Application of state estimation to target tracking

Article *in* IEEE Transactions on Automatic Control · March 1984

DOI: 10.1109/TAC.1984.1103466 · Source: IEEE Xplore

Chaw-Bing Chang
Massachusetts Institute of Technology

Application of State Estimation to Target Tracking

CHAW-BING CHANG, MEMBER, IEEE, AND JOHN A. TABACZYNSKI, MEMBER, IEEE

Abstract—**In this paper, we present a survey of problems and solutions in the area of target tracking. The discussion includes design tradeoffs, performance evaluation, and current issues.**

I. Introduction

FOR nearly two decades the target tracking-trajectory estimation problem has been a fruitful applications area for state estimation. Many problems have been solved, yet new and diversified applications still challenge systems engineers. In this paper, we provide a review of the subject and discuss some current issues.

This paper is aimed at two groups of audiences. For practicing engineers with an estimation problem which is the same as or similar to the target tracking problem, this paper may provide a starting point for filter design. It may also be of interest to researchers in the applied estimation area. For them this paper may provide further topics of interest stimulated by new problems in target tracking.

The trajectory estimation problem is a problem of nonlinear estimation. A rigorous treatment of the nonlinear estimation problem requires the use of stochastic integrals and stochastic differential equations (e.g., [1], [6]). In this paper, we will use the familiar "formal" manipulations of the white noise process and omit the rigorous derivations using Ito calculus. This is appropriate especially for the problems discussed in this paper because a mathematical model representing a target trajectory is usually not exact (even in the statistical sense) and the process noise is hardly a strictly "white" noise process. Furthermore, the optimal (conditional mean) nonlinear estimator cannot be realized with a finite-dimensional implementation, and consequently all practical nonlinear filters must be suboptimal. For these reasons, discussion contained in this paper will largely be oriented from a practitioner's point of view and rigorous mathematical terms are only of secondary interest.

An earlier excellent reference on tracking-trajectory estimation [10] deals with tracking algorithms for ballistic reentry vehicles. Most of these algorithms are also applicable to a more general class of targets such as tactical missiles and airplanes. The emphasis of this paper will therefore not be on algorithmic details but rather on discussing problem areas and approaches. References containing estimation algorithms are cited for more interested readers.

This paper is organized as follows. The fundamental problem of target tracking, approaches, and some design tradeoffs are reviewed in Section II. Four approaches for tracking targets with sudden maneuvers are presented in Section III. The discussion includes a comparison of their relative merits. The problem of tracking with passive sensors (measurements containing only line-of-sight angles) is discussed in Section IV. In some applications, one is confronted with large scale system issues, i.e., the existence of many sensors operating in a multiple target environment. In Section V, several algorithms for processing multiple sensor data and an algorithm for correlating measurements from multiple sensors are presented. The problem of tracking in a multiple target environment is discussed in Section VI. In each tracking application, one may be interested in performance evaluation without resorting to Monte Carlo simulations. Covariance analysis techniques are outlined in Section VII for this purpose, namely, the polynomial analysis, the Riccati equations, and the Cramer–Rao bound.

Since this paper focuses discussion on design considerations, performance tradeoffs, and approaches to given problems, several related subjects discussing algorithmic details such as the square root [11] or $U-D$ factorization algorithm [4] are not included. Although not included in this paper, their importance should not be overlooked. Since numerical problems frequently cause filter divergence, serious readers should become familiar with this problem and its solutions [4], [5], [11]. One may notice that certain subjects are given more attention than others, this is due primarily to personal preferences. In each case, however, we try to provide an adequate list of references to allow an in-depth study for interested readers.

II. Fundamentals

A. Problem Definition

The tracking problem is a state estimation problem, i.e., *assuming* the state of a target evolves in time according to the equation

$$\dot{x} = f(x) + w \tag{2.1}$$

and the corresponding (discrete) measurement vector[1] is given by

$$z_k = h(x_k) + v_k \tag{2.2}$$

where w and v_k are input and measurement noise processes, respectively, one is interested in estimating the target states x_k based upon all measurements z_l, $l = 1, \cdots, k$.

We make these following remarks.

1) Equation (2.1) is a *mathematical model* representative of the target dynamics. The state vector x_k usually contains target position, velocity, and sometimes acceleration as state variables. In some situations, key parameters characterizing important target properties are also included as state variables. The filter designer usually has the option of choosing among several models with a

Manuscript received January 4, 1982; revised December 7, 1982. Paper recommended by B. Friedland, Past Chairman of the Surveys and Tutorials Committee. This work was supported by the Department of the Army. The U.S. Government assumes no responsibility for the information presented.

The authors are with Lincoln Laboratory, Massachusetts Institute of Technology, Lexington, MA 02173.

[1] The measurement device can be a radar, sonar, telescope (passive), and others. In most cases, these measurements are taken in discrete time.

0018-9286/84/0200-0098$01.00 ©1984 IEEE

different level of complexity. The tradeoff is performance versus real-time computational requirement.

2) Equation (2.2) is the measurement equation relating state variables to measurement variables. When a radar is used, z_k has at least three components, i.e., range and two angles. If a passive sensor (such as a telescope) is used, z_k only contains two angle measurements.

3) The system (input) and measurement noise processes w and v_k, respectively, are assumed to be zero mean white noise processes. The covariance of w, Q is selected to compensate for modeling errors (discrepancies between (2.1) and the actual process). The statistics of the measurement noise process v_k should also be selected to represent all possible excursions such as measurement biases, false measurements, etc.

B. Basic Approaches

The basic tracking filter is a recursive algorithm. During time t_k to t_{k+1}, the state estimate is computed by integrating an assumed target dynamics[2]

$$\dot{x} = f(x) \tag{2.3}$$

from t_k to t_{k+1} using $\hat{x}_{k/k}$ as the initial state. At time t_{k+1}, a new measurement z_{k+1} is obtained, the state estimate is updated by[3]

$$\hat{x}_{k+1/k+1} + \hat{x}_{k+1/k} + K_{k+1}\left(z_{k+1} - h(\hat{x}_{k+1/k})\right) \tag{2.4}$$

where $\hat{x}_{i/j}$ is the estimate of x_i based upon measurements z_k, $k = 1, \cdots, j$. The following two questions arise.

1) How does one choose $f(\ )$? (See the first remark above.)

2) How is the filter gain K_{k+1} computed?

These two questions which initially appear different are actually intimately related and are discussed below.

1) Target Dynamics: For some targets, a constant velocity (CV) model is sufficient, i.e., the state vector contains six variables

$$x = [x_1, x_2, x_3, x_4, x_5, x_6]^T = [x, \dot{x}, y, \dot{y}, z, \dot{z}]^T \tag{2.5}$$

where (x, y, x) are coordinates of a Cartesian coordinate system used to describe the target dynamics. The state equations are

$$\left.\begin{aligned} \dot{x}_i &= x_{i+1} \\ \dot{x}_{i+1} &= w_i \end{aligned}\right\} \quad i = 1,3,5 \tag{2.6}$$

where w_i is a process noise term used to characterize modeling errors.

If the target being tracked is maneuvering (accelerating), a constant accelerating (CA) model is sometimes used, i.e.,

$$x = [x, \dot{x}, \ddot{x}, y, \dot{y}, \ddot{y}, z, \dot{z}, \ddot{z}]. \tag{2.7}$$

The state equations can be written accordingly, i.e.,

$$\left.\begin{aligned} \dot{x}_i &= x_{i+1} \\ \dot{x}_{i+1} &= x_{i+2} \\ \dot{x}_{i+2} &= w_i \end{aligned}\right\} \quad i = 1,4,7. \tag{2.8}$$

These two state equations are also referred to as the first- and second-order polynomial dynamics, respectively.

Note also that (2.6) and (2.8) assume complete decoupling between x, y, and z. A commonly used Cartesian system for ground based sensors has the coordinate centered at the sensor location with the x-axis pointing east, the y-axis pointing north, and the z-axis perpendicular to the local horizontal plane.

