement errors only is given in [17] for an arbitrary α and β to be

$$S^{\alpha\beta}_{k+m|k} = F(m) S^{\alpha\beta}_{k|k} F(m)^T \quad (31)$$

where

$$F(m) = \begin{bmatrix} 1 & mT \\ 0 & 1 \end{bmatrix}. \quad (32)$$

Thus, for an m-step ahead prediction, the SNO covariance is given by

$$S^{\alpha\beta}_{k+m|k} = \frac{\sigma_w^2}{\alpha(4-2\alpha-\beta)} \begin{bmatrix} E_{11} & E_{12} \\ E_{12} & E_{22} \end{bmatrix} \quad (33)$$

where

$$E_{11} = 2\alpha^2 + \beta(2-3\alpha) + 2m\beta(2\alpha-\beta) + 2m^2\beta^2$$

$$E_{12} = \frac{\beta}{T}(2\alpha - \beta + 2m\beta)$$

$$E_{22} = \frac{2\beta^2}{T^2}.$$

For an m-step ahead prediction and a maneuvering target, the bias or lag in the state estimate for arbitrary α and β is given by [17] and [18] as

$$B^{\alpha\beta}_{k+m|k} = \begin{bmatrix} (1-\alpha+(\alpha-0.5\beta)m + 0.5\beta m^2)\dfrac{T^2}{\beta} \\ (\alpha + (m-0.5)\beta)\dfrac{T}{\beta} \end{bmatrix} A_k. \quad (34)$$

The RMSE in the position estimates of the alpha-beta filter for an m-step ahead prediction is given by

$$\mathrm{RMSE}^p(m) = \left[\frac{\sigma_w^2}{\alpha(4-2\alpha-\beta)}(2\alpha^2 + \beta(2-3\alpha) + 2m\beta(2\alpha-\beta) + 2m^2\beta^2) + (1-\alpha+(\alpha-0.5\beta)m + 0.5\beta m^2)^2 \frac{T^4}{\beta^2} A_k^2\right]^{1/2}. \quad (35)$$

The RMSE in the velocity estimates can be expressed as

$$\mathrm{RMSE}^v(m) = \left[\frac{2\sigma_w^2\beta^2}{\alpha(4-2\alpha-\beta)T^2} + ((\alpha + (m-0.5)\beta)\frac{T}{\beta})^2 A_k^2\right]^{1/2}. \quad (36)$$

Note that while m in (31) through (36) is treated as an integer, the results are valid for a fractional measurement interval as well.

2.5. Track Fusion for N Independent Tracks

The fusion of N uncorrelated or independent tracks can be formulated as a linear least-squares estimation problem, where the sensor tracks are treated as observations with independent errors. For N independent tracks with mean and covariance $\{X^i_{k|k}, P^i_{k|k}\}^N_{i=1}$,

$$H = [I_N \quad I_N \cdots I_N]^T \quad (37)$$

$$Y = [X^1_{k|k} \quad X^2_{k|k} \cdots X^N_{k|k}]^T \quad (38)$$

$$R = \begin{bmatrix} P^1_{k|k} & 0 & \cdots & 0 \\ 0 & P^2_{k|k} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & P^N_{k|k} \end{bmatrix}. \tag{39}$$

The block diagonal form allows for the inversion of each block individually. Thus, the fused track and covariance is given by

$$X_{k|k} = \left[\sum_{i=1}^{N}(P^i_{k|k})^{-1}\right]^{-1}\left[\sum_{i=1}^{N}(P^i_{k|k})^{-1}X^i_{k|k}\right] \tag{40}$$

$$P_{k|k} = \left[\sum_{i=1}^{N}(P^i_{k|k})^{-1}\right]^{-1}. \tag{41}$$

2.6. Track Fusion for Two Correlated Tracks

For two correlated tracks, the errors in the estimator are correlated and the covariance is represented as

$$R_k = E[W_k W_k^T] = \begin{bmatrix} P^1_{k|k} & P^{12}_{k|k} \\ (P^{12}_{k|k})^T & P^2_{k|k} \end{bmatrix}. \tag{42}$$

The inverse of this matrix is given by

$$R_k^{-1} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{43}$$

where

$$A_{11} = [P^1_{k|k} - P^{12}_{k|k}(P^2_{k|k})^{-1}(P^{12}_{k|k})^T]^{-1}$$

$$A_{12} = -(P^1_{k|k})^{-1}P^{12}_{k|k} \times [P^2_{k|k} - (P^{12}_{k|k})^T(P^1_{k|k})^{-1}P^{12}_{k|k}]^{-1}$$

$$A_{21} = -(P^2_{k|k})^{-1}(P^{12}_{k|k})^T \times [P^1_{k|k} - P^{12}_{k|k}(P^2_{k|k})^{-1}(P^{12}_{k|k})^T]^{-1}$$

$$A_{22} = [P^2_{k|k} - (P^{12}_{k|k})^T(P^1_{k|k})^{-1}P^{12}_{k|k}]^{-1}.$$

Thus, the fused track is given by

$$X_{k|k} = [D_1 + D_2]^{-1}\{D_1 X^1_{k|k} + D_2 X^2_{k|k}\} \tag{44}$$

where

$$D_1 = [I - (P^2_{k|k})^{-1}(P^{12}_{k|k})^T] \times [P^1_{k|k} - P^{12}_{k|k}(P^2_{k|k})^{-1}(P^{12}_{k|k})^T]^{-1}$$

$$D_2 = [I - (P^1_{k|k})^{-1}P^{12}_{k|k}] \times [P^2_{k|k} - (P^{12}_{k|k})^T(P^1_{k|k})^{-1}P^{12}_{k|k}]^{-1}.$$

Thus, the fusion of correlated tracks can be accomplished. However, analytical expressions are not easily achieved for more than two tracks. Furthermore, calculation of the track correlation further complicates the performance prediction process. For the case of measurement level fusion, the performance prediction is more appropriately matched by ignoring the correlation.

3. PREDICTING PERFORMANCE FOR A SINGLE SENSOR

As a numerical example, a radar tracking system with a 1 Hz measurement rate is considered. The radar measurements are corrupted with zero-mean errors that are Gaussian distributed and have standard deviations of 3 m, 1.1 mrad, and 1.1 mrad in range, azimuthal angle, and vertical angle, respectively. An extended Kalman filter is used for tracking and the filter gains are computed with the standard equations. The process noise power spectral density (PSD) of the tracking system is $q = 100\ \mathrm{m}^2/\mathrm{s}^3$ in each coordinate. This process noise PSD is selected based on the maximum expected acceleration of the target [20], which is 40 m/s^2. Selecting a larger process noise PSD results in a smaller bias during a maneuver and a state error covariance that is large (and wrong) when the target is not maneuvering. Selecting a smaller process noise PSD results in a larger bias during a maneuver and a state error covariance that is too small when the target is maneuvering.

Given the trajectory (i.e., position, velocity, and acceleration) for sensor and target and prediction codes configured with the same characterization of the tracking system described above, the SNO covariance and bias can be computed. The sensor errors in the cross range coordinate are dependent on range as reflected in the computation of the tracking index. For range, the tracking index is

$$\Gamma^r = \sqrt{\frac{\sigma_v^2 T^3}{\sigma_r^2}} \tag{45}$$

where $\sigma_r = 3$ m. For cross range, the tracking index is given by

$$\Gamma_k^{cr} = \sqrt{\frac{\sigma_v^2 T^3}{r_k^2 \sigma_\theta^2}} \tag{46}$$

where r_k is the range from sensor to target at time t_k and $\sigma_\theta = 1.1$ mrad. Using these tracking indices, measurement variances, track rate, and number of measurements, the gains and error covariance can be approximated for each coordinate in the sensor frame. In other words, the SNO covariance and bias as well as the track filter covariance are generated in the orthogonal coordinates that are aligned with the sensor frame. Given r_k, the target acceleration at t_k, the track rate, the number of measurements used in the track, and independence of the errors in each of the sensor coordinates, the predicted RMSE as a function of time can be computed for each of the sensor coordinates using the results from (35) and (36) for position and velocity, respectively. The predicted RMSE that includes all coordinates is computed by adding the mean squared error of all coordinates before taking the square root.

