Streamlining Measurement Iteration for EKF Target Tracking

THOMAS H. KERR, Senior Member, IEEE
Lincoln Laboratory

In performing estimation of the state of a nonlinear system, a standard approach that is usually adequate is to linearize the algebraic nonlinear measurement equation (that describes the operation of the measurement sensor) and to linearize the nonlinear differential equation (that describes the system or plant), as expressed in state variable notation. This linearization that is performed as part of an extended Kalman filter (EKF) algorithmic implementation corresponds to use of the first two terms of the Taylor series expansion of the system and measurement nonlinearity as a reasonable approximation.

It is well documented in the estimation theory literature that just a minor change in mechanization beyond this standard EKF implementation can result in a significant improvement in EKF performance. Frequently, with a few further iterations (of the measurement relinearization process), the resulting intermediate linearization is greatly improved, resulting in a payoff of offering significantly improved EKF performance while the penalty incurred is merely the cost of mechanizing a variation of only slightly greater complexity or slightly higher operations counts.

This work provides the details of how to implement the measurement iteration process, described above, as a software subroutine module for an exoatmospheric random variable (RV) target tracking application. The primary contribution is in providing a new more computationally efficient general method for performing measurement iteration (or relinearization) within the implementation of an EKF. The results are illustrated, as used in a radar application of tracking exoatmospheric RV targets.

Manuscript received May 2, 1990; revised July 16, 1990.

IEEE Log No. 41885.

This work was sponsored by the Department of the Air Force under Electronic Systems Division Contract F19628-85-C-0002.

The views expressed are those of the author and do not reflect the official policy or position of the United States Government.

Author's address: Lincoln Laboratory, Massachusetts Institute of Technology, 244 Wood St., Lexington, MA 02173.

0018-9251/91/0300-0408 $1.00 © 1991 IEEE

I. INTRODUCTION

First, an overview is provided in Section IA of the definition of the estimation problem and an explanation is offered for the standard notation that is employed. Section IA then proceeds to review how the linear estimation approach of Kalman filtering is extrapolated to form an extended Kalman filter (EKF), applicable for state estimation in nonlinear systems. Next, a mechanization of an EKF variation known as an iterated EKF, offering improved tracking performance, is treated in Section IB. A streamlined version of an iterated EKF that is a lesser computation burden (having fewer operations per cycle or time-step) than prior formulations is offered in Section IIA as the primary theoretical contribution of this work. A nonlinear filtering application example, to be used as a testbed for this new approach, is described in Section IIB, where the detailed modeling considerations are discussed, as needed for radar target tracking. The performance of this new version of an iterated EKF mechanization is illustrated in Section III for this radar target tracking example and comparisons are made to the performance of an EKF without measurement iteration.

A. Essentials of EKF Mechanization

The standard linear dynamical system for which Kalman-type filters are designed has a discrete-time representation consisting of an n-dimensional state vector x_k and an m-dimensional measurement vector z_k of the following well-known form:

$$\text{System:}\quad x_{k+1} = \Phi(k+1,k)x_k + w_k \qquad (1)$$

$$\text{Measurement:}\quad z_k = H_k x_k + v_k \qquad (2)$$

with initial condition $x(0) \sim \mathcal{N}(\overline{x}(0), P(0))$ (Gaussianly distributed, with known mean $\overline{x}(0)$ and covariance matrix $P(0)$) and where $\Phi(k+1,k)$ is the known *transition matrix* and the process and measurement noises, w_k and v_k, respectively, are zero mean, white Gaussian noises (independent of the Gaussian initial condition) of known covariance intensity levels Q_k and R, respectively. The three symmetric matrices $P(0)$ and Q_k must be *positive semidefinite* and R must usually be *positive definite*. The usual conditions of observability/controllability (or less restrictive detectability/stabilizability conditions [2, p. 82]) are assumed to be satisfied here by any appropriate application system of the form of (1) and (2). The above regularity conditions being satisfied guarantee that the covariance calculations from the associated Riccati equation (to be defined below) will be well-behaved and consequently that the resulting Kalman filter (KF) will be *stable* in the sense of providing an optimal estimate $\hat{x}_k$ that quickly converges to the true current state x_k through use of (i.e., processing the information provided by) the collection of available sensor measurements z_i from $i = 0$ to $i = k$.

Equation (1) above is a discrete-time difference equation (compatible with recursive implementation on a digital computer) that corresponds to the solution of an associated underlying continuous-time state variable differential equation (describing the system) of the form:

$$\frac{d}{dt}x = F(t)x + w'(t) \tag{3}$$

where the *transition matrix* for the general time-varying case of $F(t)$ is obtained by integration of the homogenous part of (3) over the time interval of interest prior to the next measurement becoming available for use by the filter. If $F(t)$ is constant, then the appropriate transition matrix simplifies to just an evaluation of the fairly well-known matrix exponential as

$$\Phi(k+1,k) = e^{F\Delta} \tag{4}$$

where Δ is the appropriate time-step between measurements. Similarly, the appropriately exact discrete-time process noise covariance intensity level Q_k to use in the KF mechanization equations corresponding to (1) is obtained by integration of the known continuous-time process noise covariance intensity level $Q_c(t)$ associated with the continuous-time white Gaussian noise $w'(t)$ of (3) as [3, p. 171, Eq. 4-127b]:

$$Q_k = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1},\tau)Q_c(\tau)\Phi^T(t_{k+1},\tau)\,d\tau \tag{5}$$

where $\Delta = t_{k+1} - t_k$.

The standard familiar KF implementation/ mechanization equations for periodic measurements available every Δ units of time are well known, as succinctly stated in [3, p. 217, Fig. 5.4], [4, p. 111, Figs. 4.2-2 and 4.2-3].

The covariance update equation is

$$\begin{aligned} P_{k|k} &= [I - K_kH_k]P_{k|k-1} \\ &= [I - K_kH_k]P_{k|k-1}[I - K_kH_k]^T + K_kRK_k^T \end{aligned} \tag{6}$$

while the above two forms are mathematically equivalent, it is the more complex final expression (known as Joseph's form) that more effectively resists the deleterious effect of roundoff in machine computations [4, pp. 305–306], [3, p. 237] and is therefore the preferred implementation. More detail on the fundamentals of a linear KF estimator implementation and recommended steps to ease its software validation/checkout are provided in [1].

Mechanization of the optimal linear estimator as an *information filter* formulation using the inverse of the covariance matrix is denoted as the "information matrix" (an approach that is especially appealing and more tractable (i.e., being a lesser computational burden than using the more familiar standard covariance form of (6))) when the linear system of (1) has no process noise, as is the case for nonmanuvering exoatmospheric reentry vehicle (RV) targets). The complete information filter can be summarized as represented in [8, p. 206, eqs. 7.41, 7.42, 7.43, 7.44 for $Q = 0$]. A side benefit of using this particular information filter formulation is that the inverse of the *information matrix*, $\mathcal{I}(k,1)$, is identical to the corresponding Cramer-Rao lower bound for this situation when the linearizations are performed about the true position and velocity states (rather than about the filter estimates of these states, as is used in an EKF).

KF usage is by no means restricted to just situations involving periodic availability of sensor measurements since a KF can handle asynchronous measurement availability of any known time spacing of measurements or even synchronized simultaneous measurements from several different sensors at a time. However, the formulation of a KF used here and its subsequent upgrade to an EKF, as implemented in software for this investigation, was instantiated for the case of periodic measurements from a single sensor (at a time) to keep the software mechanization simple.

