JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 21, No. 4, July–August 1998

Comparison of Four Filtering Options for a Radar Tracking Problem

John A. Lawton*
U.S. Naval Surface Warfare Center, Dahlgren, Virginia 22448-5100
Robert J. Jesionowski†
Sparta, Inc., Arlington, Virginia 22209-1603
and
Paul Zarchan‡
Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts 02139-3563

Four different filtering options are considered for the problem of tracking an exoatmospheric ballistic target with no maneuvers. The four filters are an alpha-beta filter, an augmented alpha-beta filter, a decoupled Kalman filter, and a fully coupled extended Kalman filter. These filters are listed in the order of increasing computational complexity. All of the filters can track the target with some degree of accuracy. Although the pure alpha-beta filter appreciably lags the other filters in performance for this problem, its augmented version is very competitive with the extended Kalman filter under benign conditions. Perhaps the most surprising result is that under all conditions examined, the decoupled (linear) Kalman filter, which is at least an order of magnitude less computationally complex than the fully coupled extended Kalman filter, performs essentially the same as the extended Kalman filter.

Introduction

KALMAN filtering became immediately popular when it was first introduced in the 1960s, even though the initial journal papers were incomprehensible to many experienced engineers.[1,2] This new filtering technique was popular in the academic community because it could be shown that the Kalman filter was optimal in some sense, and in addition, Kalman filtering was also popular in industry because its form was ideal for implementation on a digital computer. After three decades, applications of Kalman filtering have proliferated to the point where today many younger engineers do not even know the existence of other types of filters.[3,4]

In some applications today, because of the availability of computational plenty, the standard approach is often to implement a fully coupled Kalman filter without consideration of simpler implementations or even other types of filters. In this paper, we will consider several types of filters, ranging from an alpha-beta filter (non-Kalman) to a fully coupled Kalman filter, for an application in which a ship radar is tracking an exoatmospheric target. It is assumed that the target is not maneuvering, and that the filter must estimate the position and velocity of the target based on range and angle measurements.

Four different types of filters will be compared in terms of the quality of their estimates and in terms of their computational complexity. The filters will be first compared under benign conditions in which all the measurement data are available and then compared under conditions in which large portions of the measurements are lost.

Contrary to popular opinion, it will be shown that under benign conditions, for this exoatmospheric tracking problem, three of the four filters yield approximately the same performance. Under these circumstances, an augmented alpha-beta filter can yield excellent performance at orders of magnitude less computational complexity than a fully coupled Kalman filter. Under conditions in which there are large blackout periods, it will be shown that a decoupled Kalman filter can yield performance similar to that of a fully coupled filter at at least an order of magnitude less computational complexity.

Presented as Paper 16-01 at the AIAA/BMDO 6th Annual Technology Readiness Conference, San Diego, CA, Aug. 18–22, 1997; received Sept. 15, 1997; revision received Jan. 17, 1998; accepted for publication Jan. 17, 1998. This paper is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

*Aerospace Engineer, Space and Weapon Systems Analysis Division, Dahlgren Division. Member AIAA.
†Senior Engineer, Systems Analysis Division. Member AIAA.
‡Principal Member of Technical Staff. Associate Fellow AIAA.

Filter Formulations

Four different filter designs will be presented. The first three are all decoupled filters, meaning that the whole six-dimensional state is broken up into three two-dimensional states, which are updated in the filter equations independently. Each two-dimensional state is composed of position and velocity along one of the Earth-fixed Cartesian coordinate directions in an east-north-up (ENU) coordinate system centered at the sensor.

The first filter is an alpha-beta filter, which can be derived as a steady-state two-dimensional Kalman filter.[5] Because alpha-beta filters have constant gains (for a given sensor to target range), and it is well known the Kalman filter's typically steadily decreasing gain is optimal (for a linear problem), a simple design is to tack onto the alpha-beta filter a growing-memory filter during the settling portions. That is, a growing-memory filter (which is equivalent to a zero-process-noise Kalman filter) is used during the initial portions of the observation of the target,[6] and the alpha-beta filter is used after the initial settling is finished. This hybrid filter will be called the growing-memory/alpha-beta (GMAB) filter. The third filter is just a decoupled Kalman filter; that is, it is three independent two-dimensional Kalman filters, one for each coordinate direction.

The fourth filter is a multiple coordinate system, fully coupled extended Kalman filter.[7] The state and covariance propagations are done in an Earth-centered inertial (ECI) coordinate system, and the measurement updates are done in a range/direction-cosine coordinate system. The dynamics model is Keplerian motion.