The target dynamics are sometimes described in the sensor coordinates. For example, if a dish radar is used for tracking, the measurement variables include range (r), azimuth (a), and elevation (e). A constant velocity target dynamic model decoupled in r, a, and e results in the following equations:

$$\begin{aligned} \ddot{r} &= w_r \\ \ddot{a} &= w_a \\ \ddot{e} &= w_e \end{aligned} \tag{2.9}$$

where w_r, w_a, and w_e are process noise terms representing modeling errors in r, a, and e directions, respectively. The advantage of using (2.9) is that one is required to construct three two-dimensional filters instead of one six-dimensional filter. Considerable computational savings result. This model, however, is inconsistent with the assumption that the target dynamics is decoupled in the Cartesian coordinate (2.6) and (2.8). A more appropriate model is to retain the coupling among r, a, and e. This results in the following set of nonlinear differential equations:

$$\begin{aligned} \ddot{r} &= r(\dot{e}^2 + \dot{a}^2 \cos^2 e) \\ \ddot{a} &= -2\frac{\dot{r}}{r}a + 2\dot{a}\dot{e}\tan e \\ \ddot{e} &= -2\frac{\dot{r}}{r}\dot{e} - \frac{\dot{a}^2}{2}\sin 2e. \end{aligned} \tag{2.10}$$

We note that using measurement variables as state variables *may* result in more accurate state estimates because this makes the measurement equations linear (see, for example, the discussion of [10]). The Cartesian coordinate results in linear dynamics with nonlinear measurements. It, however, appeals intuitively and provides easier interpretation of target motion.

These models are often used for tracking airplanes and tactical missiles and are also used for tracking ballistic missiles. In some reentry (RV) vehicle applications, however, the vehicle aerodynamic parameters such as the ballistic and lifting coefficients must be estimated in real time. This requires a set of nonlinear differential equations to describe RV motion. A simplified[4] model is

$$\vec{a}(t) = \frac{1}{2}\rho v^2(t)\left[\frac{1}{\beta}\vec{u}_d + \alpha_t\vec{u}_t + \alpha_c\vec{u}_c\right] + \vec{g} \tag{2.11}$$

where $\vec{a}(t)$ is the total acceleration applied on the vehicle
- ρ the air density
- $v(t)$ the magnitude of the vehicle velocity
- β the ballistic coefficient
- $\vec{u}_d$ the unit vector along the drag force direction which is opposite to the velocity vector
- $\vec{u}_t, \vec{u}_c$ two orthogonal unit vectors defining a plane perpendicular to $\vec{u}_d$ for modeling lifting forces
- $\vec{g}$ the gravity force vector.

A state model including β, α_t, and α_c as state variables results in a nine-state state vector. Further discussion of this problem is found in [12]–[15].

2) Filter Gain Computation: The filter update equation (2.4) gives the weighted sum of the one-step predicted state and the difference of the new and predicted measurement. (This difference is called the filter residual process. If the optimal nonlinear filter is used, it is also called the innovation process.) The filter gain provides the weighting of this update procedure. When the filter gain is very small, the estimator becomes insensitive to the new measurements and this may result in large bias errors. When the filter gain is very large, the estimator is deemphasizing

[2]For higher order nonlinear filters, one may include additional terms involving estimation covariance; see, for example, [1].

[3]Similarly, this equation represents a first-order nonlinear filter.

[4]This is a simplified model neglecting Coriolis and centrifugal forces. They are usually included using the vectory sum as in (2.11). Coriolis and centrifugal forces become important when the length of target trajectory becomes comparable with earth radius. The case of reentry vehicle and satellite tracking is in this category of problems.

past measurements, possibly resulting in larger random errors. A balance can be achieved if one has sufficient knowledge regarding the accuracy of the state model and the measurement process. Such knowledge is embedded in the filter gain computation. Four approaches for treating this problem are discussed. These include the adjusting of the process noise covariance for the extended Kalman filter, the finite memory filter for process noise-free systems, the exponentially aging filter, and the α-β-γ filter.

a) The Extended Kalman Filter (EKF): If an extended Kalman filter is used, the filter gain is

$$K_k = P_{k/k} H_k^T R_k^{-1} \tag{2.12}$$

where H_k is the measurement Jacobian matrix, R_k is the measurement noise covariance, and $P_{k/k}$ is the covariance of the state estimate $\hat{x}_{k/k}$ obtained by solution of the matrix Riccati equations, i.e.,

1) from t_k to t_{k+1}, compute $P_{k+1/k}$ by integrating

$$\dot{P} = FP + PF^T + Q \tag{2.13a}$$

with initial condition $P_{k/k}$,

2) at t_{k+1}

$$P_{k+1/k+1} = P_{k+1/k}\Big[I - H_{k+1}^T\big(H_{k+1}P_{k+1/k}H_{k+1}^T + R_{k+1}\big)^{-1} \cdot H_{k+1}P_{k+1/k}\Big] \tag{2.13b}$$

where F is the system Jacobian matrix and Q is the process noise covariance matrix. The matrix Q must be chosen to be representative of modeling errors. For example, if a CV model is chosen while the target may actually be accelerating, Q should be chosen proportional to the expected magnitude of target acceleration. A method for selecting Q such that the filter computed covariance is an upper bound of the actual performance is suggested in [16]. Depending upon specific applications, the value of Q can also be obtained empirically using Monte Carlo simulation. This is often called "filter turning" by adjusting the Q value so that the error statistics obtained with Monte Carlo simulation is in close agreement with the filter-computed covariance [1], [5], [15].

Modeling error may become a significant contributor to estimation error when the Riccati equation reaches a very small steady-state solution (the P matrix). In this case the filter gain (2.12) becomes nearly equal to zero and the filter is essentially running open-loop (2.4). The use of a large Q forces the steady-state P matrix to stay "significantly" large, so that the estimate is sensitive to the most recent measurements. In many applications, the selection of Q is not an exact science [17].

The above discussion presents a *heuristic* argument on the use of a process noise covariance Q. Besides methods using covariance upper bounds and Monte Carlo studies there also exists considerable literature which discusses methods of estimating Q, R (and sometimes K directly) in real time based upon the statistics of the innovation process. One earlier work in this area is by Mehra [18]. We briefly outline Mehra's method below.

It is well known that if both the system model and noise statistics are true representations of the actual physical process, the filter innovation process for linear systems

$$\gamma_k = z_k - H\hat{x}_{k/k-1} \tag{2.14}$$

is white Gaussian with zero mean and covariance[5]

$$P_{\gamma_k} = H_k P_{k/k-1} H_k^T + R_k \tag{2.15}$$

and the filter achieves optimal performance. In [18], a method for testing filter optimality based upon the aforementioned property of the innovation process was presented. If the test indicates that the filter does not attain optimal performance, one then proceeds to adjust Q and R so that the covariance of the innovation process will be consistent with that of filter prediction. This method first computes a sampled correlation function assuming that γ_k is ergodic over a certain time interval, one therefore has

$$\begin{aligned} C_j &\triangleq E\big[\gamma_i\gamma_{i-j}^T\big] = HP_\infty H^T + R; \qquad j = 0 \\ &= H\big[\Phi'(I - KH)\big]^{j-1}\Phi\big[P_\infty H^T - KC_0\big]; \qquad j > 0 \end{aligned} \tag{2.16}$$

and

$$\hat{C}_j = \frac{1}{N}\sum_{i=1}^{N}\gamma_i\gamma_{i-j}^T \tag{2.17}$$

where P_∞ denotes the steady-state error covariance matrix and Φ is the state transition matrix of the system dynamics.

Using the above equations and the steady-state Riccati equation, Mehra gives a procedure for solving for Q and R. There are situations in which there is no sufficient number of independent equations for solving them; Mehra then gives a recursive procedure for solving for the filter gain K directly.

In addition to Mehra's method, a number of additional approaches are available in the literature. This includes a different solution algorithm for the Mehra approach above [19], maximum likelihood estimator [20], Bayesian estimator [21], covariance matching technique [22], among others. Reference [22] gives a survey of techniques in the area of adaptive filtering.

We emphasize that the concept presented above is important because it indicates that systematic procedures for identifying the noise covariance or filter gain can be established. It may not be very useful for real-time applications because of its computational requirement. For nonreal-time applications, however, this method is useful for simulations and postmission data analysis studies.