Mechanization of an EKF[1] can be applied to nonlinear situations [6, pp. 39–59] (such as are encountered in a more realistically detailed model of an RV trajectory). This is accomplished by linearizing (as the first two terms of a Taylor series expansion, involving a constant term and a 1st derivative term, known as the *Jacobian*) of either the system equation or measurement equation (or both if each is nonlinear) as evaluated about the current state estimate $\hat{x}(k)$ as obtained via an on-line mechanization of the KF implementation equations of [3, p. 217, Fig. 5.4], [4, p. 111, Figs. 4.2-2 and 4.2-3] (corresponding to an assumed underlying model of the form of (1) and (2)) even though the actual system under consideration is now of the more general nonlinear continuous-time system/discrete-time measurement form

$$\text{System:}\quad \frac{d}{dt}x(t) = f(x(t),t) + w(t) \tag{7}$$

$$\text{Measurement:}\quad z_k = h(x_k,k) + v_k \tag{8}$$

[1]Other approaches to handling nonlinear situations involve use of more terms in a Taylor series approximation beyond just the first two that are used in the EKF. These other approaches are called *Gaussian or second-order filters* such as the particularly nice formulation of [11] which avoids a prior unlikely assumption that the resulting estimates are Gaussianly distributed within a nonlinear filtering scenario in favor of utilizing the more plausible assumption that just the errors $\tilde{x}(t) \triangleq x(t) - \hat{x}(t)$ are Gaussian. Unfortunately some confusion has arisen since a recent textbook [10, pp. 106–109] has referred to a nonlinear filter formulation involving *Hessians* (or the second derivative of the nonlinearities) as being an EKF, while prevalent almost universal convention is that an EKF involve use of only *Jacobians* (or first derivatives of the nonlinearities).

where $f(\cdot)$ and $h(\cdot)$ are the particular nonlinearities that are actually encountered in appropriately modeling the particular application.

The explicit implementation equations for an EKF[2], as posed for a discrete-time model of the form of (7) and (8), are covered here in Table I, as summarized in [8, p. 278, p. 338] and derived in [10, Section 3.3]. Notice that this implementation differs from that of a standard KF for purely linear systems only in the propagate and update steps of the filter (viz., compare the last row of Table I here with the last row of Table I in [1]). However, another implicit difference is that $\Phi(k,k-1)$ is now a function of the measurements z_{k-1}, since it is now to be obtained as a linearization about the estimate $\hat{x}_{k-1|k-1}$, which itself is a function of the measurement z_{k-1}. Consequently, the propagate step of the covariance $P_{k|k-1}$ in Table I is implicitly a function of z_{k-1} since $\Phi(k,k-1)$ *is* and so is the EKF filter gain K_k. A slight variation on the above EKF approach (as addressed in detail in Section IB) is to iterate within the linearization step a few times to greatly improve the quality of the estimates with but a slight penalty in increased number of operations.

B. Explicit Implementation Equations of Iterated EKF

The claim is well documented in the estimation theory literature that just a minor change in mechanization beyond this standard EKF implementation can result in a significant improvement in EKF performance. The slight change involves inclusion of a few further iterations on the linearized measurement equation (as in (9) below, as initialized with the condition of (10)). Frequently, with a few further iterations (of the measurement relinearization process), the resulting intermediate linearization is greatly improved, resulting in a payoff of having significantly improved EKF performance [8, pp. 279–280, 349–351], [4, pp. 190–191], [7] while the penalty incurred is merely the cost of mechanizing a variation of only slightly greater complexity or slightly higher operations counts than that of a standard EKF.

The variation on a conventional EKF that was implemented here, called an iterated EKF, is as summarized in [8, p. 279] and derived in [10, Section 3.5]. The primary distinctive feature of an iterated EKF is that instead of the filter update step portrayed in Table I, we instead use the result of a few iterations[3] on i (say $i = 1,2,3$ for 3 iterations) of the following alternative mechanization equation

$$\eta_{i+1} = \hat{x}_{k|k-1} + K_k(z_k - h(\eta_i) - H_k\,|_{x=\eta_i}\,[\hat{x}_{k|k-1} - \eta_i]) \tag{9}$$

starting with an initial value of $\eta_1 = \hat{x}_{k|k-1}$. The final value used for the filter update at time-step t_k for the iterated EKF is

$$\hat{x}_{k|k} = \eta_3. \tag{10}$$

Other than this slight variation, everything else associated with an EKF mechanization, as depicted in Table I, is also used for an iterated EKF mechanization.

A milestone implementation of an iterated EKF approach was flawlessly enunciated by Gura in [5] and this treatment is known for its clarity but, unfortunately, without clearly indicating how it should be computationally mechanized. This current paper provides the details herein of how to implement the measurement iteration process, described above, as a software subroutine module for an exoatmospheric RV tracking application. The primary contribution of this work is in providing a new, more computationally efficient general method for performing measurement iteration (or relinearization) within the implementation of an EKF, used here for target tracking. The results are illustrated in a radar application of tracking exoatmospheric RV targets, as described in detail in Section IIB, with an evaluation of simulation results provided in Section III.

II. MECHANIZATION OF MEASUREMENT ITERATION FOR EXOATMOSPHERIC RV TRACKING

A. Detailed Specification of Measurement Iteration Mechanization for an EKF

This section offers the details of how to implement the measurement iteration described above as a software subroutine module for this application. The following are *inputs* to the measurement iteration software module: the prediction step state estimate, $\hat{x}^{(-)}(6 \times 1)$, the measurement $z(m \times 1)$ and the prediction step covariance matrix, $P^{(-)}(6 \times 6)$ (all at the current time k); the calculations constituting a *measurement relinearization* or *measurement iteration*, as depicted in the flowchart of Fig. 1, are implemented to obtain the desired *outputs*: the update step state estimate, $\hat{x}^{(+)}(6 \times 1)$, and the update step covariance, $P^{(+)}(6 \times 6)$, at the current time k.

An erroneous version of EKF measurement iteration and its mechanization was originally published in an earlier edition of [4] that appeared prior to 1982; but The Analytic Sciences Corp. (TASC) corrected its EKF measurement iteration formulation in all later editions of their *Applied Optimal Estimation*

[2]The EKF mechanization is to be distinguished from the so-designated *linearized KF* [4, Table 6.1-3, p. 189], which is also applied to the linearized versions of the system and measurement equations of (7) and (8), respectively, but for which filter gains can be precalculated, unlike the situation for an EKF.

[3]With all these intermediate iterations being performed while the current time value k is fixed (before proceeding to the next value of $k+1$).

TABLE I
Extended Kalman Filter Implementation/Mechanization Equations

	PROPAGATE STEP	UPDATE STEP
COVARIANCE	$P_{k\|k-1} = \Phi(k,k-1)P_{k-1\|k-1}\Phi^T(k,k-1) + Q_k$	$P_{k\|k} = [I - K_k H_k]P_{k\|k-1}[I - K_k H_k]^T + K_k R K_k^T$
FILTER GAIN	$K_k = P_{k\|k-1}H_k^T[H_k P_{k\|k-1}H_k^T + R]^{-1}$	
FILTER	$\hat{x}_{k\|k-1} = \hat{x}_{k-1\|k-1} + \int_{t_{k-1}}^{t_k} f(\hat{x}_{\tau\|t_{k-1}})\, d\tau$	$\hat{x}_{k\|k} = \hat{x}_{k\|k-1} + K_k(z_k - h(\hat{x}_{k\|k-1}))$

textbook (following helpful proding from R. J. Fitzgerald).[4] Because of its general clarity and ease of use, this textbook has had considerable influence on the entire estimation field since it is in widespread use. The subsequent correction utilized here is now essentially faithful to how measurement iteration was first described (alas, without instructions for actual mechanization) by the original developers [9] in its effect on performance and in the changes in the calculations; however, this current version offered here goes further to offer a sizable reduction in the required operations counts of its mechanization beyond what is indicated in [4, pp. 190–191].