Decoupled Filters

All three of the decoupled filters are designed with a piecewise-constant acceleration model, assumed to act uniformly on all neighboring trajectories. Hence, each two-dimensional state $\boldsymbol{x}$, with the first component of $\boldsymbol{x}$ being position and the second component being velocity, is assumed to have the following discrete-time dynamics, propagating the state from the time of the kth measurement t_k to that of the $(k+1)$st:

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}(k+1,k)\,\boldsymbol{x}_k + \boldsymbol{m}_k(T) + \boldsymbol{\omega}_k \tag{1}$$

where

$$\Phi(k+1,k) \triangleq \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \tag{2}$$

$$T \triangleq t_{k+1} - t_k \tag{3}$$

where $\boldsymbol{m}_k(T)$ is the motion due to the constant acceleration acting over the interval T, and ω_k is the process noise, with

$$E(\omega_k) = 0, \qquad E\left(\omega_k \omega_k^T\right) = \boldsymbol{Q} \tag{4}$$

Because the position vector is measured by a radar, the scalar measurement is

$$z = \boldsymbol{H}\boldsymbol{x} + v \tag{5}$$

where

$$H = [1 \quad 0] \tag{6}$$

and v is a white noise disturbance random variable, with

$$E(v) = 0, \qquad E(v^2) = R = \sigma_m^2 \tag{7}$$

For all of the decoupled filters, the measurement standard deviation σ_m will be approximated by

$$\sigma_m = \bar{\rho} \cdot \sigma_a \tag{8}$$

where $\bar{\rho}$ is the current estimated range to the target, and σ_a is the angular error of the radar.

All of the filters used for this study have process noise predicated upon a piecewise-constant white acceleration model.[8] Hence, for the decoupled filters,

$$\boldsymbol{Q} = \begin{bmatrix} T^4/4 & T^3/2 \\ T^3/2 & T^2 \end{bmatrix} \cdot \epsilon^2 \tag{9}$$

where ϵ is a parameter with dimensions of acceleration. Theoretically, ϵ accounts for the lack of knowledge of the true acceleration in the model. In practice, it is used as a tuning parameter for the filter. That is, its value is varied until the filter performs acceptably under a wide variety of situations. For this study, this tuning process led to a value of 1 m/s being used for all of the filters except for the pure alpha-beta filter (which will be discussed in the results section).

Whereas the filter theory has its dynamics predicated on Eq. (1), the decoupled filters actually use a piecewise-constant acceleration model that is computed based on the current state estimate, for the propagation of the state vector from the kth measurement to the $(k+1)$st measurement. The total acceleration vector, denoted by $\boldsymbol{a} = [a_e \;\; a_n \;\; a_u]^T$ (e, n, and u represent east, north, and up, respectively), is the difference of the inverse-square gravity acceleration and the Coriolis and centripetal accelerations, computed at time t_k. Therefore, with $\bar{\boldsymbol{x}}_k$ representing the filter's estimate of the state at time t_k before measurement (for any of the three coordinate directions) and $\hat{\boldsymbol{x}}_k$ representing the estimate at time t_k after measurement,

$$\bar{\boldsymbol{x}}_{k+1} = \Phi(k+1,k)\,\hat{\boldsymbol{x}}_k + \begin{bmatrix} \frac{1}{2} a_i T^2 \\ a_i T \end{bmatrix} \tag{10}$$

where a_i (with $i \in \{e, n, u\}$) is the component of acceleration in this particular direction. The term $\boldsymbol{a}$ is computed by

$$\boldsymbol{a} = \boldsymbol{g} - \boldsymbol{c} - \boldsymbol{p} \tag{11}$$

where

$$\boldsymbol{g} = -(\mu/R^3)\boldsymbol{R} \tag{12}$$

$$\boldsymbol{c} = 2\boldsymbol{\omega} \times \boldsymbol{v} \tag{13}$$

and

$$\boldsymbol{p} = \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{R}) \tag{14}$$

with $\boldsymbol{r} = [r_e \;\; r_n \;\; r_u]^T$ and $\boldsymbol{v} = [v_e \;\; v_n \;\; v_u]^T$ being the position and velocity vectors in ENU at time t_k,

$$\boldsymbol{R} = \boldsymbol{r} + \begin{bmatrix} 0 \\ 0 \\ R_E \end{bmatrix} \tag{15}$$

and $\boldsymbol{\omega}$ is the angular motion of the Earth in ENU. Denoting the latitude of the sensor by ϕ and the rotation rate of the Earth by ω_E,

$$\boldsymbol{\omega} = \omega_E \begin{bmatrix} 0 \\ \cos\phi \\ \sin\phi \end{bmatrix} \tag{16}$$

which yields

$$\boldsymbol{c} = \begin{bmatrix} 2\omega_E(v_u \cos\phi - v_n \sin\phi) \\ 2\omega_E v_e \sin\phi \\ -2\omega_E v_e \cos\phi \end{bmatrix} \tag{17}$$

and

$$\boldsymbol{p} = \begin{Bmatrix} -\omega_E^2 r_e \\ \omega_E^2[(r_u + R_E)\cos\phi - r_n \sin\phi]\sin\phi \\ -\omega_E^2[(r_u + R_E)\cos\phi - r_n \sin\phi]\cos\phi \end{Bmatrix} \tag{18}$$

The components of $\boldsymbol{c}$ and $\boldsymbol{p}$ are used in Eq. (10) to propagate to the next measurement time.