The single sensor, single target scenario used in this analytic computation of RMSE is shown in Fig. 1, where only Sensor 1 is active. In this scenario, a target maneuvers with a near constant acceleration turn of 20 m/s^2 between the intervals $242 \le t_k \le 261$ s (Maneuver 1) and $461 \le t_k \le 479$ s (Maneuver 2), and a near constant acceleration turn of 40 m/s^2 between the interval $961 \le t_k \le 977$ s (Maneuver 3). The predicted performance results for Sensor 1 are shown in Figs. 2 and 3 and compared with the average results from Monte Carlo simulation with 50 runs. Note that predicted results match rather closely to those averages from the

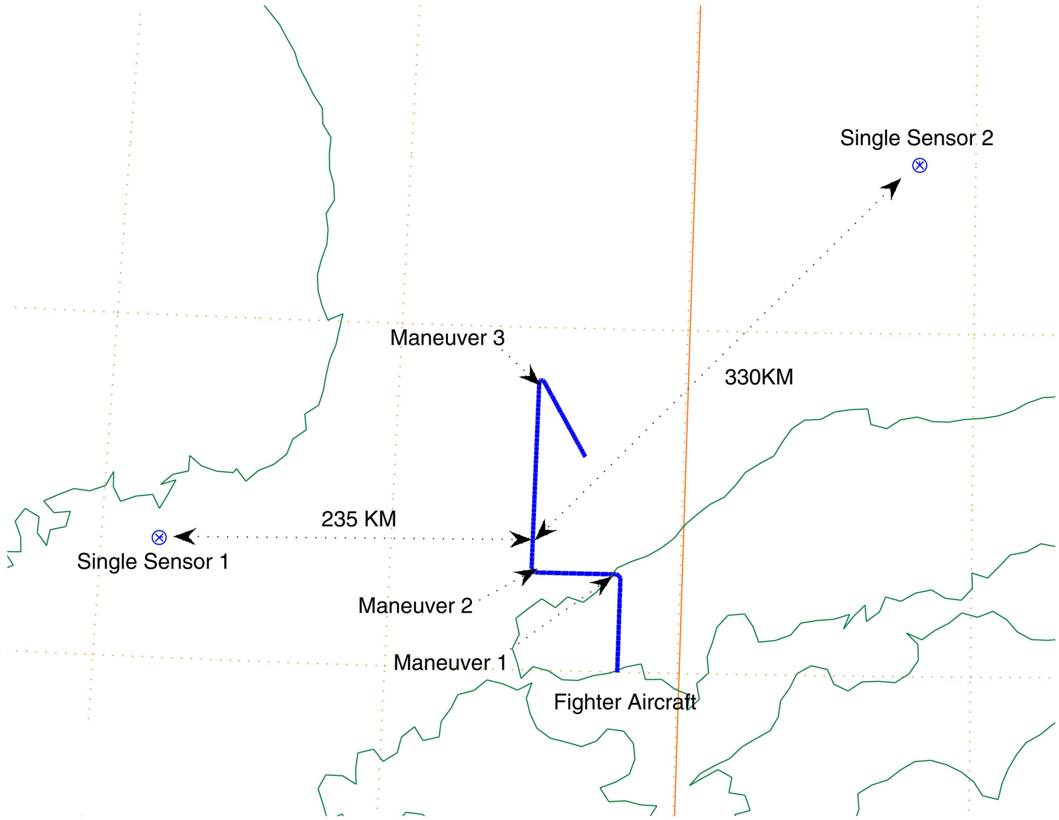

Fig. 1. Multiple sensor and single target scenario.

Monte Carlo simulations. The track filter covariance was also used to predict the RMSE by taking the square root of the sum of the diagonal elements of the covariance matrix for position and velocity.

Comparing Figs. 4–5 highlights an additional improvement for modeling the filter settling time. Since the prediction method is a function of the true target acceleration, the response to maneuvers is assumed instantaneous while in the simulations the response is not instantaneous, and thus one expects the predicted "jump" in RMSE during a maneuver to lead that of the simulated results. Further, one expects the difference between predicted and simulated results to be noticeable when the acceleration in a given coordinate quickly spikes up. Figure 4 illustrates the divergence for maneuver 2 where the target acceleration in cross range quickly spikes and then gradually tapers off. To model the filter settling time, a moving average can be run on the RMSE of the predicted results as shown in Fig. 5. The moving average implemented gives the current predicted RMSE as an average of the current update and the four previous updates. The number of updates chosen for the moving average was selected based on the expected filter settling time given the filter parameters and update rate [20]. Note that all figures illustrating predicted performance, with the exception of Fig. 4, were generated using a moving average as part of the prediction codes.

Figures 2–5 illustrate the inability of the track covariance to accurately predict filter performance. During steady-state tracking with no target maneuver, the track covariance indicates that the position errors are 10% larger than simulation results. In addition, the track covariance indicates that the position errors are significantly smaller than simulation results when the target does maneuver. The same result holds true for velocity as well. The performance prediction of errors in velocity by the track filter covariance has significant error when the target is not maneuvering and this is due to the closer coupling of the random acceleration error modeled to velocity than position. Similarly, the predicted performance versus simulation results for Sensor 2 are shown in Figs. 6 and 7. In this case, the predicted errors also match quite well. However, as illustrated in Fig. 8, we have not modeled the filter setting time as accurately as we did with Sensor 1. In the sensor coordinate frame for Sensor 2, the target acceleration spikes negative before quickly tapering positive. In this case, the negative cross range acceleration did not last long enough for the filter in the simulations to settle. In addition, the quick tapering in the opposite direction had the effect of reducing the maximum lag that was developing in negative cross range coordinate.

4. PREDICTING PERFORMANCE FOR MULTISENSOR TRACKING OF A MANEUVERING TARGET

A method for predicting the performance of a multisensor system is developed in this section. The result

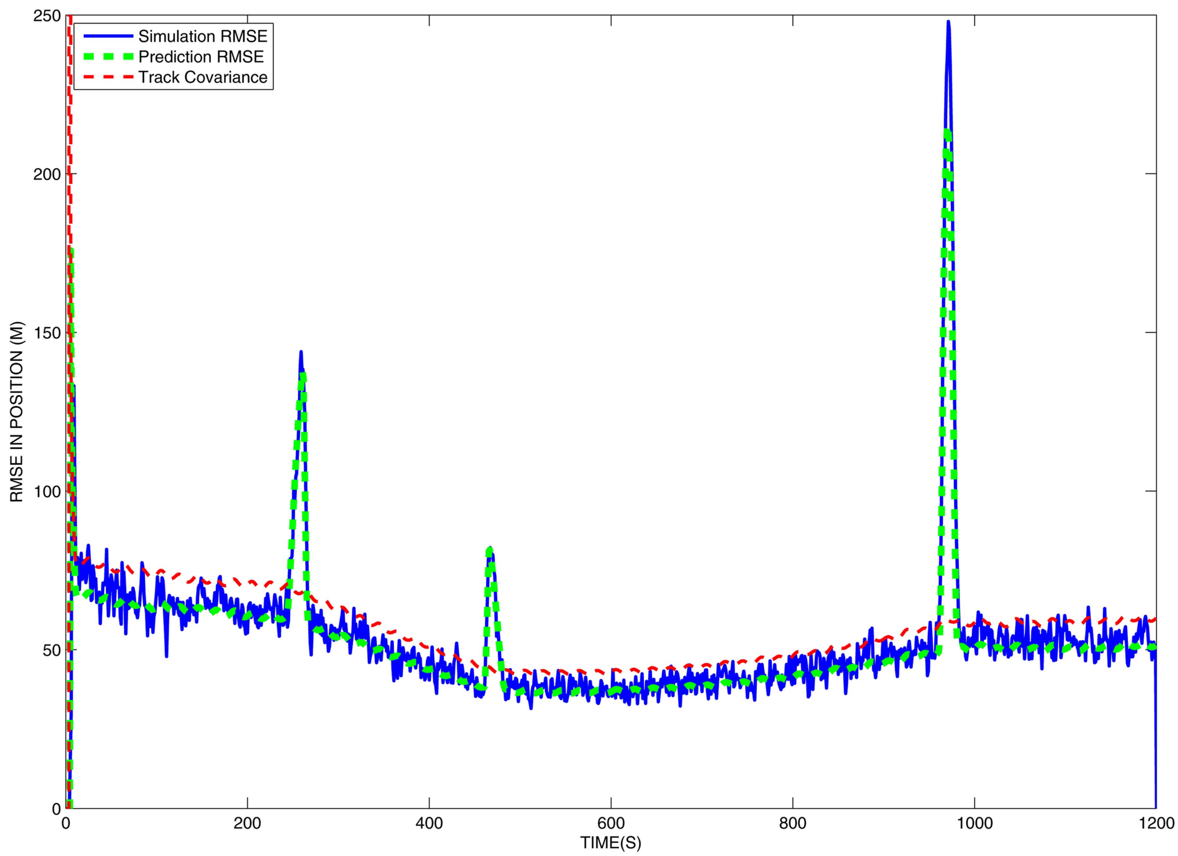

Fig. 2. RMSE in position for Sensor 1.