Instead of just imitating what has always been done before on this important topic of measurement iteration mechanization (see Fig. 1), it was noticed in this investigation that inclusion of the calculation of the covariance update within the loop (as all the other prior investigators had evidently done) wasn't really necessary. This covariance update is the most computationally burdensome entity previously iterated in the loop, so to remove it and postpone it until after the loop represents a considerable savings in computational effort or computer time expended per a measurement sample time point. The iterative η equation (9) and calculation of the effective Kalman gain K are to still be iterated within the loop, but the covariance update need be calculated only once almost as an apparent afterthought (using the final value of the Kalman gain K^* which completely captures the essence of having performed the additional relinearizing iterations at each time point k). The covariance update can then be calculated just once per a time point k from

$$P^{(+)} = (I - K^*H)P^{(-)}(I - K^*H)^T + K^*RK^{*T}. \tag{11}$$

It is already fairly well known (as discussed following (6)) that use of the above covariance update equation is computationally superior to use of the algebraically equivalent but apparently simpler

$$P^{(+)} = (I - K^*H)P^{(-)}. \tag{12}$$

[4]Extensive prior experience in the use of linear and nonlinear Kalman filters is exhibited within the designs of RV tracking radars [12–14] such as those of PAVE PAWS, Cobra Judy, etc.

An aspect uncovered by Nishimura [31] that is now well known, and which we utilize here in validly removing the covariance calculation from within the measurement iteration loop is that given any filter gain K^* (optimal or otherwise), the expression of (11) provides the correct corresponding consequential uncertainty incurred (while (12) does *not*). An explicit quantification of the computational savings to be reaped by using the modification advocated here can be gauged using the real-time assembly language operations count assessment of [32, Tables IV and VI] (augmented here to consider (11) instead of (12) and using matrix dimensions of n and m, as defined just prior to (1)) as depicted in Table II. By sidestepping the covariance calculation in the first two iterations of the measurement iteration loop as we recommend on the right hand flowchart in Fig. 1, the computational burden of having to perform the processing calculations associated with the operations counts tallied in the first row of Table II is reduced by 66.6 percent (rather than having to be performed two times more than is really necessary). This is approximately a factor of four speed up that is more significant the larger n is and consequently as n^3 is, as the actual computational burden being avoided. We later found that Maybeck [6, pp. 58–59] appears to endorse use of the *new* path that we are taking here in implementing measurement iteration according to the right-hand flowchart of Fig. 1, but Maybeck doesn't mention the conclusion arrived at here concerning the computational savings accrued by not strictly following what the measurement relinearization pioneers have advocated for so long.

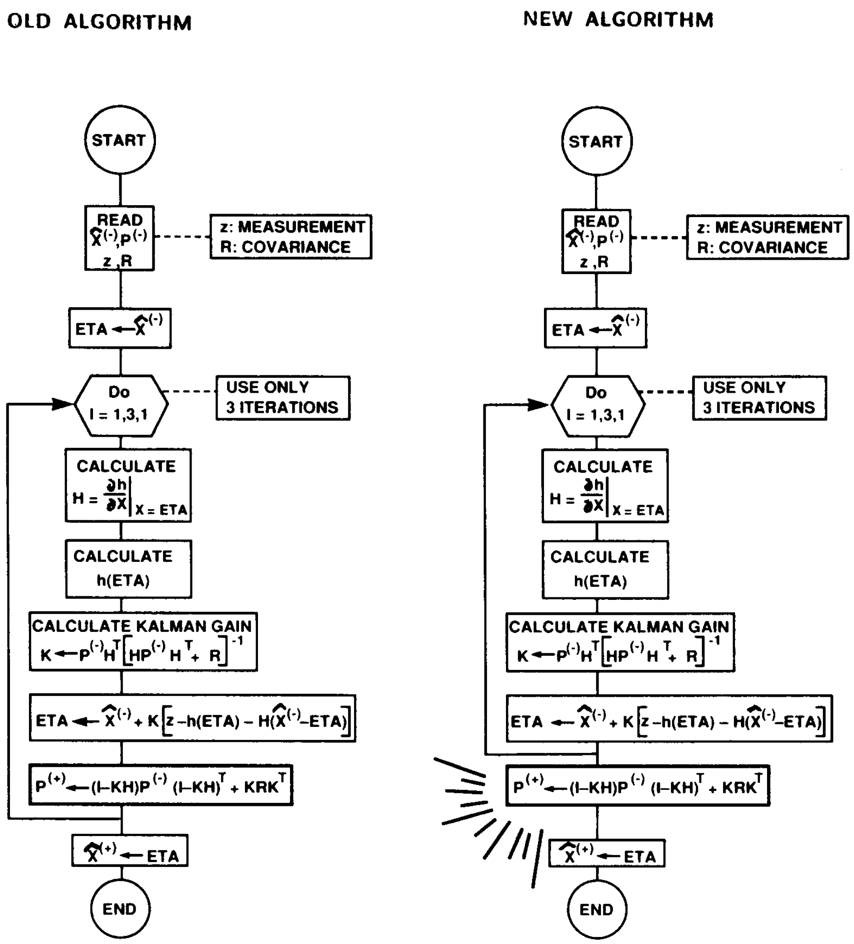
An additional wrinkle in the mechanization which relinearization can take but which we refrain from pursuing at this time is "iterative relinearization of the *entire system*" of both measurement and system equations. This more encompassing topic (and larger computational burden), as well as the specific closely related topic of measurement iteration, is discussed further in [5].

B. Earth-Centered Inertial System and Measurement Models Used as Preferred RV Target Motion Model

Reference [26] offers a good discussion of how to handle RV (or satellite) target models (a topic that, unfortunately, is usually absent from other

TABLE II
Operations Counts for Measurement Iteration Option for Standard EKF Implementation/Mechanization Equations

	MULTIPLICATIONS	ADDITIONS
COVARIANCE UPDATE EQ. 11	$2n^3 + 2n^2m + nm$	$2n^3 + 2n^2m - 2n^2$
FILTER GAIN	$n^2m + 2nm^2 + m^3$	$n^2m + 2nm^2 - 2nm + m^3$
FILTER UPDATE EQ. 10	$2nm$	$2nm$



NEW ALGORITHM AVOIDS SIGNIFICANT COMPUTATIONAL BURDEN OF UNNECESSARY COVARIANCE CALCULATION IN THE LOOP

ADDITIONAL OPTIONS: RELINEARIZE ABOUT THE SMOOTHED ESTIMATE AND/OR RELINEARIZE A NONLINEAR SYSTEM MODEL AT EACH TIME STEP

Fig. 1. Flowchart showing how implementation of measurement relinearization reduces computational burden below that of conventional approaches.

discussions concerned with the same type of RV tracking application) and provides a derivation of the particulars from first principles as well as providing an accounting and motivation for use of the various necessary coordinate systems. Other important analytic modeling considerations underlying a rigorous analysis are treated in [29]. We follow a similar path in the choice of mathematical models used here.

In our investigation, a Keplerian trajectory is introduced within a detailed simulation of the exoatmospheric target motion to include the effect of an inverse square pull of gravity and use is made of a more sophisticated filter model than had been used in an earlier phase of this same investigation to now handle tracking in the presence of these inverse square nonlinearities. Use of this more exacting methodology to represent gravity more realistically requires that we depart from just the use of simplified covariance analysis (essentially corresponding to evaluation of a Cramer-Rao lower bound for the estimation objective in the exoatmospheric regime of no process noise being present, as used in earlier investigations [23, 30]) and instead now requires that we incorporate full nonlinear filtering techniques (and the associated standard approximations). The prior straight line constant velocity target model previously used in generating convenient RV trajectories was extricated from the simulation in favor of using this more sophisticated and realistic nonlinear target trajectory involving use of an inverse square gravity term (discussed above) and the resulting RV trajectories will no longer be expected to be straight lines or have constant velocity. Instead of linearizing about the true target, as done in prior simplified covariance analysis, the EKF linearizes about the filter state estimates at each time-step (with perhaps an iteration or two, between the prediction step and measurement incorporation as the update step, as the standard technique treated here in Sections IB and IIA).