The only difference between the three decoupled filters in this study is the choice of the values used in the 2×1 gain matrix $\boldsymbol{K}$ for use in the usual update equation

$$\hat{\boldsymbol{x}} = \bar{\boldsymbol{x}} + \boldsymbol{K}(z - \boldsymbol{H}\bar{\boldsymbol{x}}) \tag{19}$$

Alpha-Beta Filter

The name of the alpha-beta filter comes from the parameters alpha and beta (α and β), which are defined such that the gain matrix is

$$\boldsymbol{K} = \begin{bmatrix} \alpha \\ \beta/T \end{bmatrix} \tag{20}$$

Because $\boldsymbol{K}$, by definition, is the gain after the Kalman filter has reached steady state, α and β can be obtained by setting the state covariance at the current measurement to that of the previous measurement, in the Kalman filter recursion equations, and then algebraically solving for α and β. For algebraic convenience, define the so-called maneuvering index:

$$\lambda = (\epsilon \cdot T^2)/(\bar{\rho} \cdot \sigma_a) \tag{21}$$

[where ϵ is the process noise parameter used in Eq. (2), so that λ is essentially the ratio of the motion uncertainty to the measurement uncertainty]. Then α and β can be written as[5]

$$\alpha = -\tfrac{1}{8}\left(\lambda^2 + 8\lambda - (\lambda+4)\sqrt{\lambda^2 + 8\lambda}\right) \tag{22}$$

and

$$\beta = \tfrac{1}{4}\left(\lambda^2 + 4\lambda - \lambda\sqrt{\lambda^2 + 8\lambda}\right) \tag{23}$$

Hybrid GMAB Filter

The alpha-beta part of this hybrid filter is the same as described in the previous subsection. Bar-Shalom and Li[6] show that the gains can be expressly written as a function of k, the measurement update number (or time index), for the case of a zero-process-noise Kalman filter (a Kalman filter with $\omega_k = 0$) with the dynamics given in Eq. (1). The result is

$$\boldsymbol{K} = \begin{bmatrix} \dfrac{4k+2}{(k+1)(k+2)} \\ \dfrac{6}{(k+1)(k+2)T} \end{bmatrix} \tag{24}$$

These gains monotonically decrease with time because the larger amount of data incorporated (or stored) in the state estimate from past measurements indicates that state needs to be altered less and less based on an individual measurement. That is, the memory (or information) resident in the state estimate is growing; hence, this is also called a growing-memory filter. For small k, these gains are much larger than the alpha-beta gains. Also, for small k, these gains are very similar to Kalman filter gains (with nonzero process noise) because they are more dominated by the large decreases in the state covariance due to additional measurement updates than by the process noise. These gains tend to zero for large k, however, because of the no-noise assumption. Hence, it is advantageous to transition to the steady-state Kalman filter (the alpha-beta filter) for some judiciously chosen value of k. For this study, the simple strategy is used of switching to the alpha-beta filter when the position gain (the 1, 1 element of $\boldsymbol{K}$) from Eq. (24) is smaller than α of Eq. (22).

Decoupled Kalman Filter

The Kalman filter approach is to keep track of an estimate of the state errors in the covariance matrix $\boldsymbol{P}$ and the covariance of the so-called innovation or residual S and then compute the gain as the ratio between the two matrices

$$\boldsymbol{K} = \bar{\boldsymbol{P}}\boldsymbol{H}^T S^{-1} \tag{25}$$

The innovation is the $(\boldsymbol{z} - \boldsymbol{H}\bar{\boldsymbol{x}})$ term in Eq. (19); $\boldsymbol{H}\bar{\boldsymbol{x}}$ could be labeled $\bar{\boldsymbol{z}}$ because it is the estimate of the measurement before measurement, and so the difference is the measurement residual, also called the innovation because it is the mechanism for injecting new information into the state estimate. The covariance of this innovation is

$$S = \boldsymbol{H}\bar{\boldsymbol{P}}\boldsymbol{H}^T + R \tag{26}$$

The state covariance is propagated between measurements by

$$\bar{\boldsymbol{P}}^{k+1} = \Phi(k+1,k)\hat{\boldsymbol{P}}^k\Phi^T(k+1,k) + \boldsymbol{Q} \tag{27}$$

and is updated at measurement time by

$$\hat{\boldsymbol{P}}^{k+1} = \bar{\boldsymbol{P}}^{k+1} - \boldsymbol{K}^{k+1}S^{k+1}(\boldsymbol{K}^{k+1})^T \tag{28}$$