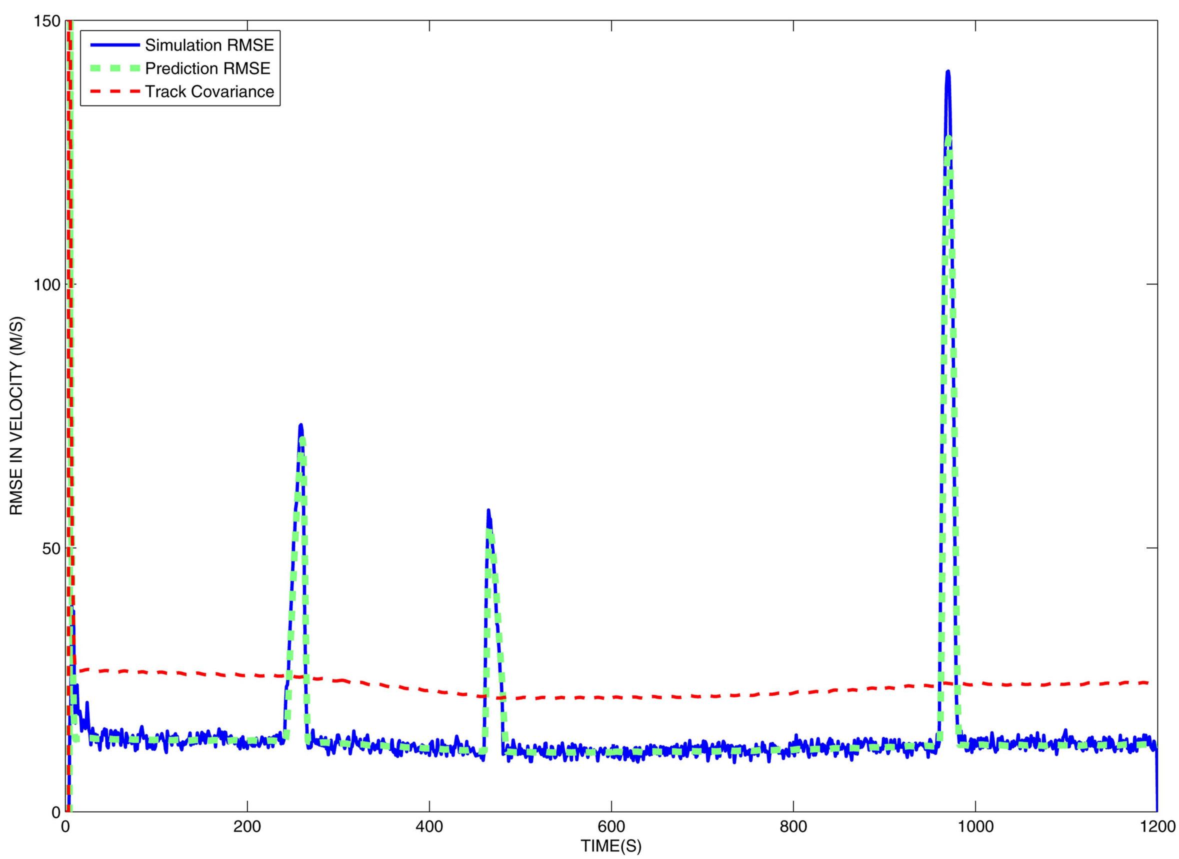

Fig. 3. RMSE in velocity for Sensor 1.

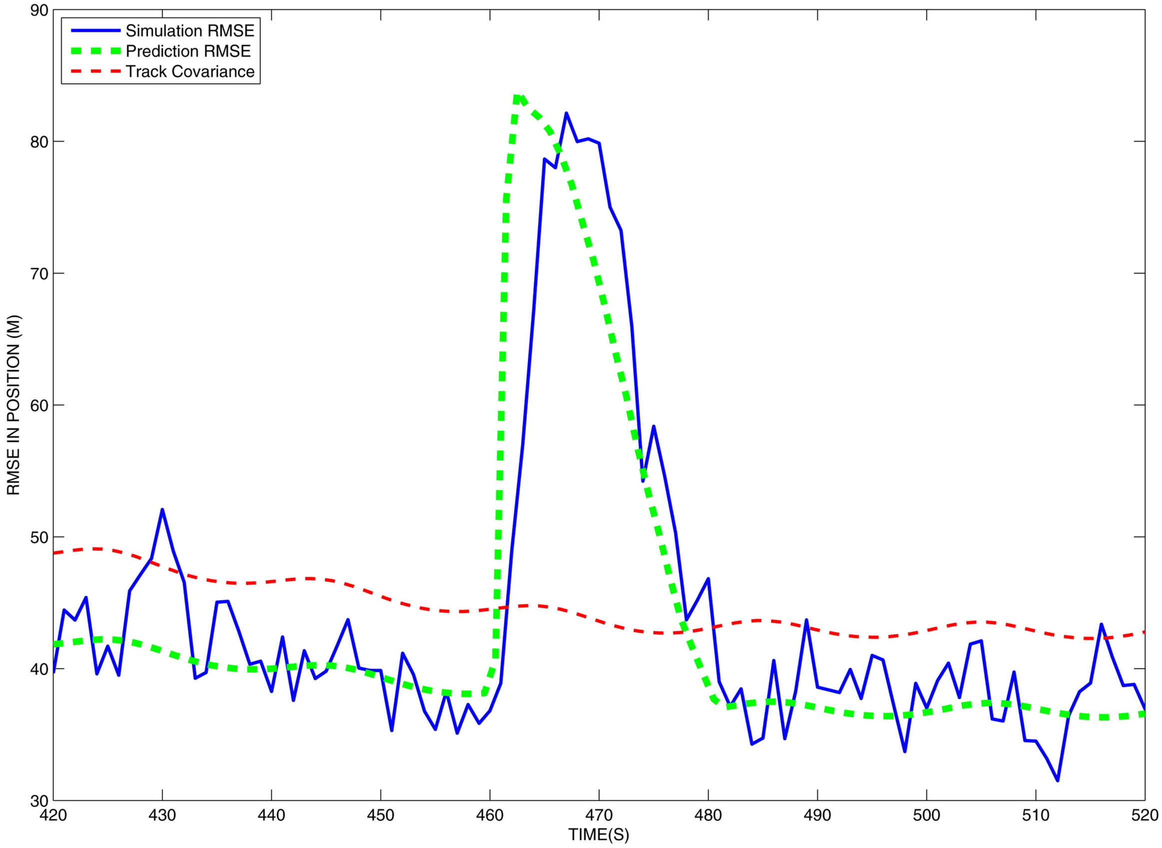

Fig. 4. RMSE during maneuver 2 for Sensor 1 with no moving average applied.

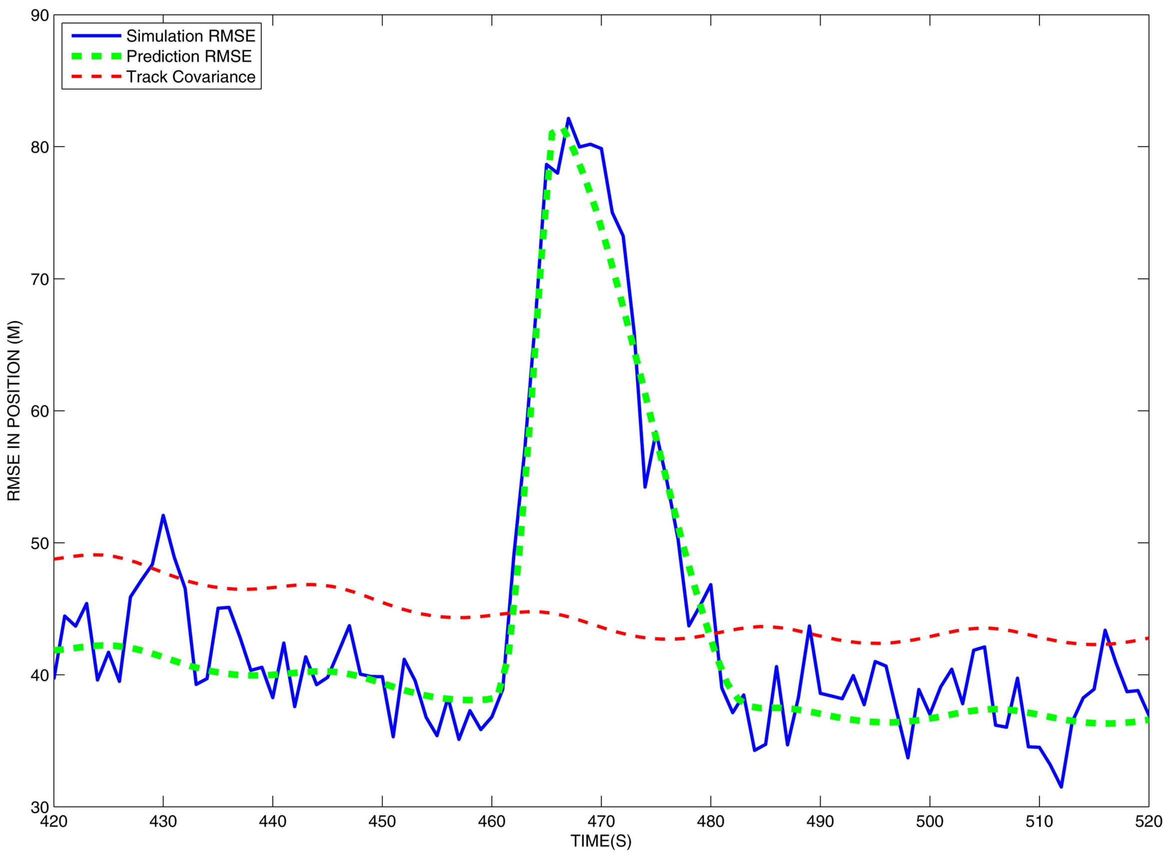

Fig. 5. RMSE during maneuver 2 for Sensor 1 with 5 sample moving average applied.