The RV target system model used in the earlier phase of this investigation (as we bootstrapped ourselves up in the software development to incrementally include more realism) was initially nonlinear but was linearized and streamlined (in the manner exhibited in detail in the derivation in [16, pp. 36–39] to be a convenient simple approximation. The resulting simple system equation, as represented in continuous-time, was

$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ 0 \\ 0 \\ -g \end{bmatrix} \tag{13}$$

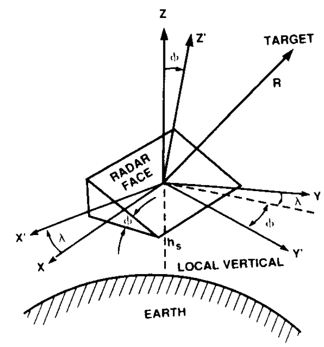


Fig. 2. ECI coordinates using this formulation (see [26]).

with an appropriate value for the acceleration of gravity g used in the above simplified model, specified to match up the target velocity of traverse to have a speed of 7 km/s and with similar motivation for the choice of initial conditions of position and velocity that were used in the earlier phase of this investigation. The coordinate convention used as a reference in (13) is with respect to the convenient inertial coordinate system known as Earth-centered inertial (ECI, having an origin that is at the center of the Earth and which is motionless with respect to the "fixed stars" that are astronomically far away), since the equations of motion simplify for this ECI coordinate frame in the sense that no additional Coriolis cross-product terms normally associated with a moving coordinate frame ([25, pp. 89–93]) need be considered.[5]

Instead of using the simplified linear model of (13) (corresponding to a local flat Earth approximation, as invoked in an earlier phase of this investigation), we now seek to work with the full nonlinear 6-state system

[5]In [7], the RV tracking problem is decomposed into primary and secondary contributing effects to be considered in the modeling, and the effect of a rotating Earth on the overall problem is of the later category. Our goal here of investigating filter tracking performance in an exoatmospheric regime initially dispenses with use of the rotation of the Earth (and as a consequence ECI is identical to an Earth-centered moving (ECM) frame, with the Earth rotation rate intentionally taken here to be zero for convenience and as a planned software validation benchmark). In the ~ 30 min of an ICBM/SLBM trajectory evolution, the Earth rotation doesn't alter the accuracy in EKF tracking performance. As in simulations for navigation applications [3, ch. 6] where similar concerns about the effect of ECM versus ECI arise in missile launch investigations, additional realism is introduced in a controlled quantized manner, where software implementations are demonstrated to work first for a mathematical model devoid of Earth rotation; then in a later phase the same software is shown to produce identical performance/output for ECM transformations that are introduced (but with the rotation parameter zeroed) as a logical step in the validation; and, finally, as the last step (not shown here) the rotation parameter of $360°/24$ h is introduced as part of this standard bootstrapping software validation approach to increased complexity and realism.

model, which in continuous-time is of the form:

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ \dfrac{-\mu x_1}{\left(\sqrt{x_1^2+x_2^2+x_3^2}\right)^3} \\ \dfrac{-\mu x_2}{\left(\sqrt{x_1^2+x_2^2+x_3^2}\right)^3} \\ \dfrac{-\mu x_3}{\left(\sqrt{x_1^2+x_2^2+x_3^2}\right)^3} \end{bmatrix} \triangleq f(x) \qquad (14)$$

where μ is the familiar gravitational constant Earth mass product GM. This is one of the equations that is linearized in implementing an EKF and for which a Jacobian for the nonlinearity on the right hand side of (14) must be calculated.

Explicit evaluation of the requisite Jacobian, obtained by performing the indicated differentiations on the system nonlinearity $f(x)$, yields

$$A(x) = \frac{\partial f(x)}{\partial x} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ a_{41} & a_{42} & a_{43} & 0 & 0 & 0 \\ a_{51} & a_{52} & a_{53} & 0 & 0 & 0 \\ a_{61} & a_{62} & a_{63} & 0 & 0 & 0 \end{bmatrix} \qquad (15)$$

where

$$a_{41} = \frac{\mu[2x_1^2 - x_2^2 - x_3^2]}{(x_1^2+x_2^2+x_3^2)^{5/2}} \qquad (16)$$

$$a_{42} = \frac{+3\mu x_1 x_2}{(x_1^2+x_2^2+x_3^2)^{5/2}} \qquad (17)$$

$$a_{43} = \frac{+3\mu x_1 x_3}{(x_1^2+x_2^2+x_3^2)^{5/2}} \qquad (18)$$

$$a_{51} = \frac{+3\mu x_2 x_1}{(x_1^2+x_2^2+x_3^2)^{5/2}} \qquad (19)$$

$$a_{52} = \frac{\mu[2x_2^2 - x_1^2 - x_3^2]}{(x_1^2+x_2^2+x_3^2)^{5/2}} \qquad (20)$$

$$a_{53} = \frac{+3\mu x_2 x_3}{(x_1^2+x_2^2+x_3^2)^{5/2}} \qquad (21)$$

$$a_{61} = \frac{+3\mu x_3 x_1}{(x_1^2+x_2^2+x_3^2)^{5/2}} \qquad (22)$$

$$a_{62} = \frac{+3\mu x_3 x_2}{(x_1^2+x_2^2+x_3^2)^{5/2}} \qquad (23)$$

$$a_{63} = \frac{\mu[2x_3^2 - x_1^2 - x_2^2]}{(x_1^2+x_2^2+x_3^2)^{5/2}} \qquad (24)$$

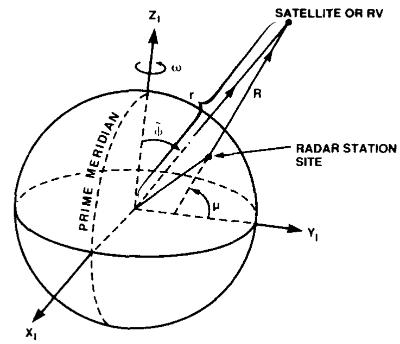


Fig. 3. Sensor measurements must now be referenced to ECI coordinates used in this formulation (see [26]).

In order to use the above ECI coordinate frame and system equations, the measurement equations are of a form addressed below. The measurement equations used for the present sensor model are as obtained from Figs. 2 and 3. The resulting sensor measurements in terms of range R, and the direction cosines u and v, to the target RV are

$$R = \sqrt{x'^2 + y'^2 + z'^2} \qquad (25)$$

$$u = \frac{x'}{R} \qquad (26)$$

$$v = \frac{y'}{R} \qquad (27)$$

where x', y', and z' are as in Fig. 3 (and are to be defined next).