There exist other methods for reducing performance sensitivities to model errors. These include the finite memory filter [1], [10], and the fading memory filter [1], [10], [23]–[25].

b) The Finite Memory Filter: The finite memory filter applies a sliding window to the data and computes the state estimate based only upon data in that time span. The window width is selected such that the system model is an adequate approximation to the actual process over the time interval. During this time interval, the system is assumed to be *noise free*. This assumption is related to the selection of the time interval in that the variation of the unknown parameter over this interval is small. The filter to be discussed is due to Jazwinski [1]. We briefly outline it below.

Let the measurement sequence be denoted by

$$z_1, z_2, \cdots, z_m, z_{m+1}, \cdots, z_k$$

and $N = k - m$, where N is the total number of measurements desired in the finite memory filter. Let the finite memory state estimate and covariance be denoted by $\hat{x}_{k/m,k}$ and $P_{k/m,k}$, respectively; they can be computed using the following equations:

$$\begin{aligned} \hat{x}_{k/m,k} &= P_{k/m,k}\big(P_{k/k}^{-1}\hat{x}_{k/k} - P_{k/m}^{-1}\hat{x}_{k/m}\big) \\ P_{k/m,k}^{-1} &= P_{k/k}^{-1} - P_{k/m}^{-1} \end{aligned} \tag{2.18}$$

where $\hat{x}_{k/k}$ is the state estimate at time k based upon all data up to and including z_k, and $\hat{x}_{k/m}$ is the state estimate at time k based upon all data up to and including z_m. One can interpret the above equations to say that the finite memory estimate is obtained by subtracting an estimate based on all data prior to the

[5] In most applications involving nonlinear systems using an extended Kalman filter, this expression appears to be a close approximation.

time window from the estimate based upon all data. We note that the above equations can be *rederived* to obtain a computationally more efficient and numerically more stable algorithm. For more discussions, see [1], [3].

Note that the finite memory filter requires that a batch of N measurement vectors be stored (for updating $\hat{x}_{k/m}$). Furthermore, the computational burden is much larger than that of the EKF. Typically, the cost of added computation outweighs the benefit when compared to other methods.

c) The Fading Memory Filter: The fading memory filter (sometimes referred to as the aging filter) weights recent data exponentially higher than past data. A derivation of this filter can be found in [2], [23]–[25]. The resulting algorithm turns out to be very simple. Let $P^*_{k+1/k}$ denote the covariance of the one-step predicted estimate of the aging filter; this is related to the Kalman filter covariance [solution of (2.13a)] by

$$P^*_{k+1/k} = \alpha P_{k+1/k} \tag{2.19}$$

where α is a scalar quantity greater than unity. The $P^*_{k+1/k}$ is then used in (2.13b) to obtain the update covariance $P_{k+1/k+1}$. The scalar α is the exponential weighting factor. This is accomplished by changing the measurement noise covariance matrix to

$$R^*_i = R_i(\alpha)^{k-i} \tag{2.20}$$

for $i=1,\cdots,k$, where k is the current time. With measurement noise covariance, data rate, and an assumed modeling error, one can find an optimum α for minimizing the mean square error.

d) A Constant Gain Filter—The α-β-γ Tracker: In some cases, because of computational constraints, it may be impractical to compute the filter gain in real time. Under such conditions one must use either a set of precomputed filter gains or a constant gain filter. One commonly used constant gain filter is based upon the steady-state solution of the Riccati equation. In order to minimize the modeling error effect, a sufficiently large process noise covariance Q must be selected so that the residual process resembles a white noise process.

Another popular constant gain filter is the α-β-γ filter (or α-β filter when using a CV model) [26], [27]. The main difference between the α-β-γ filter and the steady-state gain filter is that the former assumes the complete independence of the three spatial coordinates in the filter update equation. Notice that the state space and the measurement space may be related through a nonlinear function (2.2). In using the α-β-γ filter, this assumption is not allowed. Let (r, a, e) denote the radar range, azimuth, and evaluation, respectively; the state vector becomes

$$x = [r, \dot{r}, \ddot{r}, a, \dot{a}, \ddot{a}, e, \dot{e}, \ddot{e}]^T. \tag{2.21}$$

Let r denote $[r, \dot{r}, \ddot{r}]^T$, then the filter update equation becomes

$$\hat{r}_{k+1/k+1} = \hat{r}_{k+1/k} + K_{k+1}[z_{k+1} - \hat{z}_{k+1/k}] \tag{2.22}$$

where z_{k+1} is the range measurement at time $k+1$. The gain matrix K_{k+1} for the α-β-γ filter is

$$K_{k+1} = \left[\alpha, \frac{\beta}{t_{k+1} - t_k}, \frac{2\gamma}{(t_{k+1} - t_k)^2}\right]. \tag{2.23}$$

The update equations for azimuth and elevation can be obtained accordingly. Note that α, β, and γ are filter gain components for position, velocity, and acceleration, respectively. The structure of the gain equation (2.23) allows these parameters to be free of time interval between measurements $t_{k+1} - t_k$.

Note that in (2.12), when an extended Kalman filter is used, the gain matrix K_k has the dimension (9×3) while the gain matrix for the α-β-γ filter consists of three (3×1) vectors. There is a wide range of methods for choosing the values of α, β, and γ. One method is to compute the steady-state Kalman gain in the above chosen coordinate. A method often used in practice is to conduct extensive Monte Carlo simulation studies to define α, β, γ over a variety of cases for a given application [28].

e) Summary: In Section II-B-2 we have discussed the filter gain design for reducing performance sensitivity to modeling errors. Four approaches, which include the adjustment of the process noise covariance for the extended Kalman filter, the finite memory filter, the fading memory filter, and the α-β-γ filter, were discussed. We feel that the extended Kalman filter with an appropriately chosen process noise covariance is a good choice for a variety of problems provided that the real-time computational requirement is not excessive. The α-β-γ filter offers simplicity although with degraded performance. The finite memory and fading memory filter are only secondary choices because they do not seem to provide any advantages over the extended Kalman filter while the computational requirement is comparable or even more severe. We emphasize that the above conclusion is based upon our experiences over a wide range of problems. Readers *may*, however, find different conclusions for their *specific* applications.

C. A Batch Filter for Deterministic Systems

In Section II-B-2-b we discussed a finite memory filter developed by Jazwinski. The input elements of that algorithm are the outputs of the recursive Kalman filter. The estimate is based upon data from a finite time interval and a batch of N measurement vectors corresponding to that time interval must be stored. Another assumption of that filter is that the system model is noise free (deterministic) and this assumption is reasonable because the finite time interval is selected to reflect the tolerance on model errors. The Jazwinski algorithm is optimum for linear systems under the given conditions.

In this section, we present an algorithm for nonlinear systems. This algorithm iteratively solves for the optimum estimate based upon a *noise-free* system model and a batch of N measurement vectors $z_1, z_2, \cdots, z_N$. This filter is fundamentally different from the approach of Jazwinski. For linear systems, this approach gives the same result as the Kalman filter. For nonlinear systems this approach gives a better estimate than that of the extended Kalman filter (see, for example, [29]).

Let the system and measurement equations be the same as (2.1) and (2.2) except that w is now zero. The batch of N measurement vectors is denoted by $\{z_1, z_2, \cdots, z_N\}$. Let $\hat{x}^k_{N/N}$ denote the kth iteration of the estimate of x_N, then

$$\hat{x}^{k+1}_{N/N} = \hat{x}^k_{N/N} + P^k_{N/N}\left[\sum_{n=1}^{N} G^{k^T}_n H^{k^T}_n R_n^{-1}\left(z_n - h\left(\hat{x}^k_{n/N}\right)\right)\right] \tag{2.24}$$

$$P^k_{N/N} = \left[\sum_{n=1}^{N} G^{k^T}_n H^{k^T}_n R_n^{-1} H^k_n G^k_n\right]^{-1} \tag{2.25}$$

where

G^k_n $\Phi^{k^{-1}}_n G^k_{n+1}$; $n = N-1, N-2, \cdots, 1$

G^k_N I (an identity matrix)

Φ^k_n the solution of $\partial\Phi^k(t,\tau)/\partial t = F^k_t \Phi^k(t,\tau)$; $\Phi(\tau,\tau) = I$, $\tau = t_n$, $t \in [t_{n+1}, t_n]$

F^k_s the Jacobian matrix of $f(\hat{x}^k_{s/N})$

H^k_n the Jacobian matrix of $h(\hat{x}^k_{n/N})$

$\hat{x}^k_{n/N}$ result of integrating $\dot{x} = f(x)$ backward from t_N to t_n using $x_N = \hat{x}^k_{N/N}$.