Because P is symmetric, the 2×2 matrix for this problem has only three unique terms. For Eq. (27), these are

$$\bar{P}_{11}^{k+1} = \hat{P}_{11}^k + 2\hat{P}_{12}^k T + \hat{P}_{22}^k T^2 + Q_{11} \tag{29}$$

$$\bar{P}_{12}^{k+1} = \hat{P}_{12}^k + \hat{P}_{22}^k T + Q_{12} \tag{30}$$

and

$$\bar{P}_{22}^{k+1} = \hat{P}_{22}^k + Q_{22} \tag{31}$$

The elements of Q come from Eq. (49). The innovation covariance is

$$S^{k+1} = \bar{P}_{11}^{k+1} + R^{k+1} \tag{32}$$

Equation (28) becomes

$$\hat{P}_{11}^{k+1} = \frac{\bar{P}_{11}^{k+1}R^{k+1}}{S^{k+1}} \tag{33}$$

$$\hat{P}_{12}^{k+1} = \frac{\bar{P}_{12}^{k+1}R^{k+1}}{S^{k+1}} \tag{34}$$

and

$$\hat{P}_{22}^{k+1} = \bar{P}_{22}^{k+1} - \frac{\left(\bar{P}_{12}^{k+1}\right)^2}{S^{k+1}} \tag{35}$$

Finally, the Kalman gain, Eq. (25), is

$$\boldsymbol{K}^{k+1} = \begin{bmatrix} \bar{P}_{11}^{k+1}/S^{k+1} \\ \bar{P}_{12}^{k+1}/S^{k+1} \end{bmatrix} \tag{36}$$

Coupled Kalman Filter

The coupled Kalman filter is by far the most complex of the four filters. The six-dimensional state is

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{v} \end{bmatrix} \tag{37}$$

where $\boldsymbol{r}$ and $\boldsymbol{v}$ are the position and velocity vectors of the target in ECI coordinates. The state is propagated forward in time using analytical Keplerian motion equations, which are the solution to the problem of motion under inverse-square gravity.[9] As for $\Phi(k+1, k)$, it is computed with numerical partial derivatives of the Keplerian solutions of the state:

$$\Phi(k+1,k) = \frac{\partial \boldsymbol{x}(t_{k+1})}{\partial \boldsymbol{x}(t_k)} \tag{38}$$

The process noise covariance Q is the same as Eq. (49), except that each term is multiplied by the 3×3 identity matrix. Given these, Eq. (27) is used to propagate the state covariance forward in time to the next measurement.

To effect the measurement update, the state and covariance are transformed from the ECI to the RUV space. The RUV state is defined to be

$$\boldsymbol{y} = \begin{bmatrix} r \\ u \\ v \\ \dot{r} \\ \dot{u} \\ \dot{v} \end{bmatrix} \tag{39}$$

where r is the range from the sensor to the target, u is the direction cosine of the range vector with respect to the x axis of the floating frame, and v is the direction cosine with respect to the y axis of the floating frame (see Fig. 1 for a depiction of the floating frame and RUV space). The floating frame is a Cartesian frame defined at each measurement update time; z is pointing toward the estimated position at the update time, x is in the vertical plane (generally up—that is, having a nonnegative projection onto any vector that is purely vertical at the sensor), and y is in the horizontal plane. The measurement is taken to be

$$\boldsymbol{z} = \begin{bmatrix} r \\ u \\ v \end{bmatrix} \tag{40}$$

so that

$$\boldsymbol{H} = [I \quad 0] \tag{41}$$

The measurement covariance matrix is approximated well by

$$\boldsymbol{R} = \begin{bmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_a^2 & 0 \\ 0 & 0 & \sigma_a^2 \end{bmatrix} \tag{42}$$

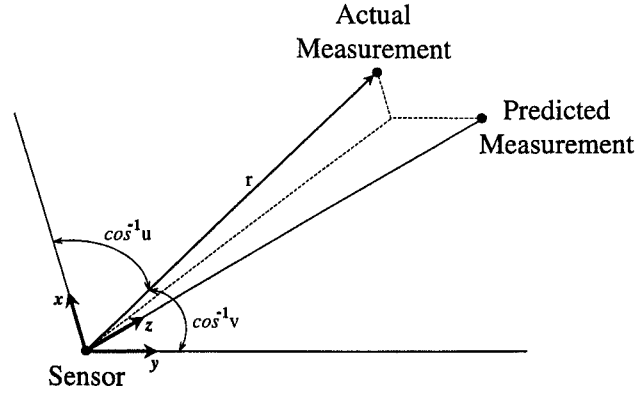

Fig. 1 Floating frame and the RUV space.