In Fig. 3, the local coordinates x, y, z are located at the center of the sensor face in the plane of the array. In this coordinate system, z is directed along the local vertical and x and y lie in the horizontal plane, with x pointing East and y pointing North. From [26, sect. II], these local level coordinates x, y, z can be reexpressed in terms of x', y', z' coordinates, via the following transformation

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = T \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

where

$$T = \begin{bmatrix} \cos\lambda & -\sin\lambda & 0 \\ \cos\phi\sin\lambda & \cos\phi\cos\lambda & -\sin\phi \\ \sin\phi\sin\lambda & \sin\phi\cos\lambda & \cos\phi \end{bmatrix}$$

as the appropriate change of coordinates corresponding to the rotation depicted in Fig. 3, where the above parameters of λ and ϕ are also defined in Fig. 3. The coordinates x', y', z' are oriented so that z' is normal to the face of the sensor array, and y' lies on the face of the array, and x lies along the intersection of the sensor face and the horizontal plane.[6]

The above received sensor signal-processed measurement can be reexpressed in terms of the measurement of target range (as appropriate for a radar or other active sensor if not range-denied due to jamming), elevation, and azimuth as, respectively,

$$r = \sqrt{x^2 + y^2 + z^2} \tag{28}$$

$$E = \arctan\left[\frac{z}{\sqrt{x^2 + y^2}}\right] \tag{29}$$

$$A = \arctan\left[\frac{x}{y}\right] \tag{30}$$

where the length in (28) is identical to the length in (25) since the transformation T is a rotation (and as such is an orthogonal transformation which preserves lengths). The expressions of (28)–(30) correspond to the following measurement equation:

$$z(t) = \begin{bmatrix} r \\ E \\ A \end{bmatrix} + v(t) = \begin{bmatrix} \sqrt{x^2 + y^2 + z^2} \\ \arctan\left[\dfrac{z}{\sqrt{x^2 + y^2}}\right] \\ \arctan\left[\dfrac{x}{y}\right] \end{bmatrix} + v(t) \tag{31}$$

where the Gaussian white measurement noise $v(t)$ has a covariance that is of the form

$$R = \begin{bmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_E^2 & \\ 0 & 0 & \dfrac{\sigma_E^2}{\cos^2(E)} \end{bmatrix} \tag{32}$$

and the proper generic values to use for these variances (as squares of the following standard deviations) are found from [17, Table 1] (for radars comparable to Cobra Dane) to be

$$\sigma_r = 15 \text{ ft (per pulse)} \tag{33}$$

$$\sigma_{\text{angle}} = 0.05 \text{ deg (per pulse)}. \tag{34}$$

(If characteristics comparable to generic PAVE PAWS radars are desired instead, the requisite parameters

$$\frac{\partial r}{\partial x_o} = a_{11}(t) = \frac{x}{r}\ ; \quad \frac{\partial r}{\partial \dot{x}_o} = ta_{11}(t) = a_{14}(t)$$

$$\frac{\partial r}{\partial y_o} = a_{12}(t) = \frac{y}{r}\ ; \quad \frac{\partial r}{\partial \dot{y}_o} = ta_{12}(t) = a_{15}(t)$$

$$\frac{\partial r}{\partial z_o} = a_{13}(t) = \frac{z}{r}\ ; \quad \frac{\partial r}{\partial \dot{z}_o} = ta_{13}(t) = a_{16}(t)$$

$$\frac{\partial E}{\partial x_o} = a_{21}(t) = -\frac{xz}{\rho r^2}\ ; \quad \frac{\partial E}{\partial \dot{x}_o} = ta_{21}(t) = a_{24}(t)$$

$$\frac{\partial E}{\partial y_o} = a_{22}(t) = -\frac{yz}{\rho r^2}\ ; \quad \frac{\partial E}{\partial \dot{y}_o} = ta_{22}(t) = a_{25}(t)$$

$$\frac{\partial E}{\partial z_o} = a_{23}(t) = \frac{\rho}{r^2}\ ; \quad \frac{\partial E}{\partial \dot{z}_o} = ta_{23}(t) = a_{26}(t)$$

$$\frac{\partial A}{\partial x_o} = a_{31}(t) = \frac{y}{\rho^2}\ ; \quad \frac{\partial A}{\partial \dot{x}_o} = ta_{31}(t) = a_{34}(t)$$

$$\frac{\partial A}{\partial y_o} = a_{32}(t) = -\frac{x}{\rho^2}\ ; \quad \frac{\partial A}{\partial \dot{y}_o} = ta_{32}(t) = a_{35}(t)$$

$$\frac{\partial A}{\partial z_o} = a_{33}(t) = 0\ ; \quad \frac{\partial A}{\partial \dot{z}_o} = ta_{33}(t) = a_{36}(t)$$

$$\rho = \sqrt{x^2 + y^2}$$

Fig. 4. Linearization of measurement equation in local coordinates (see [29, pp. 22, 23]).

are in [22].) An additional aspect not to overlook is that target location is referred back to ECI coordinates within the software by subtracting out the known location of the stationary radar array. Notice that for the above described RV target model of (14) and (31), respectively, both the system model *and* the measurement model are nonlinear. The linearization of the above nonlinear measurement of (31) is as provided in Fig. 4 (from [29, pp. 22, 23]). In the earlier phase of this investigation that only performed covariance analysis and which utilized (13) as a simplified system model, evaluation of an effective linearized observation matrix H was about known true states, as obtained from a system simulation with zero process noise present. The linearization of the current EKF is about the most recent state estimate $\hat{x}_{k|k}$ instead of the actual state (which is realistically presumed to be unknown to the observing sensor). Other transformations to easily get between local geodetic and Earth-centered coordinates are as in [18–20] (where operations counts and efficiencies of implementation are explored).

A clever reformulation of how the radar range, azimuth, and elevation (RAE) measurements can be utilized within the nonlinear filtering application of exoatmospheric RV target tracking is offered in [27]. A more recent perspective is offered in [28]. The measurement model in [27] is identical to what is used

[6]The mathematics of this transformation is consistent with Fig. 3. R. M. Miller's software implementation code for getting between sensor face-centered coordinates to Earth-centered inertial coordinates avoids sinusoids within the transformation by resorting instead to the underlying right triangles corresponding to each angle measurement. R. M. Miller's alternative implementation appears to offer some nice efficiencies so we also employed it in our investigation. A standard approach for transforming between local level and ECI coordinates is provided in [21].