The iteration is terminated when $\hat{x}^{k+1}_{N/N}$ and $\hat{x}^k_{N/N}$ are sufficiently close.

We make the following remarks.

1) The above algorithm is a realization of the maximum likelihood estimator with Gaussian measurement noise process. It is well known that the maximum likelihood estimate is asymptotically efficient and Gaussian and approaches the Cramer–Rao bound. This algorithm does not assume initial state estimate and covariance. If available, they can be included straightforwardly.

2) The $P_{N/N}$ of (2.25) is an approximate expression for the covariance of $\hat{x}_{N/N}$. The $P_{N/N}$ evaluated at the true state is the Cramer–Rao lower bound on the covariance of $\hat{x}_{N/N}$. Since $\hat{x}_{N/N}$ approaches the true state with probability one, $P_{N/N}$ also approaches the Cramer–Rao bound with probability one.

3) Note that the inverse of $P_{N/N}$ is Fisher's information matrix. The invertibility of the information matrix is tied with the observability of the system; see, for example, [30].

4) For linear systems a closed-form solution can be found and the iterative procedure becomes unnecessary. This solution can be shown to be the same as the Kalman filter for process noise-free systems.

There are many application areas for this algorithm. For example, in tracking space objects where the target dynamics can be modeled very accurately, the algorithm of this section is particularly suitable. This method has been used for ballistic trajectory tracking with angle-only measurements [29] and tracking of deep space satellites [31]. Another application is for *track initiation*. Since the initial covariance and state estimates are not generally given *a priori*, the above algorithm can obtain the best estimates based on the first N measurement vectors and then proceed to use $\hat{x}_{N/N}$ and $P_{N/N}$ as the initial state and covariance estimates, respectively. This method is sometimes referred to as the information matrix approach for filter initiation.

D. Summary

In the above, we have discussed various algorithms for addressing the basic tracking problem. These approaches employ simple to sophisticated system models for target dynamics and attempt to compensate for modeling errors in a variety of ways. Several well-known nonlinear estimation algorithms were not discussed. These include the second-order filter [32], the single stage iterative filter [33], and the Gaussian sum filter [34] among others [35]. They are not included because of their excessive computational requirement although these algorithms can indeed improve the estimation accuracy.

Before closing this section, a brief algorithm comparison is given. If the objective of tracking is to obtain precision information about the target dynamics, then one should use the most accurate target model and apply the EKF (or even more sophisticated algorithms). If the dynamic model is sufficiently accurate so that the process noise term is negligible, then the algorithm considered in Section II-C is a good choice provided that the computation time and data storage requirements are not excessive.

If the objective is just to maintain the target in track, then one may use the simplest track algorithm such as the α-β-γ tracker. One exception to this case is when tracking in a dense target environment where precision tracking may be necessary for target correlation. This subject will be discussed later.

The finite and fading memory filters are usually the secondary choices (especially the finite memory filter) because the same purpose (reducing sensitivity to model errors) can be achieved by adjustment of the process noise covariance Q. They are nevertheless included here because 1) they reflect the historical development of adaptive filtering techniques; and 2) they still provide an option for readers to choose for their applications.

Finally, we comment that a number of additional approaches which can be used for reducing performance sensitivities to model errors and preventing filter divergence are not reviewed in this paper. These include the Schmidt's epsilon technique [1], [2] and arbitrarily lower bounding the diagonal terms of the covariance matrix. The Schmidt technique is to include a bias corrective term for the state update equation and the final algorithm results in an additional term for the filter gain. This is similar to the form of limited or fading memory filter. The method of lower bounding the diagonal term of the covariance matrix is rather arbitrary, it is usually selected with "engineering judgment" for a specific application.

III. Targets with Sudden Maneuvers

Targets with sudden maneuvers can be modeled as systems with abrupt changes. We modify (2.1) to become two sets of equations, one representing the premaneuver dynamics and the other incorporating the maneuver feature

$$\dot{x} = f(x, x_m) + w \tag{2.1a}$$

where x_m is the vector representing maneuvering force and satisfies

$$x_m = 0, \qquad \text{for } t < t_m \tag{3.1}$$

and

$$\dot{x}_m = f_m(x_m) + w_m, \qquad \text{for } t \geq t_m \tag{3.1a}$$

where t_m is the time the maneuver begins, $f_m(\cdot)$ is the maneuvering dynamics, and w_m is the system noise for $f_m(\cdot)$. For targets with sudden maneuvers, t_m is unknown, $f_m(\cdot)$ and w_m may be unknown or partially known.

In tracking airplanes for example, the target may first be flying in a straight line with constant speed. A CV model is adequate for the airplane dynamics in this case. A sudden maneuver of this airplane implies that the airplane is accelerating unexpectedly and the acceleration is time-varying and following an unknown profile. The instantaneous acceleration vector is therefore the maneuvering vector x_m defined above.

In tracking a ballistic reentry vehicle, the target dynamics is the equation (2.11) with α_c and α_t equal to zero. A sudden aerodynamic maneuver means that α_c and α_t become nonzero and follow unknown time-varying profiles. The maneuvering vector in this case contains α_c and α_t as its elements. Four approaches to this problem are discussed individually below.

A. Filter Compensation Using Process Noise Covariance

In this first method, one simply ignores the maneuver vector x_m and lumps the system errors introduced by x_m with the process noise term w. If the estimator's only concern is to maintain the target in track (adequate position estimation accuracy), this method can work quite well.

Basically, it examines the "regularity" of the filter residual vector

$$\gamma_k = z_k - h(\hat{x}_{k/k-1}) \tag{3.2}$$

against its covariance matrix

$$P_{\gamma_k} = H_k P_{k/k-1} H_k^T + R_k \tag{3.3}$$

using (the chi-square variable)

$$l_k = \gamma_k^T P_{\gamma_k}^{-1} \gamma_k. \tag{3.4}$$

When l_k becomes too large one suspects that the target is maneuvering and the covariance of w, Q is increased so that l_k is reduced to a reasonable value. This method therefore has the combined feature of maneuver detection and filter compensation.

This method is based upon the adaptive filter of Jazwinski [36] and shares great similarities with the method of Mehra [18] (see also Section II-B-2-a). The main difference is that the above method tests the filter regularity using chi-square statistics, and stresses the simplicity (not the optimality) of the approach.

A thorough discussion of this method and its performance against maneuvering reentry vehicles can be found in [14], [15]. A similar technique was discussed in [37] for target tracking with an infrared sensor.

B. State Augmentation

The second method is straightforward, computationally more costly, but with substantially better performance than the previous method. This method is to include x_m as part of the state vector, i.e., the augmented state consists of

$$x_a = \left[x^T, x_m^T\right]^T. \tag{3.5}$$

In the case when the target maneuvering dynamic is completely unknown, one uses

$$\dot{x}_m = w_m \tag{3.6}$$

where w_m is modeled as a zero mean white noise process with covariance Q_m. If bounds on the magnitudes of maneuvers are known, a method for choosing Q_m such that the actual filter performance is bounded by the computed filter covariance can be found in [16]. As a rule of thumb, the values of the entries in Q_m should be a fraction of the expected magnitude squared of the maneuver force and proportional to the measurement time interval. A method for selecting Q_m for estimating ballistic and lifting coefficients of reentry vehicles is discussed in [14], [15]. For some applications, the process noise covariance term may be required to appear at states in addition to the states representing target maneuvers.

Notice that (3.6) assumes x_m uncorrelated in time. This assumption is sometimes not very realistic. A model often used in airplane tracking is

$$\dot{x}_m = -\alpha x_m + w_\alpha \tag{3.7}$$

where α is the correlation constant and w_α is a noise process. Methods for selecting values for α and statistics for w_α and the performance against airplane tracking can be found in [38].