Let $g(x)$ represent the nonlinear transformation from ECI to RUV. Then at update-time index k,

$$\bar{y}_k = g(\bar{x}_k) \tag{43}$$

is the state transformation. To transform the state covariance, first the Jacobian of the transformation is analytically computed at $\bar{x}_k$:

$$J = \left.\frac{\partial g(x)}{\partial x}\right|_{\bar{x}_k} \tag{44}$$

Then, denoting the covariance of y by P_y,

$$\bar{P}_y^k = J\bar{P}^k J^T \tag{45}$$

At this point, the state and its covariance are updated in the usual manner in the RUV space. The innovation covariance, Eq. (26), and the Kalman gain, Eq. (25), are computed using R and $\bar{P}_y^k$, whence the state update and the state-covariance update are performed using Eqs. (19) and (28), respectively [with y being used in place of x in Eq. (19)].

The final step is to effect the inverse transformation from the RUV space to the ECI space:

$$\hat{x}_k = g^{-1}(\hat{y}_k) \tag{46}$$

and

$$\hat{P}^k = J^{-1}\hat{P}_y^k(J^{-1})^T \tag{47}$$

(with J having been evaluated again at $\hat{x}_k$). With this, a full measurement cycle, from propagation to measurement update, is complete.

Filter Initialization

All of the filters are initialized in essentially the same manner. The state is initialized with the position vector equal to the first measurement (or the first measurement transformed to ECI for the coupled Kalman), and the initial velocity is set to zero.

The Kalman filters need their covariances initialized also. For the decoupled Kalman,

$$P_0 = \begin{bmatrix} (\rho\sigma_a)^2 & 0 \\ 0 & L \end{bmatrix} \tag{48}$$

with L set to 5 km^2/s^2. For the coupled Kalman,

$$P_0 = \begin{bmatrix} J^{-1}R(J^{-1})^T & 0 \\ 0 & L \cdot I \end{bmatrix} \tag{49}$$

where I is the 3 × 3 identity matrix, and R and J are as defined in Eqs. (42) and (44).

Other initialization methods are possible, such as initializing the velocity with the difference between the first two measurements divided by the time between the measurements. It can be shown, however, that this yields significantly degraded performance for the first several measurements for the ballistic missile problem (because excellent upper bounds are known for ballistic missiles, which can be taken advantage of in the covariance initialization). On the other hand, it can also be shown that after the first few seconds the differences in the performance of the different initialization methods are totally negligible.

Computational Complexity

The issue of computational complexity is important to this study. The two alpha-beta filters are the least complex by far because, first of all, the computations for the gains are simple. More importantly, the alpha-beta filter gains can be computed via a table lookup, as a function of the maneuvering index. This process can be accomplished with six multiplications/divisions and one addition/subtraction. Similarly, the growing memory filter can be computed via table lookup, using the measurement number k directly as the lookup index; this portion of the filter requires one division.

The decoupled Kalman filter is also very competitive in having low complexity. It is propagated with the same equations as the alpha-beta filters, and the whole covariance process and gains calculations for one cycle have only 13 multiplications/divisions and 8 additions/subtractions. That is hardly a computational burden! In fact, it is even slightly less of a computational load than the pure alpha-beta filter if the alpha-beta filter were computed on line as opposed to via table lookup.

The coupled Kalman filter, on the other hand, is full of matrix inversions, matrix multiplications, and, for the specific variety examined here, nonlinear transformations and trigonometric evaluations. The major computations comprise 1660 multiplications/divisions and 1301 additions/subtractions. This is two orders of magnitude more complex than the decoupled Kalman. If the computations for all of the transformations resulting from the multiple coordinate system approach are removed, to get a feel for the complexity of a simpler coupled Kalman filter, the remaining typical Kalman recursion equations require 648 multiplications/divisions and 549 additions/subtractions. Further simplifications are possible, such as choosing a simplified form for the transition matrix that is essentially the same as Eq. (2) and making use of this simple form to reduce subsequent computations; with this, the computations can be reduced to 216 multiplications/divisions and 153 additions/subtractions. The bottom line is that a fully coupled Kalman filter will require at least an order of magnitude more computations than the decoupled Kalman filter for this problem; on the other hand, the decoupled Kalman is only slightly more complex than the alpha-beta filters.

Results

The target trajectory used for the numerical results is a 1000-km, minimum energy ballistic missile trajectory created with a rotating-Earth model, using oblate-Earth gravity. The ground track of this trajectory is plotted in Fig. 2, and the ground-range/altitude profile is shown in Fig. 3. The two sensor locations used are also shown, along with vectors to the first detection point on the trajectory from each of these locations.