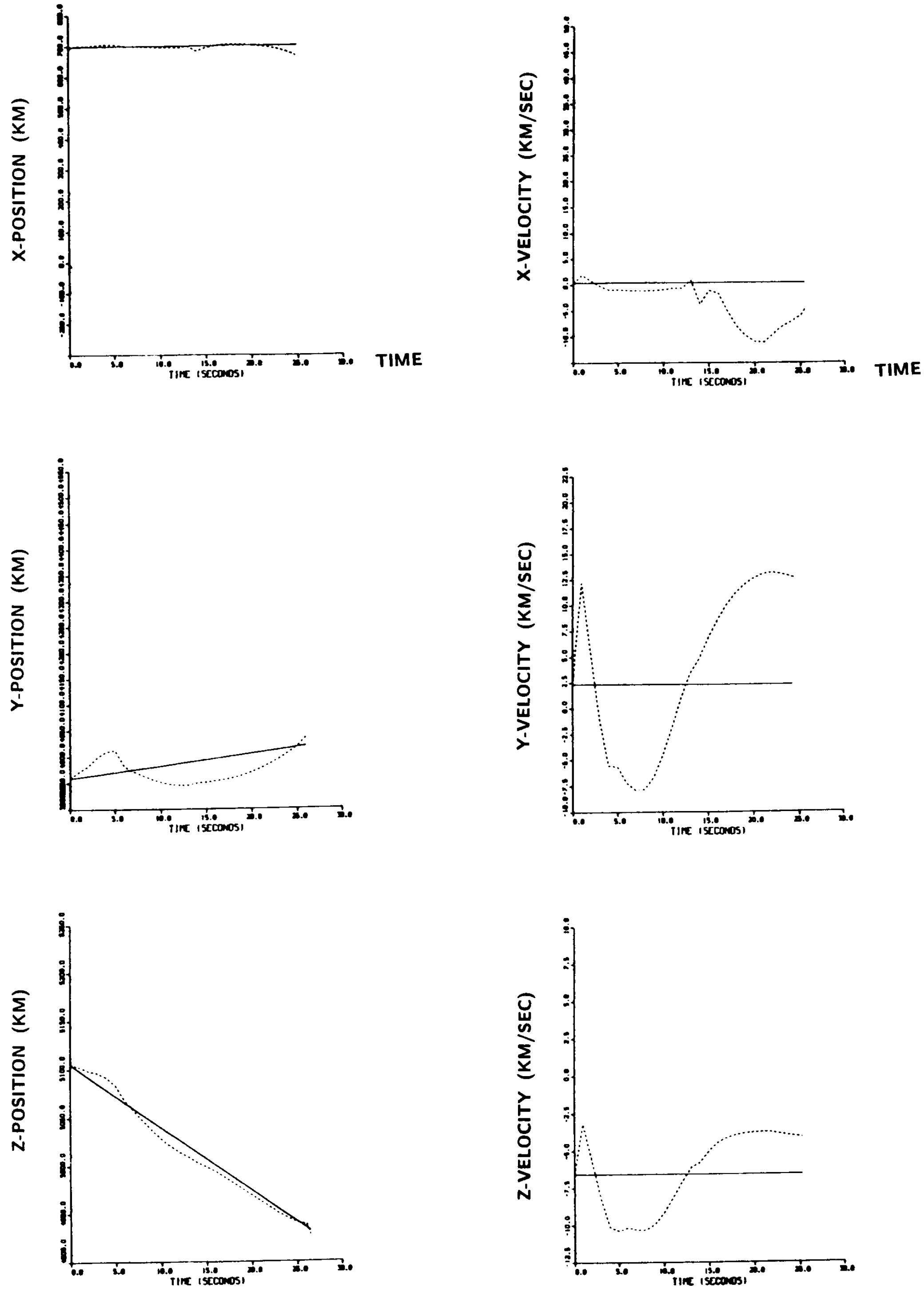


Fig. 5. Radar tracking of target RV position and velocity for linear trajectory with constant velocity.

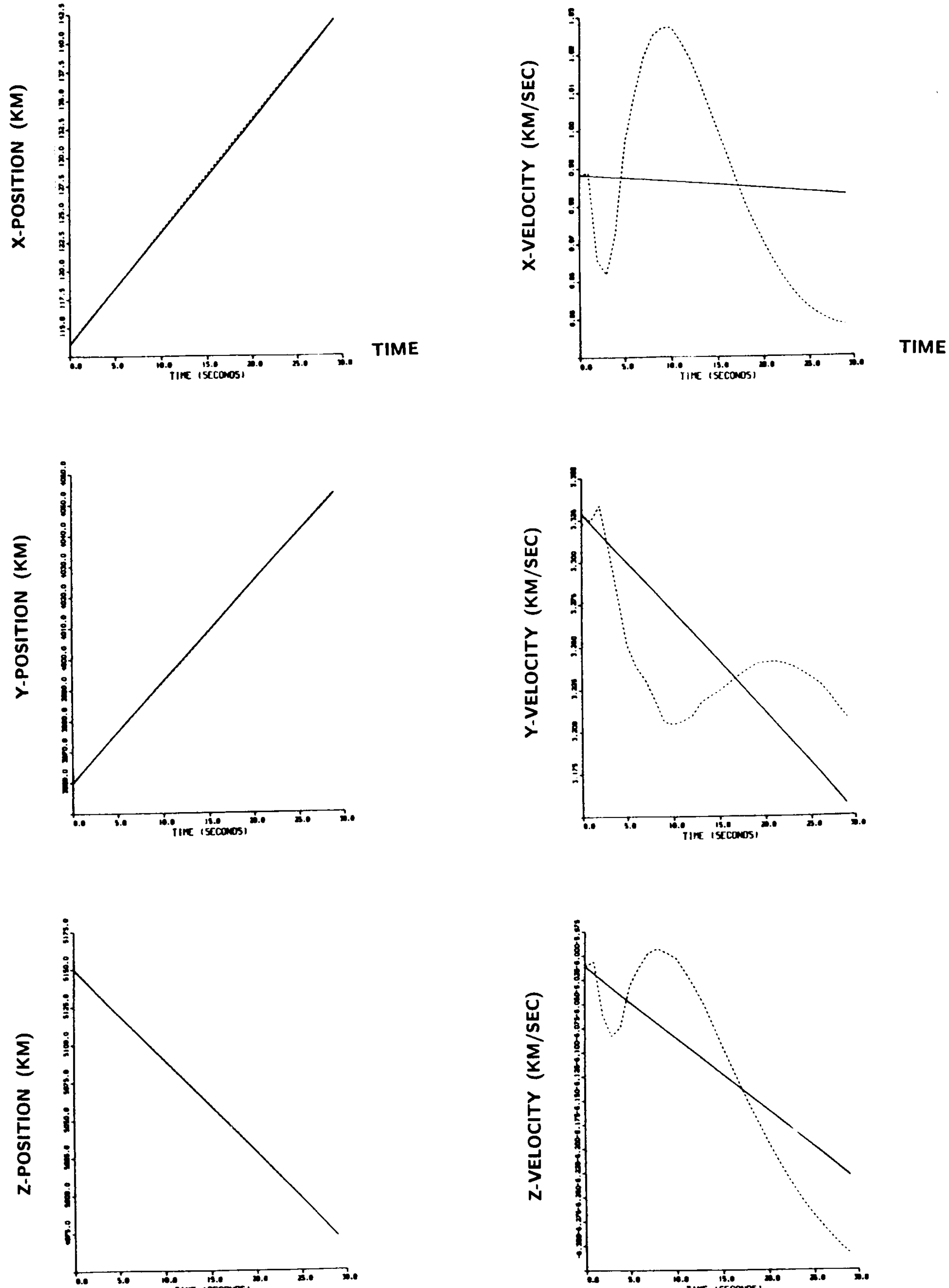


Fig. 6. Radar tracking of target RV position and velocity for nonlinear trajectory due to inverse-squared gravity model in both simulator and EKF.