The maneuvering state x_m is usually not influenced by the state vector x (2.1a), (3.1a). With this assumption, one can compute the state and maneuver estimates separately to obtain

$$\hat{x}_{k/k} = \tilde{x}_{k/k} + A_k \hat{x}_{m,k} \tag{3.8}$$

where $\tilde{x}$ is the estimate assuming x_m is zero, $\hat{x}_{m,k}$ is the maneuver estimate, A_k is a gain matrix, and $\hat{x}_{k/k}$ is the final state estimate. The advantage of this decoupled implementation is a saving in computation. It can be shown that the decoupled estimator is optimum for linear systems when t_m is known and the maneuver state model is known and deterministic; see, for example, [39], [40]. Suboptimal designs for the case when the above assumptions are not true can be found in [16].

Since the maneuver time t_m is unknown, a trival application of the above method is to use it throughout the entire track. Inevitably, the filter performance is degraded when the target is nonmaneuvering [14], [15]. An ideal extension is to use a maneuver detector for switching the tracking filter from nonmaneuver to maneuver mode, this subject is discussed in the following subsection.

C. Maneuver Detection

The detection problem is the problem of discriminating the following two hypotheses based upon filter residuals:

$$\begin{aligned} &H_1: \tilde{\gamma}_k = \gamma_k + g_k(x_m); \text{ maneuvering target hypothesis} \\ &H_0: \tilde{\gamma}_k = \gamma_k; \text{ nonmaneuvering hypothesis} \end{aligned} \tag{3.9}$$

for $k = k_0, \cdots, K$ where γ_k is the residual vector before the maneuver starts, and $g(\ )$ is a known function relating the maneuvering vector x_m to the residual vector. A generalized likelihood ratio test is

$$\Lambda = \frac{\max\limits_{x_m} p(\tilde{\gamma}_k; k_0, \cdots, K/H_1)}{p(\tilde{\gamma}_k; k_0, \cdots, K/H_0)} \gtrless \lambda. \tag{3.10}$$

For a given application (target dynamics, sensor type, etc.), the above equation can be further simplified [15], [40]–[43].

The use of a generalized likelihood ratio test for detecting maneuvers is discussed in [14], [15] for reentry vehicles and in [41] for airplanes. General discussions on the detection of sudden changes in linear systems can be found in [40] and [42], [43].

Drawbacks of using a detector-directed tracking include: 1) detection delay; 2) large transient errors during filter switch; and 3) storage and processing time requirement for a large batch of past measurements (see [42]). Numerical results in [15] show large estimation errors immediately after filter switch. This is partially due to the maneuvering filter going through its transient period.

Note that the maneuver detection relies on the fact that target maneuvering generates residual bias in a nonmaneuvering tracker. Once the filter is switched to maneuvering, the above detection scheme cannot be used to discriminate if the target has returned to nonmaneuvering status.

All the above problems can potentially be alleviated if one employs the adaptive multiple model estimator [44], [50], discussed in the next subsection.

D. Multiple Model Estimator

Let H_1 denote the hypothesis that the target is maneuvering and H_0 the hypothesis that the target is nonmaneuvering. One may construct two filters, the first one uses a maneuvering dynamic while the other one uses a nonmaneuvering model. Let these respective states be denoted by x_k^1 and x_k^0; the optimum estimate $\hat{x}_{k/k}$ is obtained using

$$\hat{x}_{k/k} = P_1(k)\hat{x}^1_{k/k} + P_0(k)\hat{x}^0_{k/k} \tag{3.11}$$

where $P_i(k)$ is the *a posteriori* probability that H_i is true at time t_k. The computation of $P_i(k)$ can be very complicated. If one uses a simplifying assumption that hypotheses at time t_k are independent of hypotheses before t_{k-1}, i.e., a first-order Markov process, then one obtains a much simplified expression for $P_i(k)$:

$$P_i(k) = \frac{\sum\limits_{j=0}^{N} p_{ij}(k) P_{ij} P_j(k-1)}{\sum\limits_{i=0}^{N}\sum\limits_{j=0}^{N} p_{ij}(k) P_{ij} P_j(k-1)} \tag{3.22}$$

where $p_{ij}(k)$ is the residual density assuming that the ith hypothesis is true at time t_k and the jth hypothesis is true at time t_{k-1}, P_{ij} is the transition probability, and $P_j(k-1)$ is the *a posteriori* probability that H_j is true at time t_{k-1}. Note that we have made the above equation slightly more general by assuming that there are total of $N+1$ hypotheses. The $\hat{x}^i_{k/k}$ is computed using

$$\hat{x}^i_{k/k} = \sum_{j=0}^{N} \hat{x}^{ij}_{k/k} P\big(H_j(k-1)/H_i(k), Z_k\big) \tag{3.23}$$

$$P\big(H_j(k-1)/H_i(k), Z_k\big) = \frac{p_{ij}(k) P_{ij} P_j(k-1)}{\sum_{j=0}^{N} p_{ij}(k) P_{ij} P_j(k-1)} \tag{3.24}$$

where $\hat{x}^{ij}_{k/k}$ is the state estimate using $\hat{x}^j_{k-1/k-1}$ updated with the ith hypothesis and Z_k denotes all the measurements up to time t_k.

Derivations and discussion of the above results in a more general context are given in [44], [45]. The advantage of the above approach is that when the target switches between maneuvering and nonmaneuvering modes, the hypothesis probability values change to provide a smooth transition of the final estimates $\hat{x}_{k/k}$.

A higher level multiple model approach is to use several hypotheses to model different maneuvering force levels. This approach enhances the estimation performance at the cost of more computation resources. Furthermore, it introduces the problem on how to select the discretized maneuvering levels for the best tracking performance. This problem in a more general context is a current topic of active research.

We would like to emphasize that the multiple model estimator is a general adaptive estimation technique. It was first derived by Magill [21] for the time invariant hypothesis case, i.e.,

$$P_{ij} = \begin{cases} 1, & \text{when } i = j \\ 0, & \text{otherwise.} \end{cases} \tag{3.25}$$

Its extension to the switching hypothesis case was the subject of [44]. References [46] and [47] stated the above concept in the continuous domain and gave a representation theorem known as the partition theorem.

A word of caution about the multiple model method; it is a tightly tuned algorithm (for the case of modeling various maneuvering levels) and therefore very vulnerable to the situation where none of the models match the actual dynamics. In this case, strange filter behavior may appear and this behavior is not yet completely understood from a theoretical point of view [45], [48].

Discussions applying this method to target tracking can be found in [49], [50]. Its extension to adaptive control was applied to the flight control of a F-8C experimental aircraft [48].

E. Summary

In this section, we have discussed four approaches to tracking targets with sudden maneuvers. The first method (Section III-A) provides a way of adjusting the process noise covariance level through use of the filter residual "regularity" to compensate for the modeling error induced by target maneuvering. This method uses the least computation, but does not give very precise velocity and parameter estimates. The second method (Section III-B) is to augment the state vector with maneuvering variables. The dimension of the state vector is enlarged and the filter is therefore computationally more costly. It does however provide more accurate state estimates. The drawback to this method besides the higher computational burden is that the filter performance is degraded when the target is nonmaneuvering. A method to circumvent this problem is to use a maneuver-detector-directed filter (Section III-C). This method suffers from large errors occurring during filter switch over and requires the storage and reprocessing of a large batch of past measurements. A compromise to this problem is to construct two filters (Section III-D) with one using a maneuvering dynamic model and the other a nonmaneuvering dynamic model. The final estimate is a weighted sum of outputs of these two filters using the *a posteriori* hypothesis probabilities as weighting factors. This method has the advantages of all the above approaches but at a cost of a much larger computational burden. One may also use a bank of filters to model different maneuver levels and apply the multiple model adaptive estimation method. This method may work very well, it is however, also very sensitive to model mismatch errors and system nonlinearites.