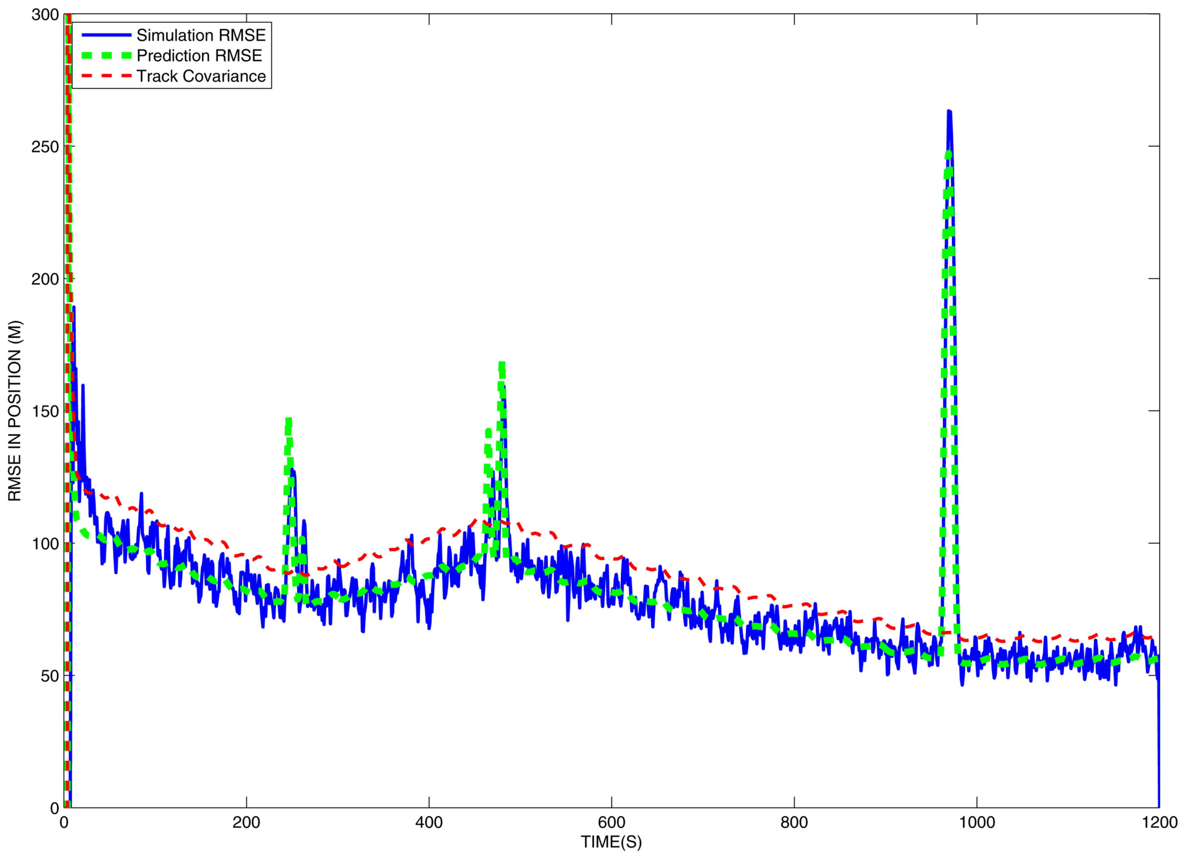

Fig. 6. RMSE in position for Sensor 2.

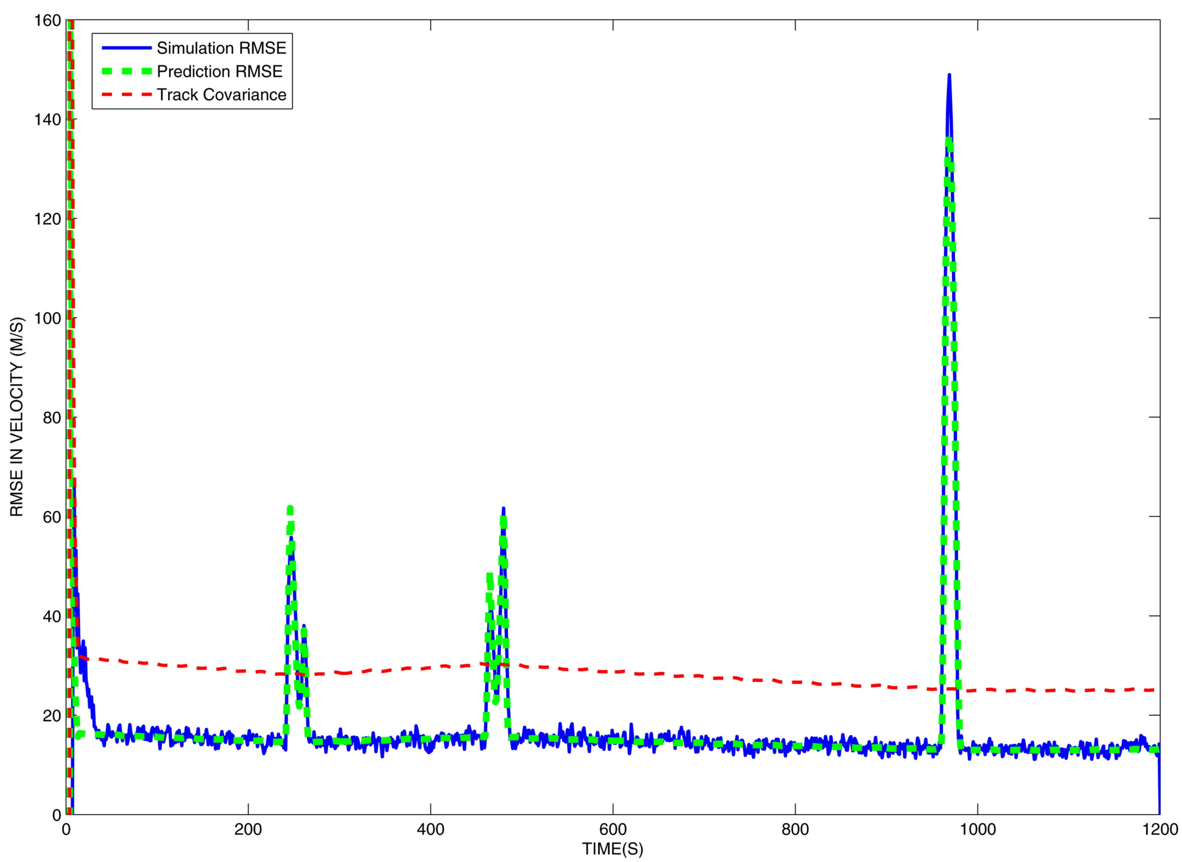

Fig. 7. RMSE in velocity for Sensor 2.

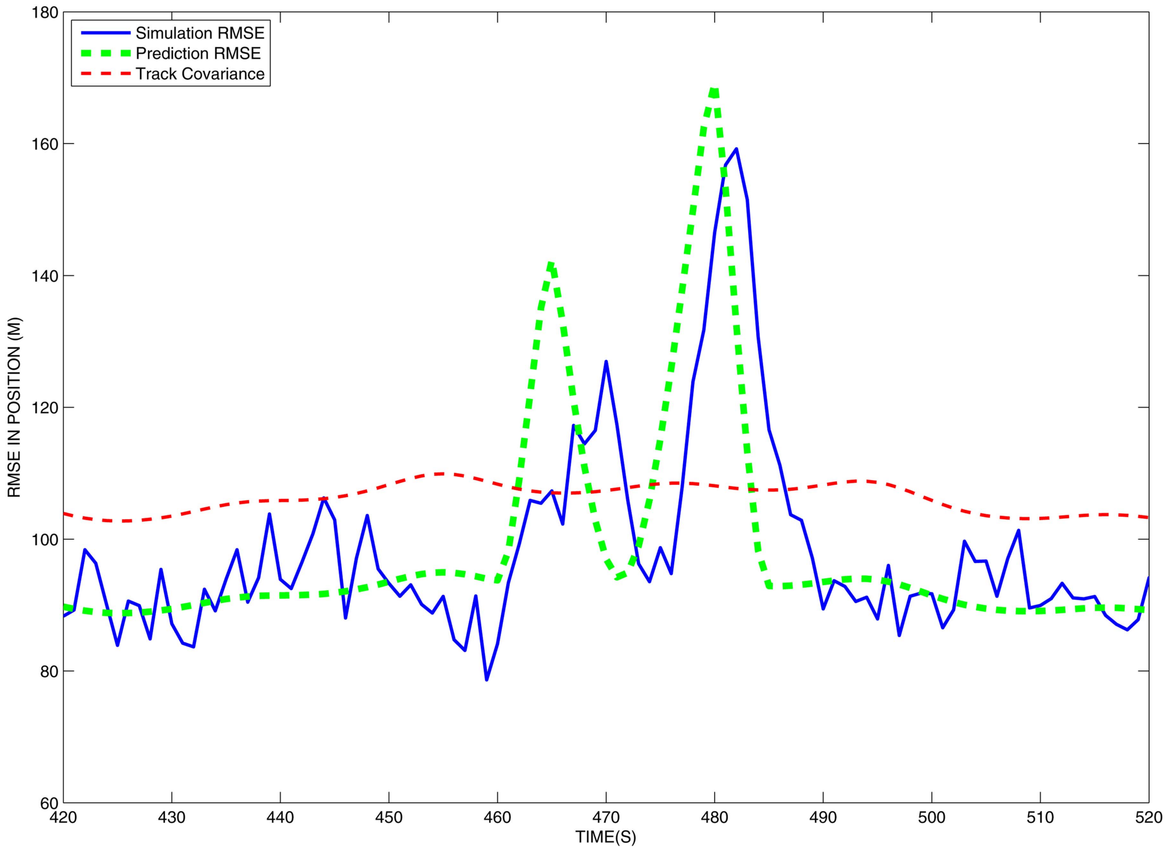

Fig. 8. RMSE during maneuver 2 for Sensor 2.