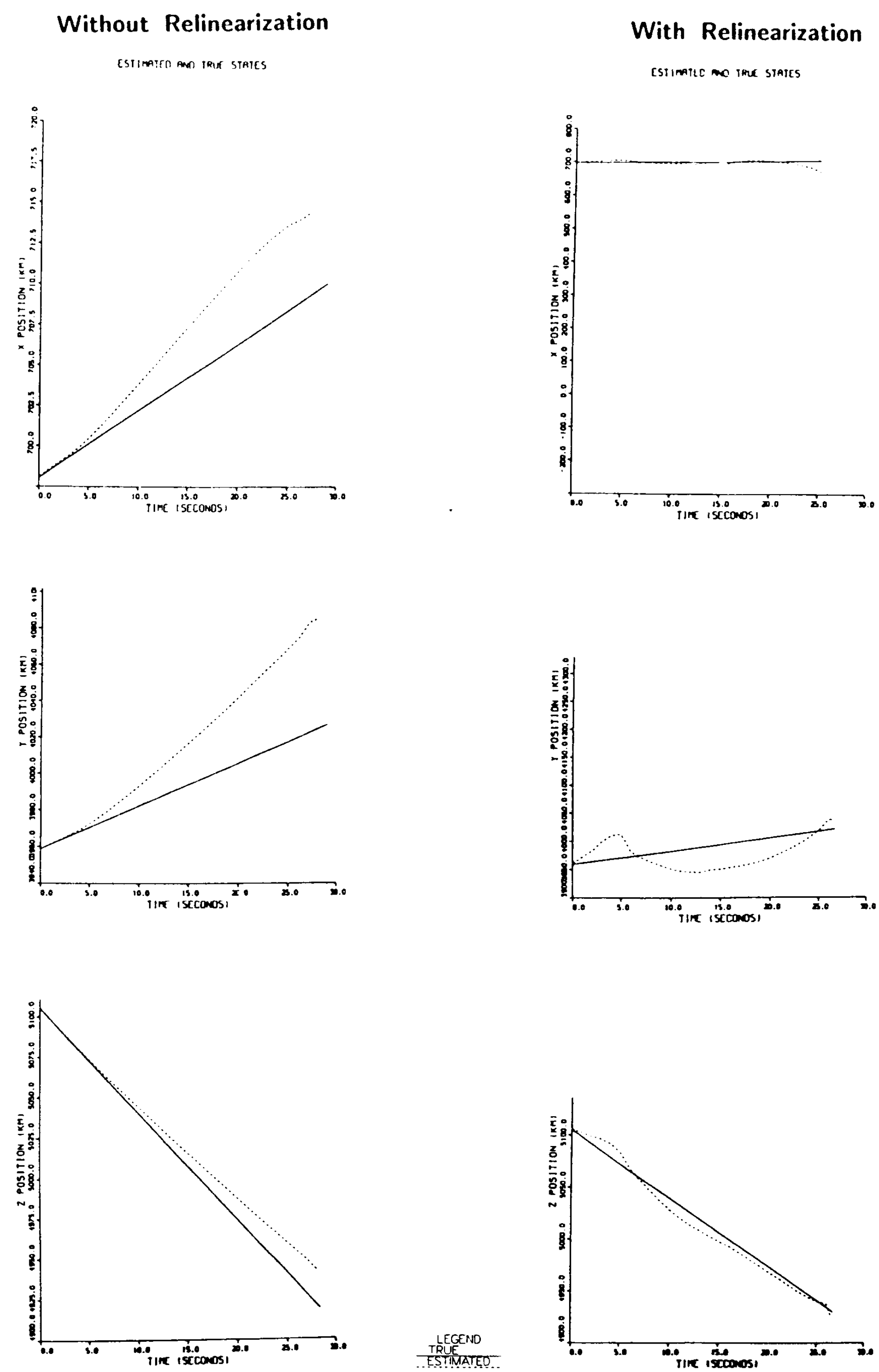


Fig. 7. Radar tracking of target RV position before/after measurement iteration.

in this investigation. However, this approach of [27] does not involve use of measurement iteration as is our thrust here.

III. USE OF MEASUREMENT ITERATION YIELDS BETTER RADAR TARGET TRACKING

The simulations of the radar case (using published Cobra Dane measurement covariances for generic range and angle[7] being $(15\ \text{ft.})^2$ and $(0.05°)^2$, respectively) appear to be performing properly as depicted in Fig. 5 (corresponding to prior use of the system model of (13) for simulating the RV trajectory) for the case of a linear (point) target and in Fig. 6 for the case of a nonlinear target (corresponding to use of the system model of (14) for simulating the trajectory, but linearized within the EKF) while both situations utilized the nonlinear measurement model of (31)). The true trajectory is depicted here as a solid continuous curve while the estimated trajectory is shown as a dashed curve.[8] Although, the estimates of velocity in Figs. 5 appear to vacillate about the true velocity states having a known total speed of 7 km/s, please notice from the coordinate scales that the estimates in Fig. 6 do in fact closely match the true velocity states within 10 percent, as is a goal of good estimator tracking performance. The better velocity tracking depicted in Fig. 6 is apparently due to the greater observability afforded by the system dynamics of (14), with inverse square gravity used rather than a constant for gravity, thus allowing better calibration/tracking of acceleration, and hence its consequence as better tracking of target velocity. The results depicted in Fig. 5 were obtained after using measurement iteration or relinearization, as portrayed both before and after invoking its use in Fig. 7. It is reassuring to see use of measurement relinearization improve EKF tracking performance in the manner that was claimed by KF specialists twenty years ago, despite encountering recent contradictory "conventional wisdom" (it is apparently commonly believed by several KF experts, recently polled, that no real improvement is offered through the use of measurement iteration since they claimed "to have never seen *any* improvements in EKF performance with the use of measurement iteration in their many years of experience"; a situation that is now understandable in light of the pervasiveness of the technical typo/faux pas introduced in the 1970s within this measurement iteration topic, as exposed and corrected in Section IIA). Measurement iteration brings expected improvements in tracking performance for the case of a linear target trajectory but apparently degrades performance in the case of the nonlinear target trajectory.[9]

[7]Expressed within the computer program in terms of kilometers and radians, respectively.

[8]Notice that the results for the last 30 s or so for the nonlinear case are strikingly similar to straight lines thus justifying the simplifying assumption previously invoked in the earlier phase of this investigation. However, the RV velocities obtained in using this simplification were constant, while the velocities obtained in using the nonlinear model change in such a way that the speed increases as the impact point is approached (since no drag or damping is currently modeled in this software code that was developed primarily for investigating the behavior of radar tracking RVs exoatmospherically and uses this impact segment of the trajectory as merely a convenient cross-check point). As with all the simulation runs depicted here, for convenience we cheat slightly and start the Kalman filter estimate off with the exact true value at the time of turn on or acquisition (time = 0 in the Figs. 5–7, but only at this start time). This would not be done in practice because it would give better tracking results than typically expected. However, if tracking results were lousy despite this better than average ideal initial estimate, then indications would be that something is drastically wrong, *but which we, thankfully, did not encounter for this software mechanization.* In the case of both system and measurement models being perfectly linear and observability and controllability conditions holding, the ultimate long term performance of the Kalman filter estimator in performing tracking is independent of bad initial condition guesses since their effect dies out with the passage of time. However, for nonlinear models, the effect of bad initial condition guesses is long lived and if initially off by too much can trigger filter divergence or large scale departure of the estimates from the true state that just gets worse as time elapses without any ameliorating counter actions being available.

[9]This may be due to the presence of both a nonlinear system model as well as a nonlinear measurement model in the case of a nonlinear RV trajectory; while the case of a linear RV trajectory only had a nonlinear measurement model to contend with, which is more amenable to EKF performance improvement via mere relinearization of the measurement equation. As mentioned just prior to Section IIB there is a technique, documented by Gura [5], whereby both the measurement *and* system model are relinearized, but it is a larger computational burden than the version that we already have implemented in software here. Thanks are expressed to the programers Judith Webb and Margaret Kozubal.

REFERENCES

[1] Kerr, T. H. (1990)
Use of idempotent matrices to validate linear systems software.
IEEE Transactions on Aerospace and Electronic Systems, **AES-26**, 6 (Nov. 1990), 935–952.

[2] Kwakernaak, H., and Sivan, R. (1972)
Linear Optimal Control Systems.
New York: Wiley-Interscience, 1972.

[3] Maybeck, P. S. (1979)
Stochastic Models, Estimation, and Control, Vol. 1.
New York: Academic Press, 1979.

[4] Gelb, A. (Ed.) (1974)
Applied Optimal Estimation.
Cambridge, MA: M.I.T. Press, 1974.

[5] Gura, I. A. (1968)
Extension of linear estimation techniques to nonlinear problems.
The Journal of Astronomical Sciences, **XV**, 4 (July/Aug. 1968), 194–205.

[6] Maybeck, P. S. (1982)
Stochastic Models, Estimation, and Control, Vol. 2.
New York: Academic Press, 1982.

[7] Wishner, R. P., Larson, R. E., and Athans, M. (1970)
Status of radar tracking algorithms.
In *Proceedings of Symposium on Nonlinear Estimation Theory and Its Applications*, San Diego, CA, (1970), 32–54.

[8] Jazwinski, A. H. (1970)
Stochastic Processes and Filtering Theory.
New York: Academic Press, 1970.

[9] Denham, W. F., and Pines, S. (1966)
Sequential estimation when measurement function nonlinearity is comparable to measurement error.
AIAA Journal, **4** (1966), 1971–1076.

[10] Bar-Shalom, Y., and Fortmann, T. E. (1988)
Tracking and Data Association.
New York: Academic Press, 1988.