For sensor 1, the detection range is 550 km, the angular error on the radar is 3 mrad, and the range error is 25 m. The measurement rate is one measurement every 0.25 s. Both of the Kalman filters have the process noise parameter $\epsilon = 1$ m/s^2, as does the alpha-beta

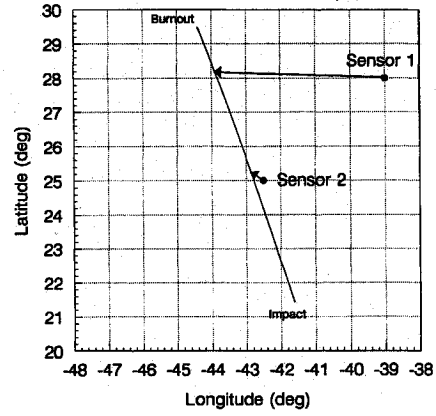

Fig. 2 Target and sensor geometries.

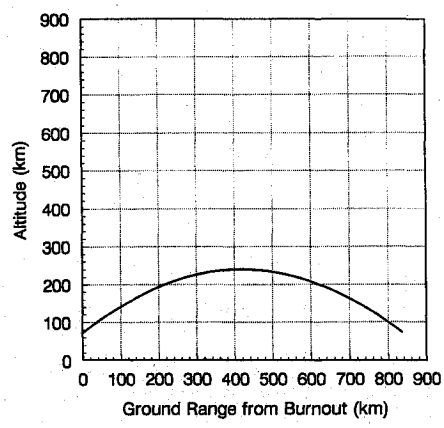

Fig. 3 Target vertical plane profile.

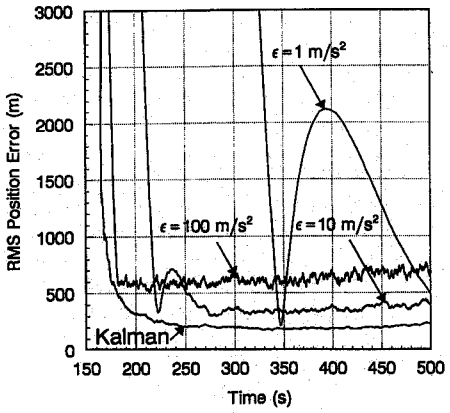

Fig. 4 Position errors for alpha-beta filter.

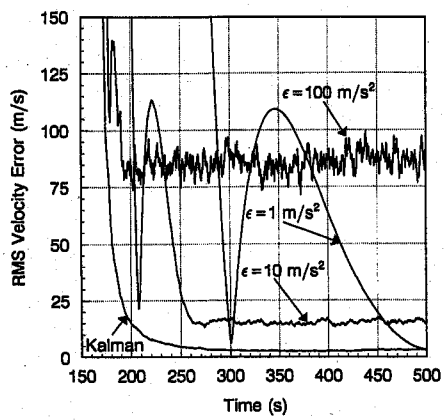

Fig. 5 Velocity errors for the alpha-beta filter.

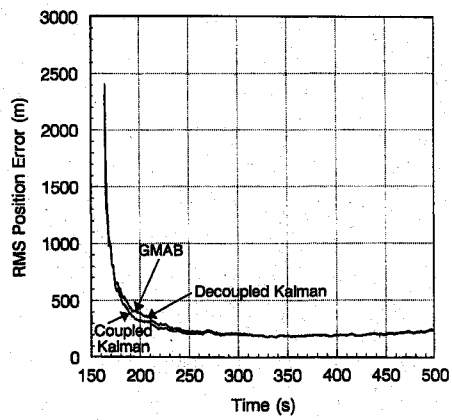

Fig. 6 Position errors for GMAB and Kalman filters at sensor position 1.

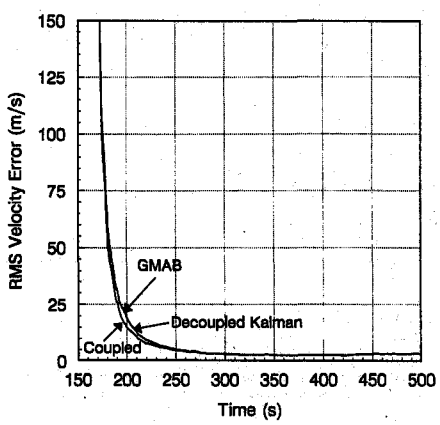

Fig. 7 Velocity errors for GMAB and Kalman filters at sensor position 1.

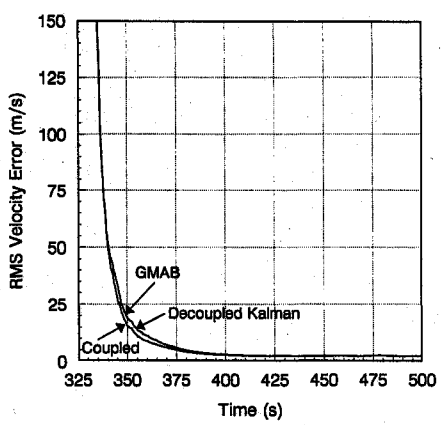

Fig. 8 Velocity errors for GMAB and Kalman filters at sensor position 2.

portion of the GMAB filter. Each of the error curves are the rms values of the true errors from 100 Monte Carlo realizations.