in (40) gives a direct relationship between the state estimate $X_{k|k}$, and the tracks from sensor i, $X^i_{k|k}$. Now consider a linear system of N independent track reports of the same target from N sensors dispersed on an oblate earth. Then,

$$\begin{bmatrix} X^1_{k|k} \\ X^2_{k|k} \\ \vdots \\ X^N_{k|k} \end{bmatrix} = \begin{bmatrix} M^1_k \\ M^2_k \\ \vdots \\ M^N_k \end{bmatrix} X_k + \begin{bmatrix} M^1_k L^1_k \\ M^2_k L^2_k \\ \vdots \\ M^N_k L^N_k \end{bmatrix} + \begin{bmatrix} W^1_k \\ W^2_k \\ \vdots \\ W^N_k \end{bmatrix} \quad (47)$$

where X_k is the unknown target state at time t_k, M^i_k is the rotation matrix of the state vector to sensor i that is aligned with a common tracking frame, L^i_k is the translation to sensor i from the common tracking frame in that frame, and W^i_k is the error in the track estimate from sensor i with covariance $P^i_{k|k}$. Assuming the errors in the tracks are independent, then

$$Y = \begin{bmatrix} X^1_{k|k} \\ X^2_{k|k} \\ \vdots \\ X^N_{k|k} \end{bmatrix} - \begin{bmatrix} M^1_k L^1_k \\ M^2_k L^2_k \\ \vdots \\ M^N_k L^N_k \end{bmatrix} \quad (48)$$

$$H = [M^1_k \quad M^2_k \cdots M^N_k] \quad (49)$$

$$H^T R^{-1} H = [(M^1_k)^T \quad (M^2_k)^T \cdots (M^N_k)^T] \times \begin{bmatrix} (P^1_{k|k})^{-1} & 0 & \cdots & 0 \\ 0 & (P^2_{k|k})^{-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & (P^N_{k|k})^{-1} \end{bmatrix} \begin{bmatrix} M^1_k \\ M^2_k \\ \vdots \\ M^N_k \end{bmatrix} = \sum_{i=1}^{N} (M^i_k)^T (P^i_{k|k})^{-1} M^i_k. \quad (50)$$

Substituting this result back into (40) yields the equation for the least squares estimate of a fused track.

$$X_{k|k} = \left[\sum_{i=1}^{N} (M^i_k)^T (P^i_{k|k})^{-1} M^i_k\right]^{-1} \times \left[\sum_{i=1}^{N} (M^i_k)^T (P^i_{k|k})^{-1} (X^i_{k|k} - M^i_k L^i_k)\right]. \quad (51)$$

Given a set of measurements with a variable number of Cartesian dimensions, a "coordinate pickoff" matrix of ones and zeros can be applied to each M^i_k so that the measurements for each sensor i are only a function of the state in the cartesian coordinates of interest. This method may be applied to a multiple sensor system that includes sensors with an arbitrary combination of one, two, or three dimensional cartesian measurements.

Array face coordinates is a cartesian system with origin at the sensor array face. In the native sensor frame, some radars measure range (r) and two angles commonly known as u and v. Since the errors in each sensor coordinate are assumed to be independent, the track covariance in orthogonal coordinates of the sensor frame (e.g., array face coordinates) can be approximated in a block diagonal form. The filter error covariance of each coordinate can be approximated with the alpha-beta filter covariance (19) with σ_w^2 representing the variance of the measurement errors in either the range or cross range coordinate. Thus, given the sensor location and measurement rate, measurement error variance for each coordinate, and process noise variance, the track indices can be defined for the range coordinate and the cross range coordinates. The error covariance for a track estimate from a three dimensional sensor i is represented as

$$P_{k|k}^{i} = \begin{bmatrix} P_{k|k}^{r} & 0_{2\times 2} & 0_{2\times 2} \\ 0_{2\times 2} & P_{k|k}^{u} & 0_{2\times 2} \\ 0_{2\times 2} & 0_{2\times 2} & P_{k|k}^{v} \end{bmatrix} \quad (52)$$

where it is assumed that the native measurement frame of the sensor is range (r) and two angles (u and v) and the covariance matrix for each coordinate is derived from (19) with σ_w^2 denoting the variance of the errors in either the range or cross range coordinates. The covariance is defined in a Cartesian coordinate system placed at the sensor with the axes aligned with the u and v measurements and the range vector to the target. The rotational transform and translation to sensor i, M_k^i and L_k^i, includes the sensor-to-local transformation as well as the transformation from sensor i to the common tracking frame. Embedded in this transformation is a rotation of the covariance in (52) into array face coordinates. For a track that requires prediction for time alignment before fusion, the standard covariance prediction equations of the Kalman filter can be used to compute the covariance for an m-step ahead prediction.

4.1. Bias of a Fused Track for a Deterministic Maneuver

Consider

$$\begin{aligned} X_{k|k} - X_k &= P_{k|k}^{f}\left[\sum_{i=1}^{N}(M_k^i)^T(P_{k|k}^i)^{-1}(X_{k|k}^i - M_k^i L_k^i)\right] - X_k \\ &= P_{k|k}^{f}\left[\sum_{i=1}^{N}(M_k^i)^T(P_{k|k}^i)^{-1}(X_{k|k}^i - M_k^i L_k^i - M_k^i X_k)\right] \\ &= P_{k|k}^{f}\left[\sum_{i=1}^{N}(M_k^i)^T(P_{k|k}^i)^{-1}(X_{k|k}^i - X_k^i)\right] \end{aligned} \quad (53)$$

where

$$P_{k|k}^{f} = \left[\sum_{i=1}^{N}(M_k^i)^T(P_{k|k}^i)^{-1}M_k^i\right]^{-1} \quad (54)$$

and X_k^i is the true target state in the coordinate frame of sensor i. Taking the expected value and substituting an estimate of the bias for the fused track as

$$B_{k|k}^{f} = E[X_{k|k} - X_k] = P_{k|k}^{f}\left[\sum_{i=1}^{N}(M_k^i)^T(P_{k|k}^i)^{-1}B_{k|k}^i\right] \quad (55)$$

where $B_{k|k}^i$ is the bias or maneuver lag in the track for sensor i that results from a maneuvering target. Thus, the biases and corresponding maneuver lags fuse with a gain proportional to the inverse of the covariance for the individual sensors. The bias of each coordinate of the track can be approximated with the bias or maneuver lag of the alpha-beta filter given by (30) with A_k representing the acceleration of the target in either the range or cross range coordinate for each sensor. The α and β are those that result from the tracking index. Thus, given the sensor location and measurement rate, measurement error variance for each coordinate, and process noise variance used for tracking in that sensor, the tracking indices can be defined for the range coordinate and the cross range coordinates. Thus, the bias or maneuver lag for a track estimate from a three dimensional sensor i is represented as

$$B_{k|k}^{i} = \begin{bmatrix} B_{k|k}^{r} \\ B_{k|k}^{u} \\ B_{k|k}^{v} \end{bmatrix} \quad (56)$$

where it is assumed the native measurement frame of the sensor is range (r) and two angles (u and v) and the bias vector for each coordinate is derived from (30) with A_k denoting the acceleration of the target in the range or cross range coordinates. The bias vector is defined in a Cartesian coordinate system placed at the sensor with the axes aligned with the u and v measurements and the range vector to the target. For a track that requires prediction for time alignment before fusion, (34) can be used to compute the bias or maneuver lag for an m-step ahead prediction and a maneuvering target.