IV. Tracking with Angle-Only Measurements

In this section the problem of tracking with a sensor measuring only the target line-of-sight angle is discussed. There may be two separate objectives of tracking. The first one is simply trying to maintain targets in track in the angle domain. In this case, techniques discussed in Section II are applicable. The only difference is that in the angle-only case the target dynamic equations are described in two-dimensional coordinates. Polynomial equations decoupled in two orthogonal angular directions are often used. The second tracking objective is to obtain estimates of the complete state vector as defined in a three-dimensional coordinate system. This objective may not always be achievable since tracking with angle-only measurements may constitute an unobservable system. Physically, it can be explained as follows. In radar tracking systems, each measurement vector determines the instantaneous target position to within a finite uncertainty volume, i.e., uncertainties in both range and angles can be expressed with finite standard deviations. In an angle-only tracking system, the uncertainty volume of each measurement vector is infinite (due to the inability to measure range). Such a system may be observable only for certain types of target dynamics. For example, when a telescope is used to track a satellite, this constitutes an observable system and target range can be estimated (with large errors, however) because the satellite trajectory is influenced by the earth's gravity [29]. If a telescope is used to track an airplane traveling with constant speed, this system is unobservable, i.e., the estimation of the three-dimensional dynamics is impossible.

There are means available for enhancing the observability of the system. For example, one may use two passive sensors at separate locations simultaneously tracking the same object. The intersection of two angular beams gives the total measurement uncertainty which now has a finite uncertainty volume. One complication of this approach occurs when tracking in a multiple target environment. Here, one is confronted with the problem of recognizing the same target at both sensors. The subject of sensor-to-sensor correlation and efficient methods of processing multiple sensor measurements are discussed in the next section. Another method of enhancing observability is to incorporate other types of measurements in addition to angle measurements. One such application is in passive sonar tracking [51] where target Doppler measurement is included. A passive sonar system with Doppler measurements makes the system completely observable.

We note that orbit estimation using angle-only measurements is an ancient problem (see [9] for a discussion of classical approaches and a list of references). Most emphasis of [9], however, was placed on the planetary mechanics and dynamic modeling and very little effort was applied from the estimation point of view. For example, the Gauss method for orbit determination ensures the solution of the entire orbit with only three angle measurements. This is true if noise-free angle measurements can be made. With even slight errors in angle measurements, the target range estimation error can be very large.

When one is tracking a ballistic object (a satellite or a long range ballistic missile) using a passive sensor, one is confronted with two questions: 1) how to initiate a Kalman type recursive tracking filter; and 2) since the range estimation error will inevitably be larger, will the extended Kalman filter provide adequate performance. In [29], these issues were studied in detail. It was shown that the batch filter described in Section II-C can be used to provide initial conditions for the extended Kalman filter. Other discussions include methods for computing an initial guess for the iterative procedure, applications of the batch filter recursively for tracking, and techniques for incorporating trajectory *a priori* knowledge (bounds on heading angles, velocity, energy, etc.) for improving the state estimation accuracy. An important conclusion of [29] is that the performance of the batch filter asymptotically approaches the Cramer–Rao bound (see Section VII-C of this paper) for the covariance of state estimates while the performance of the extended Kalman filter generally does not have this property.

Finally, we remark that recent developments in target tracking on the focal plane using an infrared sensor are in the problem areas of both signal processing and target tracking. The approach discussed in [37] and [52] is to first process focal plane data with a prescribed model, then establish two-dimensional target state estimates using a Kalman filter.

V. Tracking with Multiple Sensors

There are a number of applications in which multiple sensors operate in a multiple target environment. There are two problems which confront a multiple sensor tracking system: 1) how to efficiently update the tracking filter with multiple sensor measurements; and 2) how to identify the same target for all sensors in a multiple target environment. The first question is discussed in Section V-A. The second question is a special case of tracking in a dense target environment. It fits however, conveniently in the context of sensor-to-sensor correlation and will therefore be discussed in Section V-B.

A. Filter Update Algorithm Considerations

Consider the situation in which there are several sensors simultaneously observing a single target. If these measurements are *not* time synchronized, one can formulate the problem as that of state estimation with nonuniform measurement times. It should be noted that in this particular formulation the functional transformation from state space to observation space depends upon the geometry, hence one can be confronted with a situation in which the measurement equations change from observation to observation.

If the measurements *are* time synchronized (such as in a multistatic radar system), then one is interested in seeking the most efficient way of processing these measurements. Let $z_{k,i}$ denote the measurement at time t_k from the ith sensor, then

$$z_{k,i} = \boldsymbol{h}_i(\boldsymbol{x}_k) + \boldsymbol{v}_{k,i}, \qquad i = 1, \cdots, I. \tag{5.1}$$

Three processing methods are discussed by Willner *et al.* [53].

1) Parallel filter: the set of measurement vectors are concatenated to form a new measurement vector

$$z_k = \left[z_{k,1}^T, z_{k,2}^T, \cdots, z_{k,I}^T \right]^T \tag{5.2}$$

which is processed by the filter at once.

2) Sequential filter: each measurement is processed sequentially by the filter with zero prediction time between measurements.

3) Data compression filter: prior to processing, measurement vectors are transformed to a common coordinate system and compressed to a single measurement vector. Let $z_{k,i}$ denote the measurement of ith sensor and all $z_{k,i}$'s have been transformed to a common coordinate system. If measurements among sensors are uncorrelated, the compressed measurement vector z_k is

$$z_k = R_k \left[\sum_{i=1}^{I} R_{k,i}^{-1} z_{k,i} \right] \tag{5.3}$$

$$R_k = \left[\sum_{i=1}^{I} R_{k,i}^{-1} \right]^{-1}. \tag{5.4}$$

The above three methods can be shown to be equivalent and optimum for linear systems [53]. The data compression method is computationally the most efficient for a wide range of cases [53].

It is possible to use decentralized processing wherein each sensor produces its own estimate. These are then combined to form a single estimate. The major drawback to such an approach is that the computational burden is significantly higher than any of the methods described above.

Algorithms for data compression with nonsynchronized data vectors can be derived straightforwardly. This may involve integration of measurement vectors to achieve time alignment followed by an application of the data compression equations.

B. Sensor-to-Sensor Correlation

One important problem facing a multiple sensor tracking system in a multiple target environment is the unique identification of the same target as observed by each sensor. There are two approaches to this problem. The first approach attempts to correlate the existing track files (state estimate of a given string of measurements) with measurements. This method is the same as the track continuation mode of tracking in the dense target environment. We will address this approach in the next section. The second method is to directly correlate the set of measurements from the ith sensor with that of the jth sensor. A brief discussion of the second method is given below.

Consider the case when there are two sensor simultaneously tracking multiple targets. The results which follow can be extended to that of multiple sensors. Let $\{y_i; i = 1, \cdots, N\}$ and $\{z_j; j = 1, \cdots, N\}$ denote N measurements obtained by the first and second sensors, respectively,[6] and respective covariances are given by P_i and E_j. We further assume that measurements between these two sensors are uncorrelated. Consider now the question of which y_j corresponds to a particular z_i. This is an N-array hypothesis decision problem. That is, for a given z_i, one asks which of the following hypotheses is true:

$$\begin{cases} H_1: \; z_i = y_1 + \boldsymbol{n}_i \\ \quad \vdots \\ H_N: \; z_i = y_N + \boldsymbol{n}_i \end{cases} \tag{5.5}$$

where $\boldsymbol{n}_i$ is a random noise vector with zero mean and covariance E_i. Using the likelihood ratio test procedure, it can be shown that the y_j is chosen as the one which maximizes the joint density of z_i and y_j. That is, decide that H_j is true when

$$p(z_i, y_j) = \max_l p(z_i, y_l). \tag{5.6}$$

If z_i and y_j are Gaussian random vectors, then one obtains the following equivalent procedure.

Decide that H_j is true when y_j minimizes

$$w_{il} = (z_i - y_l)^T (P_l + E_i)^{-1} (z_i - y_l) + \ln(|P_l + E_i|) \tag{5.7}$$

[6] We have obviously neglected the problem of imperfect detection at the sensor and unequal coverage problem by assuming that each sensor observes the same number of targets and that there is a one-to-one assignment between them. For discussions which include these factors, consult [54].

for all $l=1,\cdots,N$. Notice that (5.7) is a chi-square random variable or a weighted distance measure.