The first results to be presented are the position and velocity errors of the alpha-beta filter at sensor position 1, for three different levels of ϵ, compared with the coupled Kalman filter, shown in Figs. 4 and 5. For higher values of the process noise, the alpha-beta filter responds much more rapidly. That is, the errors drop down to near their steady-state value more rapidly. The $\epsilon = 100$ m/s^2 case responds almost as rapidly as the coupled Kalman. On the other hand, the lower the value of ϵ is, the lower the steady-state errors are. For example, the $\epsilon = 1$ m/s^2 case settles as low as the Kalman, but it requires nearly the whole trajectory to do so. The $\epsilon = 10$ m/s^2 case is a compromise that yields a reasonable balance between the settling rate and the steady-state error level; the resulting performance, however, considerably lags that of the Kalman filter, with a minute delay in settling time and significantly higher steady-state errors.

This example demonstrates the general trend of pure alpha-beta filters for ballistic trajectories: You can have fast settling or low settling but not both. Of course, the alpha-beta filter will perform better if varying levels of gains are chosen, starting high and ending low. This, however, is precisely what the GMAB and the Kalman filters are designed to do automatically.

Figures 6 and 7 show the results of the GMAB and the Kalman filters for the same case as just presented. All three of the filters behave nearly the same. In fact, the differences between the GMAB and the decoupled Kalman cannot be discerned on the plots. And both of these decoupled filters behave nearly the same as the fully coupled extended Kalman filter.

Next, consider a totally different geometry. At sensor position 2, the detection range is 250 km, and so the transverse linear error is smaller for the same 3-mrad angular error, and the direction is nearly vertical (because the apogee height of the trajectory is nearly 250 km), as opposed to the nearly horizontal position vector of the sensor 1 case. In other words, the geometry is quite different from the first case. The relative results between these three variable-gain filters, however, is the same, as can be seen in Fig. 8. (For the sake of brevity, only the velocity curves will be shown for most of the subsequent cases because the position and the velocity curves show basically the same relative performance.) The results are also qualitatively the same in Fig. 9, which is the sensor 1 case again except the measurement update rate is one measurement every 4 s instead of every 0.25 s. The conclusion from these cases is that the three filters behave similarly for a variety of geometries and measurement update rates.

These cases were all run with a constant measurement rate. Consider now the case when there are data loss periods. These could occur for a variety of reasons. First, the ballistic object could have a radar cross section that is changing in time due to changing angular orientation relative to the line-of-sight vector, so that the target could be fading in and out of track. In this case, the data loss periods are short enough that there is a high probability of reacquiring the target within one beam width of the predicted target position based on the estimated state from the last measurement. A logical strategy for

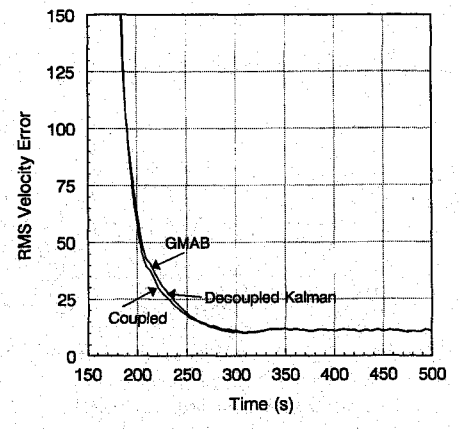

Fig. 9 Velocity errors for GMAB and Kalman filters at sensor position 1 with 4 s between measurements.

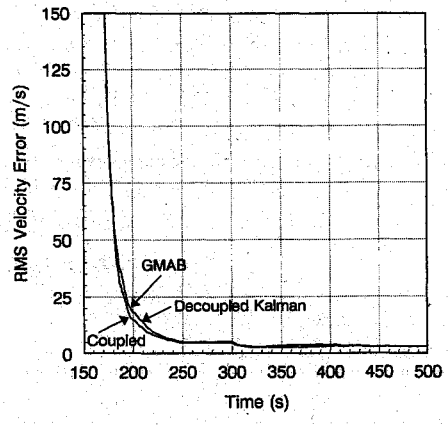

Fig. 10 Velocity errors for one data loss period after settling.