4.2. SNO Covariance for a Fused Track

Taking the expected value of (51) for $\bar{X}_{k|k} = E[X_{k|k}]$ gives

$$X_{k|k} - \bar{X}_{k|k} = P_{k|k}^{f}\left[\sum_{i=1}^{N}(M_k^i)^T(P_{k|k}^i)^{-1}(X_{k|k}^i - \bar{X}_{k|k}^i)\right]. \quad (57)$$

Let the SNO covariance of the fused track be denoted as

$$S_{k|k}^{f} = E[(X_{k|k} - \bar{X}_{k|k})(X_{k|k} - \bar{X}_{k|k})^T]. \quad (58)$$

Assuming sensor errors in the sensor tracks are independent and taking the expected value, the SNO covariance of a fused track is approximately by

$$S_{k|k}^{f} = P_{k|k}^{f}\left[\sum_{i=1}^{N}(M_k^i)^T(P_{k|k}^i)^{-1}S_{k|k}^i(P_{k|k}^i)^{-1}M_k^i\right](P_{k|k}^f)^T \tag{59}$$

where $S_{k|k}^i$ is the SNO covariance for sensor i. Since the errors in each sensor coordinate tend to be independent, the SNO covariance in sensor coordinates can be approximated in a block diagonal form. The SNO covariance of each coordinate can be approximated with the SNO covariance of the alpha-beta filter (29) with σ_w^2 representing the variance of the measurement errors in either the range or cross range coordinate for each sensor. The α and β are those that results from the tracking index. Thus, given the sensor location and measurement rate, measurement error variance for each coordinate, and process noise variance used for tracking in that sensor, the tracking index can be defined for the range coordinate and the cross range coordinates. The SNO covariance for a track estimate from a three dimensional sensor i is represented as

$$S_{k|k}^i = \begin{bmatrix} S_{k|k}^r & 0_{2\times 2} & 0_{2\times 2} \\ 0_{2\times 2} & S_{k|k}^u & 0_{2\times 2} \\ 0_{2\times 2} & 0_{2\times 2} & S_{k|k}^v \end{bmatrix} \tag{60}$$

where it is assumed the native measurement frame of the sensor is range (r) and two angles (u and v) and the sensor-noise only covariance matrix for each coordinate is derived from (29) with σ_w^2 denoting the variance of the errors in either the range or cross range coordinates. The covariance is defined in a Cartesian coordinate system placed at the sensor with the axes aligned with the u and v measurements and the range vector to the target. For a track that requires prediction for time alignment before fusion, (33) can be used to compute the SNO covariance for an m-step ahead prediction.

5. EXAMPLE OF MULTISENSOR TRACKING FOR A MANEUVERING TARGET

As a numerical example of predicted performance for multiple sensors, consider two independent systems with a 1 Hz measurement rate. The radar measurements are corrupted with zero-mean errors that are Gaussian distributed and have standard deviations of 3 m, 1.1 mrad, and 1.1 mrad in range, azimuthal angle, and vertical angle, respectively. The location of each sensor is shown in Fig. 1. The multisensor tracking system is measurement based with measurements from two unbiased sensors shared over an ideal communications link. Measurement-to-track association performs perfectly given this particular multisensor and single target scenario. The composite tracker employs an extended Kalman filter with nearly constant velocity and process noise PSD of $q = 100\ \mathrm{m}^2/\mathrm{s}^3$ in each coordinate. At each scoring time, the RMSE of the filter state is computed and averaged over all Monte Carlo runs.

The tracking system configuration in the prediction codes is identical to that of the simulations. At each time step in the prediction codes, we compute the steady state error covariance for each sensor as shown in (54). The magnitude of the diagonals for the actual filter track covariance is also displayed in the plots to illustrate the inability of the track filter covariance to accurately predict track filter performance.

The predicted performance results for position and velocity are compared in Figs. 9 and 10 respectively to the results from Monte Carlo simulations with 25 runs. From these figures, it is evident the performance prediction methods presented in this work are more accurate than using the track filter covariance to predict performance. These figures also illustrate the improvement in tracking due to the geometric diversity of the sensors. As predicted, the simulations show the bias due to the maneuver lag is significantly reduced during maneuvers one and two. Due to the short time of these two maneuvers and the geometric diversity of the sensors, the bias in position for maneuvers one and two is insignificant.

Figures 9 and 10 also illustrate the inability of the track filter covariance to predict performance. Consistent with the single sensor case, the covariances in both position and velocity are larger than simulation results when the target is not maneuvering and smaller than simulation results when the target is maneuvering. Finally, although it is noted the predicted performance is slightly optimistic when compared to the simulated results during periods the target is not maneuvering, the predicted performance is still superior to the performance as predicted by the track filter covariance.

Figures 11 and 12 compare the predicted performance results from the single radar tracking example in Section 3 to the predicted performance of two independent radars fusing measurements for a single track. During non maneuvering times, it is apparent from the figures that a variance reduction benefit is obtained from the measurement fusion. The bottom plot in Fig. 12 highlights the improvement in variance reduction for the velocity.

Given a set of circumstances whereby it is known generally where a threat will be launched and what area protection is required, the performance prediction methods presented in this work can be used with predicted trajectories and a desired level of performance to determine optimal placement of sensor resources. Further, this method easily extends to any number of sensors and therefore can be used to help determine the number of sensors required to meet a desired level of performance.

6. CONCLUSIONS

Given the sensor location, target location and accelerations as a function of time, the sensor parameters,

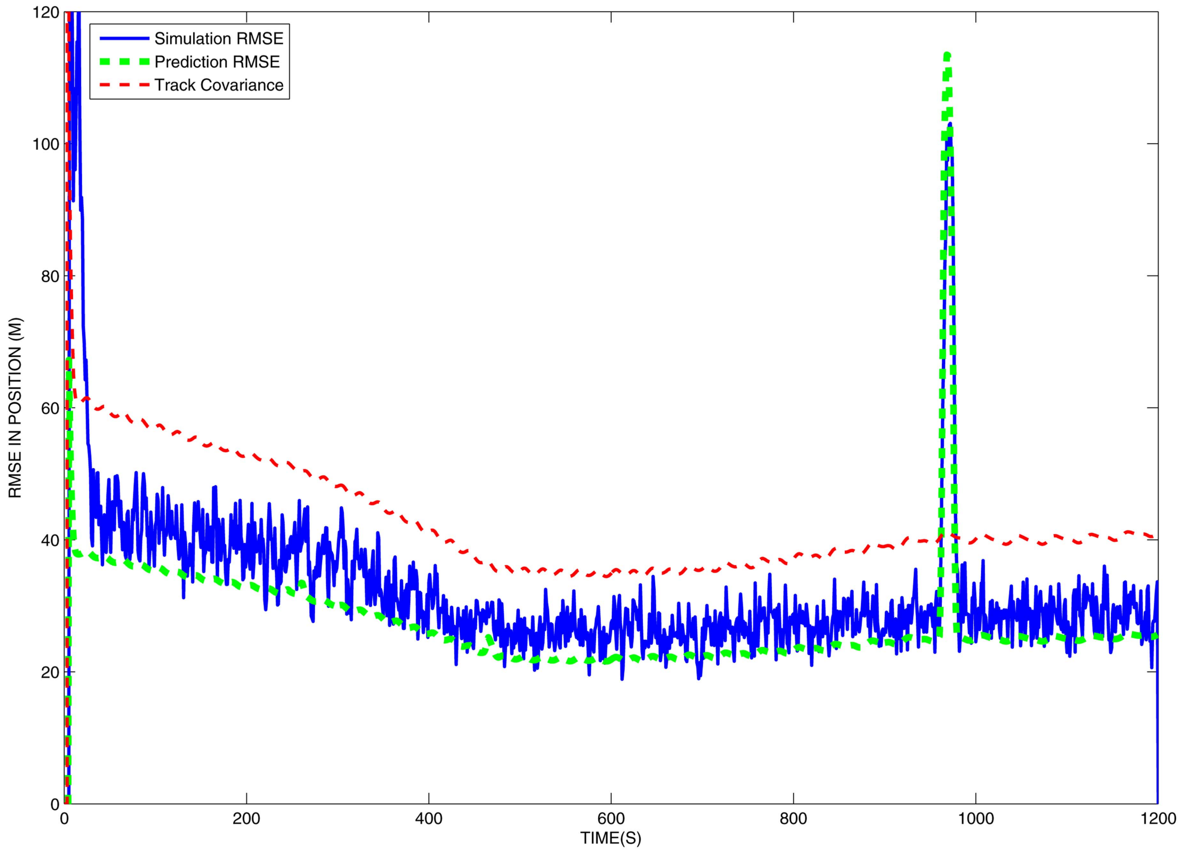

Fig. 9. RMSE in position for fused track from two sensors.