The above discussion gives a procedure for selecting a y_j vector for a given z_i vector. When one considers all z_i vectors, this procedure cannot be extended without modifications. This is because if one repeats the above procedure for all z_i's, one may obtain the correlation of the same measurement from one sensor to several measurements of the other sensor. One therefore has to impose the constraint that each measurement can only be assigned (correlated) once while optimizing some performance index (the problem of multiple correlation due to limited sensor resolution is discussed in [54]). One such performance index is the sum of all chi-squares.[7] That is, the N correlated pairs of z_i and y_j are those achieving the minimum of

$$\sum_{i,j} w_{ij} \tag{5.8}$$

under the constraint that each z_i and y_j can only be used once.

We note that the above problem is the same as the assignment problem [8] in operations research. The optimum answer may be obtained by exhaustively searching for all combinations which results in searching through $N!$ possibilities. A procedure called the Hungarian method (or the Munkres' method for an efficient implementation procedure [56]) requiring at most $(11N^3+12N^2+31N)/6$ operations is often used in this type of problem.

In the case when the target density is not very high so that the w_{ij} of a mismatched pair may attain large values, a threshold may be first applied to examine all w_{ij}'s. For Gaussian measurement vectors, this threshold can be selected to provide a given probability of leakage. Those pairs exceeding the threshold are first rejected; one is therefore only required to correlate on a subset of measurement pairs which do not exceed the threshold. For more discussions in this problem area, see [54].

VI. Tracking in a Dense Target Environment

Tracking in a dense target environment was the subject of a recent review article [57] and numerous papers in recent IEEE-CDC Conferences (e.g., 1979–1981). Representative work in this area can be found in [57]–[64]. This problem is sometimes referred to as scan-to-scan correlation or tracking data association. In this section, we offer some very general discussion on this problem and the interested reader should consult the references for details.

This problem can be divided into two phases. The first phase is track initiation and the second phase is track maintenance. They are discussed individually below.

A. Track Initiation

Consider the case of a scanning sensor. The first and second scan produce N_1 and N_2 detections, respectively. The problem is to associate the two sets of detections to form $\min(N_1,N_2)$ number of track files. Note that we have assumed that $N_1 \neq N_2$. This can be caused by 1) imperfect detection, 2) emergence of new targets in the second scan, and 3) targets leaving the sensor field of view before the second scan. In the following, an approach for track initiation with k scans of data is described.

Let Z_k denote all the measurements (N) collected during kth scan, i.e.,

$$Z_k=\{z_1(k),z_2(k),\cdots,z_N(k)\}. \tag{6.1}$$

Let Z^k-denote the set of measurements up to and including the kth scan, i.e.,

$$Z^k=\{Z_i;\, i=1,\cdots,k\}. \tag{6.2}$$

For simplicity, we assume that N is the number of detections for all Z_k's. Assume also that sensors have perfect target detection. When this is not true, one has to enumerate more hypotheses to account for all possibilities. With Z^k, there can be N^k combinations of measurement sequences and each measurement sequence represents a possible track. Let each possible combination be denoted by a hypothesis, $H_{m_k}(k)$ which is defined by

$$H_{m_k}(k)=\{z_{n_1}(1),z_{n_2}(2),\cdots,z_{n_k}(k)\}. \tag{6.3}$$

Suppose that a tracking filter is applied to process each possible measurement sequence. The *a posteriori* hypothesis probability of $H_{m_k}(k)$ being true can be computed recursively using

$$P\big(H_{m_k}(k)/Z^k\big)=\frac{p\big(z_{n_k}(k)/H_{m_{k-1}}(k-1),Z^{k-1}\big)}{p\big(z_{n_k}(k)/Z^{k-1}\big)}\cdot P\big(H_{m_{k-1}}(k-1)/Z^{k-1}\big) \tag{6.4}$$

where $p(z_{n_k}(k)/H_{m_{k-1}}(k-1),\, Z^{k-1})$ is the probability density of the residual from the tracking filter using $H_{m_{k-1}}(k-1)$ and $z_{n_k}(k)$.[8] The above equation was shown in [59] and can also be derived as a special case of the results of [44]. The final set of tracks (total N) can be chosen as those N feasible hypotheses with the largest hypothesis probabilities, i.e.,

$$\max_{\{N;\, H_{m_k}(k)\in\mathscr{F}\}}\left\{P\big(H_{m_k}(k)/Z^k\big);\, m_k=1,\cdots,N^k\right\} \tag{6.5}$$

where the feasible set $\mathscr{F}$ is the restriction that each measurement at a given time can be used only once, i.e.,

$$\mathscr{F}=\left\{H_{m_k}(k)\colon H_{i_k}(k)\cap H_{j_k}(k)=\varnothing \text{ for } i\neq j\right\}. \tag{6.6}$$

The computational requirement of the above method is clearly nontrivial. In fact, the above optimization problem defines an *N-dimensional assignment* problem. To the best of the authors' knowledge, the N-dimensional extension of the Hungarian algorithm is not yet available. In many applications, one may be able to precluster the detections so that the search over the entire set of detections is not necessary. Other physical constraints can sometimes be imposed to reduce the search requirements depending upon given systems and application.

A similar approach using a maximum likelihood method was described in [59] in which the multidimension search problem was reduced to a 0–1 integer programming problem. Discussions on other track initiation techniques can be found in [58] and [63].

B. Track Maintenance

Once track files have been established, the computational requirement is greatly reduced. This is because for each track file one is only required to search the "admissible" region dictated by the covariance of the filter residual process.

We note that a slightly modified method of the track initiation algorithm discussed in (6.1) can be applied to the track mainte-

[7]This is corresponding to a weighted distance measure (e.g., the Mahalanobis distance [7]) between two distributions.

[8]A more parametric approach for modeling this probability density function is given in [57], [59], and [63] in which situations including *a priori* target distribution and the probability of a given number of detections were also considered.

nance problem. That is, one establishes a new hypothesis for each detection resident in the admissible region. This procedure results in an exponentially growing number of track files. One can inhibit the growing memory and computational requirement by selecting a tree depth and conducting a global search for a set of feasible tracks having the highest hypothesis probabilities (6.5), (6.6).[9] Another approach is to combine a set of "most likely hypothesis" growing out of the same track file using the weighted sum of state estimates with the hypothesis probabilities as weighting factors. This second approach is the basis of the Bayesian tracker presented by Singer *et al.* [60], [61]. If the depth is equal to one, i.e., one combines all admissible detections at each scan, then one obtains the probabilistic data association filter of Bar-Shalom and Tse [62]. We emphasize however, that the approaches of [60]–[62] are suitable for tracking in a cluttered environment (see also [64]) and do not directly address the multiple target tracking issue. This is because that tracking in a cluttered environment is a track maintenance problem while multiple target tracking must include track initiation in every new scan.

The above discussion did not include problems of track termination, imperfect sensor detection, false measurements, etc. Reid [63] provides an interesting paper which discusses extensively features of both track initiation and track maintenance.

We note that the subject of tracking in a dense target environment is of current interest. Although all quoted references present general approaches to this problem each application typically has its own unique features which impose certain restrictions resulting in a variety of modifications.

VII. Analysis Techniques

In system design and tradeoff studies involving target tracking, one is often required to evaluate the performance of trajectory estimation (tracking and prediction, etc.) for specific situations. Although Monte Carlo simulation is a frequently used method, it is nevertheless time consuming and costly. For a quick and general performance evaluation, particularly in system tradeoff studies, covariance analysis techniques are most appropriate. The following discussion treats the application of polynomial analysis, the Riccati equation and the Cramer–Rao bound.