[11] Liang, D. F., and Christensen, G. S. (1975)
Exact and approximate state estimation for nonlinear dynamic systems.
Automatica, **11** (1975), 603–612.

[12] Fitzgerald, R. J. (1974)
Effects of range-Doppler coupling on chirp radar tracking.
IEEE Transactions on Aerospace and Electronic Systems, **AES-10**, 4 (July 1974), 528–532.

[13] Fitzgerald, R. J. (1974)
On reentry vehicle tracking in various coordinate systems.
IEEE Transactions on Automatic Control, **AC-19**, 5 (July 1974), 581–582.

[14] Daum, F. E., and Fitzgerald, R. J. (1983)
Decoupled Kalman filters for phased array radar tracking.
IEEE Transactions on Automatic Control, **AC-28**, 3 (Mar. 1983), 269–283.

[15] Miller, R. W., and Chang, C. B. (1977)
Some analytical methods for tracking, prediction, and interception performance.
Project report RMP-129, Lincoln Laboratory, M.I.T., Lexington, MA, Aug. 9, 1977.

[16] Libelt, P. B. (1967)
An Introduction to Optimal Estimation.
Reading, MA: Addison-Wesley, 1967.

[17] Filer, E., and Hartt, J. (1976)
Cobra Dane wideband pulse compression system.
In *Proceedings of IEEE EASCON*, Washington, DC, Sept. 1976.

[18] Lupash, L. O. (1985)
A new algorithm for the computation of the geodetic coordinates as a function of Earth-centered Earth-fixed coordinates.
AIAA Journal of Guidance, Control, and Dynamics, **8**, 6 (Nov.–Dec. 1985), 787–789.

[19] Churchyard, J. N. (1986)
Comments on "A new algorithm for the computation of the geodetic coordinates as a function of Earth-centered Earth-fixed coordinates."
AIAA Journal of Guidance, Control, and Dynamics, **9**, 4 (July–Aug. 1986), 511.

[20] Nautiyal, A. (1988)
Algorithm to generate geodetic coordinates from Earth-centered Earth-fixed coordinates.
AIAA Journal of Guidance, Control, and Dynamics, **11**, 3 (May–June 1988), 281–283.

[21] Nash, R. A., Jr., Levine, S. A., and Roy, K. J. (1971)
Error analysis of space-stable inertial navigation systems.
IEEE Transactions on Aerospace and Electronic Systems, **AES-7**, 4 (July 1971), 617–629.

[22] Hoft, D. J. (1976)
Devices and techniques for all solid state radars.
In *Proceedings of EASCON*, Washington, DC, Nov. 26–29, 1976, 80A–80G.

[23] Miller, R. W., and Chang, C. B. (1978)
A modified Cramer–Rao bound and its applications.
IEEE Transactions on Information Theory, **IT-24**, 3 (May 1978), 398–400.

[24] Chang, C. B. (1980)
Ballistic trajectory estimation with angle-only measurements.
IEEE Transactions on Automatic Control, **AC-25**, 3 (June 1980), 474–480.

[25] Bate, R. R., Mueller, D. D., and White, J. E. (1971)
Fundamentals of Astrodynamics.
New York: Dover, 1971.

[26] Mehra, R. K. (1971)
A comparison of several nonlinear filters for reentry vehicle tracking.
IEEE Transactions on Automatic Control, **AC-16**, 4 (Aug. 1971), 307–319.

[27] Miller, K. S., and Leskiw, D. M. (1982)
Nonlinear estimation with radar observations.
IEEE Transactions on Aerospace and Electronic Systems, **AES-18**, 2 (Mar. 1982), 192–200.

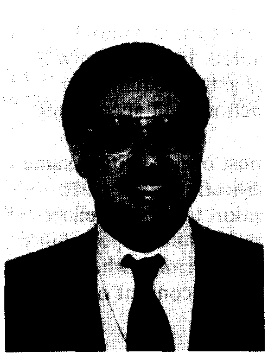
[28] Miller, K. S., and Leskiw, D. M. (1987)
An Introduction to Kalman Filtering with Applications.
Malabar, FL: Robert E. Krieger Publishing Co., 1987.

[29] Miller, R. W., and Chang, C.-B. (1977)
Some analytical methods for tracking, prediction, and interception performance.
Project report RMP-129, Lincoln Laboratory, M.I.T., Lexington, MA, Aug. 9, 1977.

[30] Miller, R. M. (1978)
A lower bound on angle-only tracking accuracy.
Project report RMP-149, Lincoln Laboratory, M.I.T., May 9, 1978.

[31] Nishimura, T. (1966)
On the a priori information in sequential estimation problems.
IEEE Transactions on Automatic Control, **AC-11**, 2 (Apr. 1966), 197–204; (and in follow-up in **AC-12**, 1 (Feb. 1967), 123.

[32] Mendel, J. M. (1971)
Computational requirements for a discrete Kalman filter.
IEEE Transactions on Automatic Control, **AC-16**, 6 (Dec. 1971), 748–759.



Thomas H. Kerr (S'67—M'74—SM'85) was born in Washington, DC on November 9, 1945. He received the B.S.E.E. in solid state electronics (magna cum laude) from Howard University, Washington, DC in 1967 and the M.S. and Ph.D. degrees (via NSF) in the electrical engineering specialty of control/estimation from the University of Iowa, Iowa City, IA in 1969 and 1971, respectively.

A staff member at M.I.T. Lincoln Laboratory since October 1986 in the areas of Kalman filtering and spectral estimation (which uses the same estimation/detection/identification/pattern recognition tools) for radar and optical target tracking applications, he had been a senior analyst/systems engineer at Intermetrics since 1979 with interests in navigation systems and sonobuoy system tracking/implementation/Kalman filtering/failure detection. His previous experience includes university teaching/research (1967–1971), analysis, computer simulation, and real-time implementation at General Electric's Corporate R&D Center in Schenectady, NY (1971–1973), and navigation/filtering/failure detection applications at TASC (1973–1979). He is currently also teaching Optimal Control in the evening at Northeastern University, Boston, MA. His Department of Defense application involvements include: investigations and computer implementations of parameter identification, mathematical and algorithmic aspects of estimation and control, decentralized and nonlinear filtering, and quantifying associated computer burdens, hypothesis testing/failure detection, effects of failure detection on the reliability/availability of reconfigurable modular composite systems, bicriteria (Pareto-) optimization algorithms, and procedures for Kalman filter sensor selection and scheduling, tractable applications of level-crossing theory, point-process detection in antisubmarine warfare, optimal search and screening techniques, multitarget tracking, RV target tracking and angles-only triangulation tracking, and pattern recognition. These above investigations were performed for various applications including Poseidon and Trident SINS/ESGN navigation systems, JTIDS RelNav, ICNIA and other GPS-aided multisensor avionics navigation applications, minesweeper PINS, PTA sonobuoy tracking, post-coherence function sonobuoy target tracking, ASW search, submarine antenna detectability to radar surveillance, and helicopter-based Missile Warning System development and refinement, SDI satellite survivability, and BSD countermeasures. As part of Lincoln's gathering of IR target signatures under varying tactical scenarios, he is applying his navigation background to the use of an INS in conjunction with possible use of differential GPS data for compensating the tilt of airborne sensors during missions and possibly tagging the measurements with "absolute" geodetic position info for ease in later associations in a data base. His current research relates to the constructive use of Kalman filters in conjunction with neural nets, learning algorithms, and ancillary "chaos" theory, as applied to traditional estimation/control situations.

Dr. Kerr is also a member of Tau Beta Pi, Eta Kappa Nu, Sigma Pi Sigma, Pi Mu Epsilon, Sigma Xi, AIAA, ION, AAAS, MAA, the Naval Institute, ADPA, NSIA, and IEEE AC, IT, AES, and ASSP groups, and CRW.