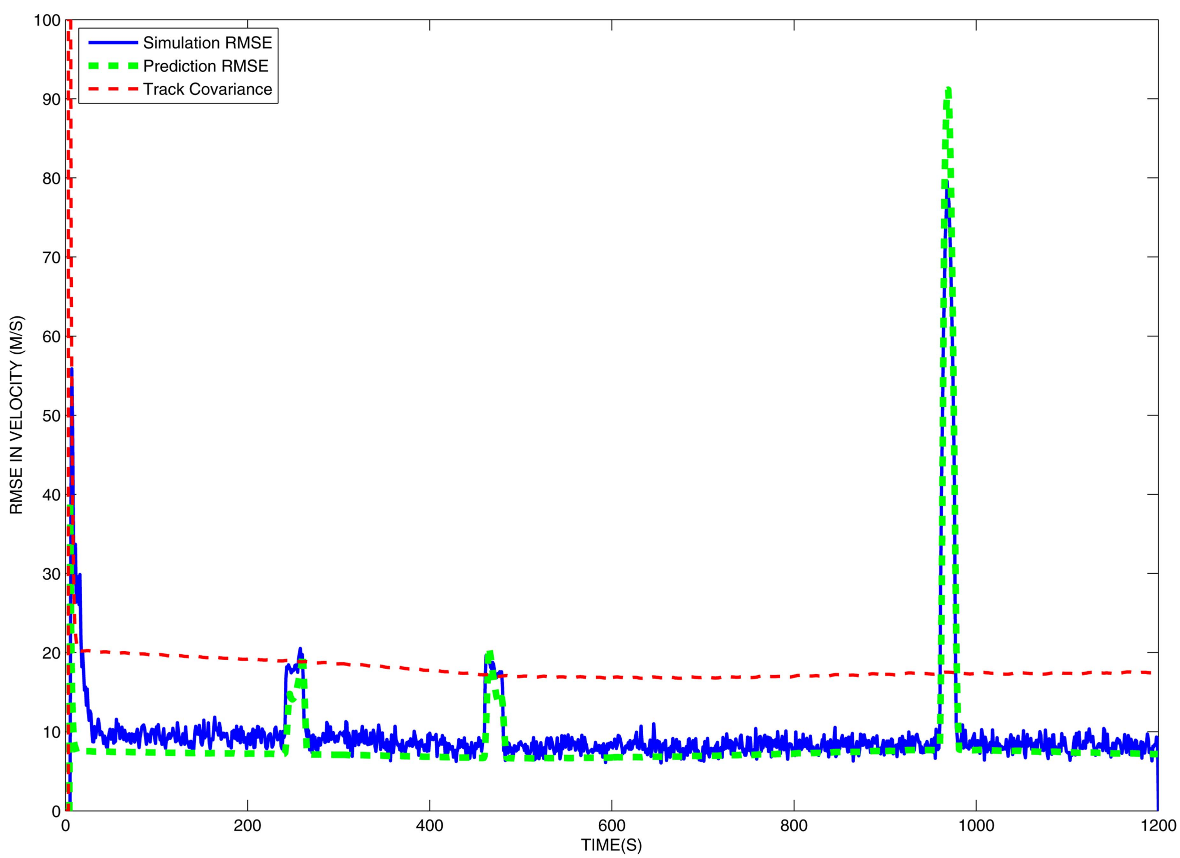

Fig. 10. RMSE in velocity for fused track from two sensors.

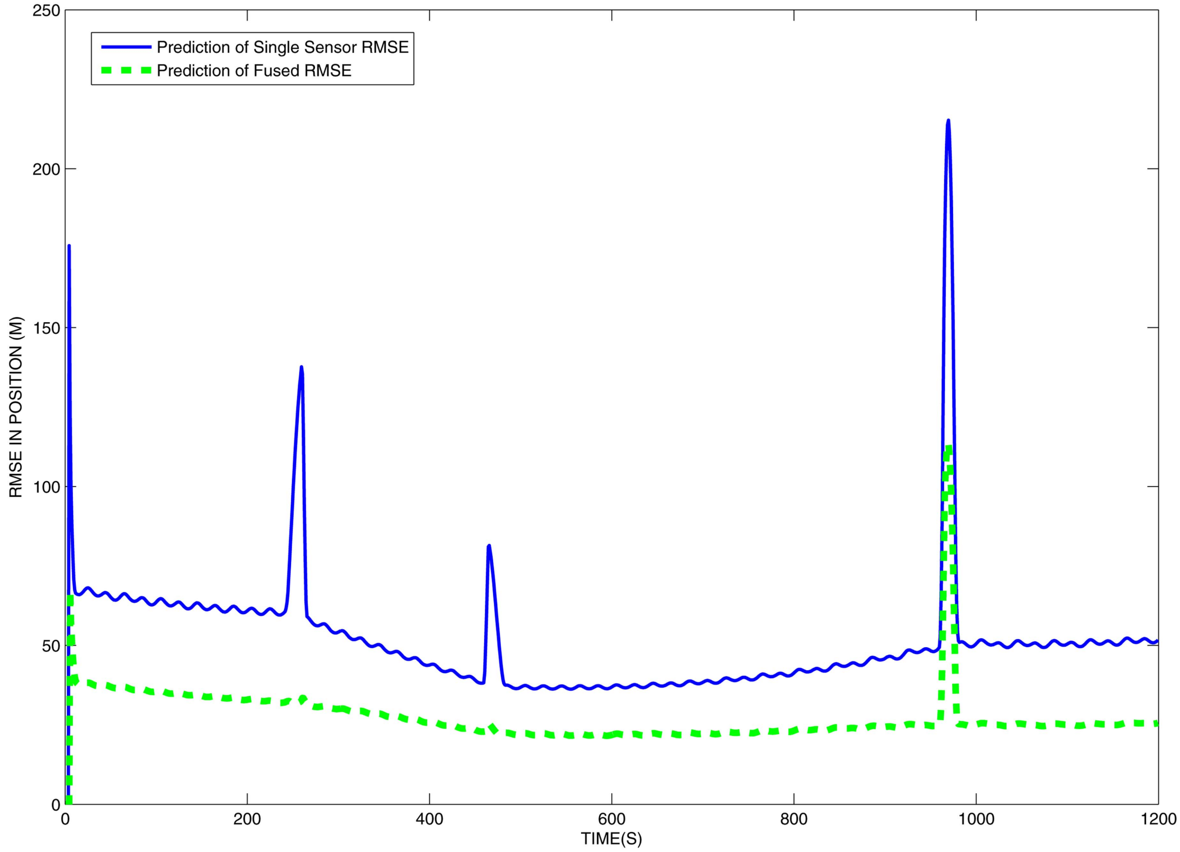

Fig. 11. Predicted RMSE in position for a track from single radar versus a fused track from two radars.

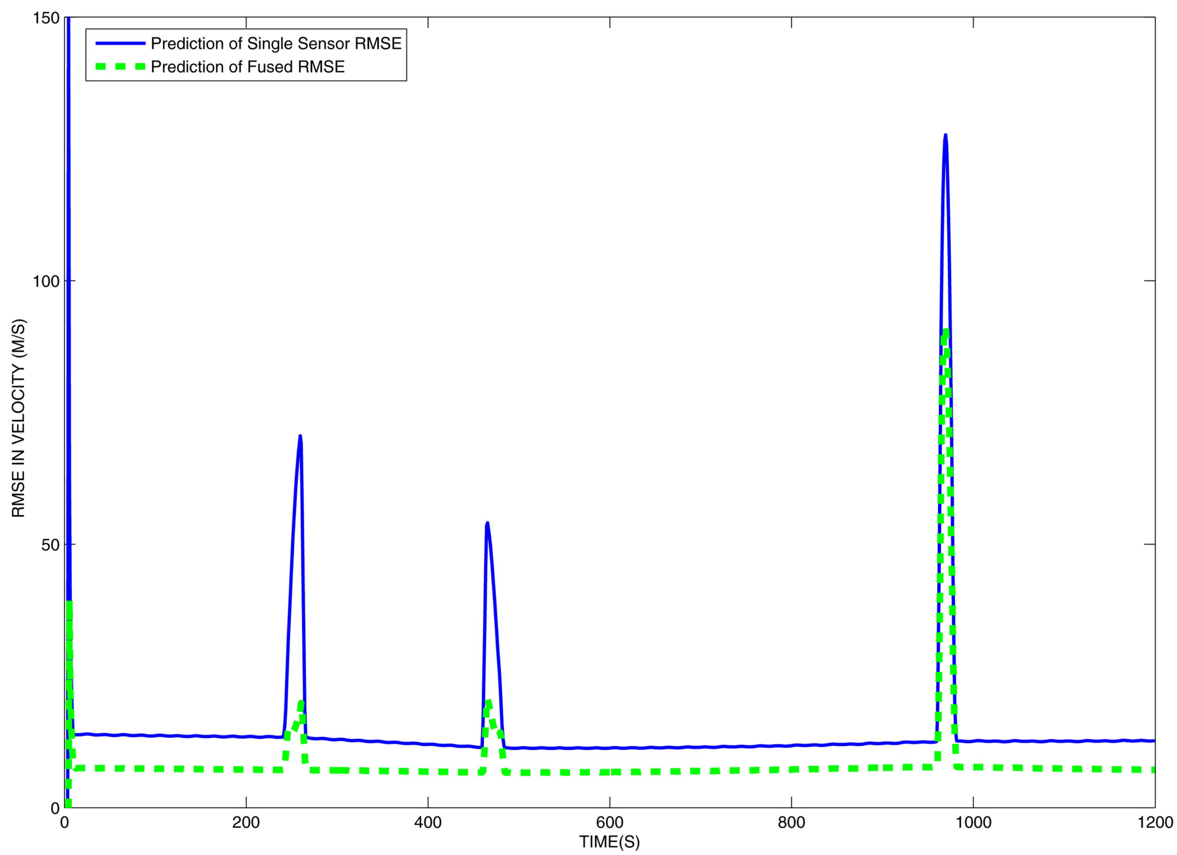

Fig. 12. Predicted RMSE in velocity for a track from single radar versus a fused track from two radars.

and the process noise covariance, it has been shown that one can predict track filtering performance of a multisensor system versus a maneuvering target. Since data association errors are ignored, the predicted performance should be considered as a lower bound for nearly constant velocity tracking. However, the predicted performance does not reflect the improvements in the track filtering that is expected from the use of the IMM estimator [1]. The predicted performance reflects that expected from nearly constant velocity track filtering. Variability between predicted and simulated results can be attributed to more complex modeling in the Monte Carlo simulation environment. While synchronized detections have been assumed in performance prediction methods, detections in simulations for multiple radars do not have synchronized dwell times due to missed detections. In addition, simulations have randomized starting times and logic devoted to resource allocation. Furthermore, complex simulations model performance degrading effects related to off beam centered tracking when targets enter regions near the radar field of view limits. While the predicted results do an remarkable job of capturing the effects of off-broadside tracking, the same can not be said for modeling the effects of off-beam center tracking. However, such conditions under which predicted results diverge from simulated results are understood and therefore, it is easy to detect and display them with the predicted data. The availability of this additional information allows for more accurate interpretation of the predicted performance data. Therefore, the predicted results offer a very reasonable confirmation of the simulated performance results and provide an analytical basis for use in real-time command and control processes and system engineering.