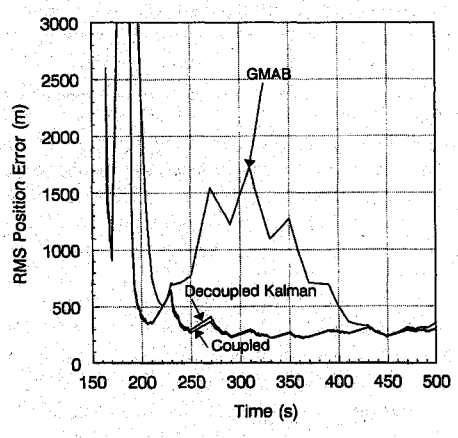

Fig. 11 Position errors for frequent data loss.

ballistic targets, because their dynamics are well known (assuming no thrusting periods that would effect orbit changes), is to continue to attempt to reacquire the target until the probability that the target is within one beam width is low.

The second class of causes for data loss would be intentional on the part of the sensor operator. For example, with a phased array radar, operational considerations could cause other targets to be of such a high priority that no looks at the target of interest would be scheduled for some period of time. Whatever the cause, data loss periods are a real consideration for real ballistic missile tracking considerations.

Figure 10 depicts what happens when one blackout period of 50-s duration occurs after the filters have settled significantly, starting at time 250 s. Notice the flat velocity curves between 250 and 300 s due to lack of measurements. Again, the filters behave very similarly.

When the target fades in and out, however, with the data losses occurring early and frequently, a different story emerges. Consider again the original sensor 1 case, except that fading in and out of data

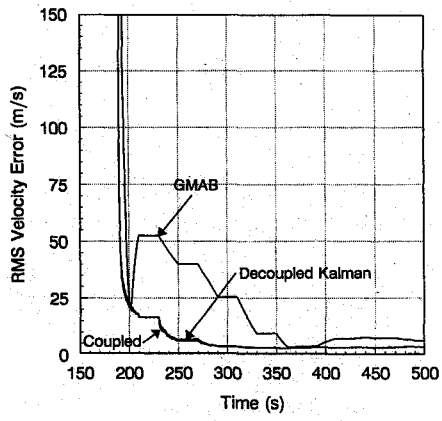

Fig. 12 Velocity errors for frequent data loss.

occurs, starting 5 s after detection for a 20-s duration, followed by subsequent periods of 20 seconds of data, then 20 seconds of loss, and so on, until the end of the trajectory. The results are depicted in Figs. 11 and 12. In this case, the GMAB filter has significantly degraded performance relative to the Kalman filters.

Conclusions

Four different filtering options have been considered for the problem of tracking an exoatmospheric ballistic target with range and angle measurements. All four filters, when properly designed and tuned, are able to track the target with some degree of accuracy. The pure, constant-gain alpha-beta filter does lag significantly the other three filters in being able to simultaneously drop the errors rapidly and to settle to a low error value. An augmented alpha-beta concept, the second filtering option considered, performs about the same as the coupled extended Kalman filter, for all typical scenarios. When data loss periods are introduced, however, its performance is degraded appreciably compared with the two Kalman filter designs.

Perhaps the most surprising of all is that the decoupled Kalman filter performs very competitively with the fully coupled extended Kalman filter for all of the different scenarios examined. Its computational complexity is even competitive with a pure alpha-beta filter (if the alpha-beta gains are generated in real time as opposed to via a table lookup), whereas it has at least an order of magnitude less complexity than the coupled Kalman filter.

Acknowledgment

This work was supported by the Navy Tactical Ballistic Missile Defense Cost and Operational Effectiveness Analysis.

References

[1]Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, Vol. 82, March 1960, pp. 34–45.

[2]Kalman, R. E., and Bucy, R., "New Results in Linear Filtering and Prediction Theory," *Journal of Basic Engineering*, Vol. 83D, March 1961, pp. 95–108.

[3]Morrison, N., *Introduction to Sequential Smoothing and Prediction*, McGraw–Hill, New York, 1969, pp. 495–602.

[4]Kalata, P. R., "The Tracking Index: A Generalized Parameter for Alpha-Beta-Gamma Target Trackers," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-20, March 1984, pp. 174–182.

[5]Bar-Shalom, Y., and Li, X.-R., "The Alpha-Beta Filter for Piecewise Constant Acceleration Model," *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, Norwood, MA, 1993, pp. 275–281.

[6]Bar-Shalom, Y., and Li, X.-R., "Explicit Filters for Noiseless Kinematic Models," *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, Norwood, MA, 1993, pp. 270, 271.

[7]Costa, P. J., and Moore, W. H., "Extended Kalman–Bucy Filters for Radar Tracking and Identification," *Proceedings of the 1991 IEEE National Radar Conference*, IEEE Aerospace and Electronic Systems Society, 1991, pp. 127–131.

[8]Bar-Shalom, Y., and Li, X.-R., "Direct Discrete Time Kinematic Models," *Estimation and Tracking: Principles, Techniques, and Software*, Artech House, Norwood, MA, 1993, pp. 266–269.

[9]Bate, R. R., Mueller, D. D., and White, J. E., "Position and Velocity as a Function of Time," *Fundamentals of Astrodynamics*, Dover, New York, 1971, pp. 177–226.