REFERENCES

[1] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan
Estimation with Applications to Tracking and Navigation.
John Wiley and Sons, Inc., New York, NY, 2001.

[2] Y. Bar-Shalom and W. D. Blair (Eds.)
Multitarget-Multisensor Tracking: Applications and Advances III.
Norwood, Massachusetts: Artech House, 2000.

[3] T. Ender, R. Leurck, B. Weaver, P. A. Miceli, W. D. Blair, P. D. West, and D. Mavris
Systems-of-systems analysis of ballistic missile defense architecture effectiveness though surrogate modeling and simulation.
In *Proceedings of the 2nd IEEE International Systems Conference*, **23** (Apr. 2008), 333–340.

[4] P. Tichavsky, C. H. Muravchik, and A. Nehorai
Posterior Cramer-Rao bounds for discrete-time nonlinear filtering.
IEEE Transactions on Signal Processing, **46** (May 1998), 1386–1396.

[5] A. Farina, S. Immediata, B. Ristic, and L. Timmoneri
CRLB with $P_d < 1$ for fused tracks.
2005 8th International Conference on Information Fusion, July 2005, 25–28.

[6] C. Hue, J-P. Le Cadre, and P. Perez
Posterior Cramer-Rao bounds for multi-target tracking.
IEEE Transactions on Aerospace and Electronic Systems, **42**, 1 (Jan. 2006), 37–49.

[7] M. L. Hernandez, B. Ristic, and A. Farina
A performance bound for manoeuvring target tracking using best-fitting Gaussian distributions.
2005 7th International Conference on Information Fusion, July 2005, 1–8.

[8] X. R. Li and Y. Bar-Shalom
A hybrid conditional averaging technique for performance prediction of algorithms with continuous and discrete uncertainties.
Proceedings 1994 American Control Conference, June 1994, 1530–1534.

[9] X. R. Li and Y. Bar-Shalom
Performance prediction of the interacting multiple model algorithm.
IEEE Transactions on Aerospace and Electronic Systems, **29**, 3 (July 1993), 755–770.

[10] C. W. Frei and L. Y. Pao
Non-simulation performance prediction methods for different implementations of a multisensor fusion algorithm.
Proceedings 13th World Congress of the International Federation of Automatic Control, July 1996, 269–274.

[11] X. R. Li and Y. Bar-Shalom
Stability evaluation and track life of the PDAF for tracking in clutter.
IEEE Transactions on Automatic Control, **36**, 5 (May 1991), 588–602.

[12] R. W. Osborne and W. D. Blair
Update to the hybrid conditional averaging performance prediction of the IMM algorithm.
IEEE Transactions on Aerospace and Electronic Systems, **47**, 4 (Oct. 2011), 2967–2974.

[13] W. D. Blair
Fixed-gain, two-stage estimators for tracking maneuvering targets.
Technical report NSWCDD/TR-92/297, Naval Surface Warfare Center Dahlgren Division, Dahlgren, VA, July 1992.

[14] W. D. Blair
Fixed-gain, two-stage estimators for tracking maneuvering targets.
IEEE Transactions on Aerospace and Electronic Systems, (July 1993), 1004–1014.

[15] W. D. Blair and Y. Bar-Shalom
Tracking maneuvering targets with multiple sensors: Does more data always mean better estimates?
IEEE Transactions on Aerospace and Electronic Systems, **32** (Jan. 1996), 450–456.

[16] R. Chen and W. D. Blair
Aeolotropic filter design and measurement gating for remote tracking with gridlock error.
Proceedings of 29th IEEE Southeastern Symposium on System Theory, Cookeville, TN, Mar. 1997.

[17] O. E. Drummond, W. D. Blair, G. C. Brown, T. L. Ogle, Y. Bar-Shalom, R. L. Cooperman, and W. H. Barker
Performance assessment and comparison of various tracklet methods for maneuvering targets.
Signal Processing, Sensor Fusion, and Target Recognition XII, **5096** (2003), 1–26.

[18] V. Jain and W. D. Blair
Tracking with LFM waveforms.
IEEE Transactions on Aerospace and Electronic Systems, (Apr. 2009).

[19] J. E. Gray and W. Murray
The response of the transfer function of an alpha-beta filter to various measurement models.
Proceedings Southeastern Symposium on System Theory, **23** (Mar. 1991), 389–393.

[20] W. D. Blair
Design of nearly constant velocity track filters for tracking maneuvering targets.
11th International Conference on Information Fusion, July 2008, 1–7.

[21] A. Gelb
Applied Optimal Estimation.
The M.I.T. Press, Cambridge, MA, 1974.

W. Dale Blair received the B.S. and M.S. degrees in electrical engineering from Tennessee Technological University, Cookeville, in 1985 and 1987, and the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, in 1998.

He is a principal research engineer with the Georgia Tech Research Institute (GTRI) and currently serves as the technical director for the C2BMC Knowledge Center of the Missile Defense Agency (MDA). While a graduate research assistant, Dr. Blair performed robotic controls research for the Center of Excellence of Manufacturing Research and Technology Utilization at Tennessee Technological University. In 1987, he joined the Naval Systems Division of FMC Corporation in Dahlgren, VA as an electrical engineer, where his work involved the development and evaluation of new algorithms for weapons control. In 1990, Dr. Blair joined the Naval Surface Warfare Center, Dahlgren Division (NSWCDD) in Dahlgren, VA. At NSWCDD, he originated two benchmark problems for target tracking and radar resource allocation, which served as themes for invited sessions at the 1994 and 1995 American Control Conferences. He also led a project that demonstrated through a real-time tracking experiment that modern tracking algorithms can be utilized to reduce the radar time and energy required by a phased array radar to support surveillance tracking. Since joining GTRI in 1997, he has led a multiorganizational team in the development of multiplatform-multisensor-multitarget benchmarks to both air defense and ballistic missile defense. His projects at GTRI focus mostly the modeling and simulation and algorithm assessment associated with the sensor netting for the battle management, command, and control for the ballistic missile defense system. He also leads projects related multiple input, multiple output (MIMO) radar, advanced radar data processing, and sensor netting for detection of chemical and biological weapons.

Dr. Blair is a Fellow of the IEEE and recipient of the 2001 IEEE Nathanson Award for Outstanding Young Radar Engineer. He served as the Editor for Radar System for *IEEE Transactions on Aerospace and Electronic Systems* (T-AES) 1996–99 and Editor-In-Chief (EIC) for IEEE T-AES from 1999–2005. He is the founding Editor-In-Chief for the *International Journal on Advances in Information Fusion* under the International Society for Information Fusion. Dr. Blair also serves on the Board of Governors of IEEE Aerospace and Electronic Systems Society (AESS) from 1998–2003 and 2005–2010. While a member of the IEEE AESS BoG, he served as Associate Vice President for Publications for 2007–10 and Chair of the Strategic Planning Committee for 2009–10, originated the Target Tracking System Panel, and initiated the Systems Panel for Cyber Security Systems. He has received technical awards that include the NSWCDD Technical Excellence Award in 1991 for his contributions to the development of innovative techniques for target trajectory estimation and prediction and the NSWCDD Independent Exploratory Development Excellence Award in 1993. His research interests include radar signal processing and control, resource allocation for multifunction radars, multisensor resource allocation, tracking maneuvering targets, and multisensor integration and data fusion. Dr. Blair's research is reported in over two hundred articles which include 38 refereed journal articles. He served as a lecturer in 1993, 1994, and 1998 for the UCLA Extension Program in three short courses related to advanced topics in target tracking. He recently originated and coordinates three short courses, *Target Tracking in Sensor Systems*, *Target Tracking Concepts*, and *Advanced Target Tracking for Ballistic Missile Defense*, for the Profession Education Department of the Georgia Institute of Technology. Dr. Blair is coeditor and coauthor of the book, *Multitarget-Multisensor Tracking: Advances and Applications III*, and the author of chapter 19 "Radar Tracking Algorithms" of the new edition of *Principles of Modern Radar*. He is also the host and coordinator of the ONR/GTRI Workshop on Target Tracking and Sensor Fusion for 1998 through 2011.

Paul Miceli received the B.S. degree in electrical engineering from Manhattan College and the M.S. degree in electrical engineering from the Georgia Institute of Technology.

He is a senior research engineer for the Electronics Systems Branch of the Georgia Tech Research Institute. During his time in graduate school, his research focused on the development of signal processing algorithms for the suppression of urban noise in radar warning receivers. Since joining the Georgia Tech research faculty full time in May 2002, he has contributed to numerous programs in applications involving target tracking, radar, signal processing, and modeling and simulation. In addition to infrastructure and algorithm development, Mr. Miceli has also long been involved with applied research studies in the fields of multi-sensor fusion and radar performance prediction.