A. Polynomial Analysis

Let p_0, v_0, and a_0 denote at $t=0$, the position, velocity, and acceleration, respectively, of a moving object projected along a given coordinate (it may be either range or angles) then its position at an arbitrary time t is

$$p(t)=p_0+v_0t+\frac{1}{2}a_0t^2. \qquad (7.1)$$

Let $\tilde{p}(t_n)$, $n=1,\cdots,N$ denote a set of noisy measurements of $p(t_n)$. For the purpose of convenience, let the time samples be taken uniformly with spacing T and let the total time interval be centered about $t=0$. Under these conditions, the minimum mean square error estimates of target position, velocity, and acceleration at an arbitrary time t are given by

$$\begin{bmatrix}\hat{p}(t)\\ \hat{v}(t)\\ \hat{a}(t)\end{bmatrix}=\Phi(t)Sy \qquad (7.2)$$

where

$$\Phi(t)=\begin{bmatrix}1 & t & t^2/2\\ 0 & 1 & t\\ 0 & 0 & 1\end{bmatrix} \qquad (7.3)$$

$$S=\begin{bmatrix}\dfrac{3(3N^2-7)}{4(N-2)N(N+2)} & 0 & \dfrac{-30}{T^2(N-2)N(N+2)}\\ 0 & \dfrac{12}{T^2(N-1)N(N+1)} & 0\\ \dfrac{-30}{T^2(N-2)N(N+2)} & 0 & \dfrac{720}{T^4(N-2)(N-1)N(N+2)(N+2)}\end{bmatrix} \qquad (7.4)$$

$$y=\begin{bmatrix}\sum\tilde{p}(t_n)\\ \sum T\left(n-1-\dfrac{(N-1)}{2}\right)\tilde{p}(t_n)\\ \sum T^2\left(n-1-\dfrac{(N-1)}{2}\right)^2\tilde{p}(t_n)/2\end{bmatrix}$$

$T=$ time between measurements. (7.5)

The covariance of the state estimate (7.2) is

$$P=\sigma^2\Phi(t)S\Phi(t)^T \qquad (7.6)$$

where σ^2 is the variance of the measurement noise.

We note that the above result, especially (7.6), is very useful because of its simplicity. It is applicable for a short track time period when the polynomial is still a close representation of the true target dynamics.

B. The Use of the Riccati Equation

The above analysis is straightforward and its calculation can be carried out using desk top (or pocket) calculators. The drawback is that it becomes overly optimistic for long track intervals $((N-1)T)$. This results since the target dynamic model (7.1) does not include process noise. To circumvent this problem, one may compute the covariance matrix using the Riccati equation (2.13). This will require a computer when nonlinear dynamics and/or measurement equations are involved. It is nevertheless a convenient method since the Monte Carlo simulation is not required.

We note that the use of Riccati equation for nonlinear problems only represents an approximate error analysis. Exactly how close is this approximation is very difficult to assess in general. When the process noise is negligible however, the solution of the Riccati equation becomes a covariance lower bound as illustrated in the next subsection.

C. The Use of the Cramer–Rao Lower Bound

There exist some situations in which the process noise is negligible. The techniques of Section VII-A are therefore applica-

[9] See [57] and [63] for a review.

ble to a limited degree, although they do ignore the coupling between coordinates and assume a linear measurement system. When the coupling between coordinates becomes significant one must use the Riccati equation to compute the covariance. When the process noise term is indeed negligible (e.g., the exoatmospheric trajectory estimation, Section V-A), the solution of the Riccati equation using Jacobian matrices evaluated along the true trajectory becomes the Cramer–Rao lower bound on the convariance of the state estimates [65], [66]. Furthermore, the Riccati equation has a closed-form solution. As an example, for the discrete case and $P_0^{-1} = 0$,[10]

$$P_N = \left[\sum_{n=1}^{N} F_n^T H_n^T R_n^{-1} H_n F_n \right]^{-1} \tag{7.7}$$

$$F_n = \prod_{n=1}^{N-1} \Phi_j^{-1} \qquad \text{for } n \leqslant N-1$$
$$= I \qquad \text{for } n = N \tag{7.8}$$

where P_N is the Cramer–Rao bound for the covariance of $\hat{x}_{N/N}$, I is an identity matrix and Φ_n and H_n are system (discrete) and measurement Jacobian matrices evaluated at x_n, respectively.

The significance of the above fact is that the actual performance is lower bounded by the solution of the Riccati equation. An extended Kalman filter generally does not perform as well while a properly designed filter can asymptotically achieve this bound; see [29].

Finally, we remark that the corresponding Cramer–Rao bound for systems with process noise is very difficult to compute. Versions of lower bounds to the Cramer–Rao bound have been proposed [66], [67]; these bounds are tight only for very large signal-to-noise ratios.

VIII. Concluding Remarks

In this paper we have presented a survey of problems and solutions which deal with target tracking. Although this problem has been of active concern to practitioners in both military and civilian applications for many years, new problems still emerge and challenge systems engineers. It is felt that the basic approach to tracking filter design is no longer an issue. However, the problem of integrating the tracking system into the overall command, control, and communication structure to achieve improved performance while minimizing data processing requirements represents the nature of problems that require current attention.

Acknowledgment

The authors have benefited from many individuals through discussions as well as coinvestigations in the area of adaptive state estimation and target tracking. They would like to specifically mention Dr. M. Athans, Dr. K. P. Dunn, Dr. M. Gruber, Dr. R. W. Miller, Dr. S. D. Weiner, Dr. R. H. Whiting, and Dr. D. Willner. Dr. Weiner has also helped to proofread the manuscript which is greatly appreciated. Patience and skillful typing of C. Tisdale were also of great help in preparing the manuscript.

References

[1] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. New York: Academic, 1970.
[2] A. Gelb, Ed., *Applied Optimal Estimation*. Cambridge, MA: M.I.T. Press, 1974.

[10]For the case when $P_0^{-1} \neq 0$, see [66].

[3] F. C. Schweppe, *Uncertain Dynamic Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1973.
[4] G. J. Bierman, *Factorization Methods for Discrete Sequential Estimation*. New York: Academic, Mar. 1977.
[5] P. S. Maybeck, *Stochastic Models, Estimation, and Control, Volume 1*. New York: Academic, 1979.
[6] E. Wong, *Stochastic Processes in Information and Dynamical Systems*. New York: McGraw-Hill, 1971.
[7] M. Kendall, *Multivariate Analysis*. New York: Hafner, 1975.
[8] F. Hillier and G. Lieberman, *Introduction to Operations Research*. Reading, MA: Addison-Wesley, 1965.
[9] P. R. Escobal, *Methods of Orbit Determination*. New York: Wiley, 2nd ed., 1976.
[10] R. P. Wishner, R. E. Larson, and M. Athans, "Status of radar tracking algorithms," in *Proc. Symp. on Nonlinear Estimation Theory and Its Appl.*, San Diego, CA, Sept. 1970, pp. 32–54.
[11] P. G. Kaminski, A. E. Bryson, Jr., and S. F. Schmidt, "Discrete square root filtering: A survey of current techniques," *IEEE Trans. Automat. Contr.*, vol. AC-16, pp. 727–736, Dec. 1971.
[12] M. Gruber, "An approach to target tracking," M.I.T. Lincoln Lab., Feb. 10, 1971, Tech. Note 1967-8.
[13] R. K. Mehra, "A comparison of several nonlinear filters for re-entry vehicle tracking," *IEEE Trans. Automat. Contr.*, vol. AC-16, pp. 307–319, Aug. 1971.
[14] C. B. Chang, R. H. Whiting, and M. Athans, "Application of adaptive filtering methods to maneuvering trajectory estimation," M.I.T. Lincoln Lab., Nov. 24, 1975, Tech. Note 1975-59.
[15] —, "On the state and parameter estimation for maneuvering re-entry vehicles," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 99–105, Feb. 1977.
[16] C. B. Chang and K. P. Dunn, "Kalman filter compensation for a special class of systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-13, pp. 700–706, Nov. 1977.
[17] M. Athans, "The role and use of the stochastic linear-quadratic-Gaussian problem in control system design," *IEEE Trans. Automat. Contr.*, vol. AC-16, pp. 529–552, Dec. 1971.
[18] R. K. Mehra, "On the identification of variances and adaptive Kalman filtering," *IEEE Trans. Automat. Contr.*, vol. AC-15, pp. 175–184, Apr. 1970.
[19] B. Carew and P. R. Belanger, "Identification of optimum filter steady-state gain for systems with unknown noise covariances," *IEEE Trans. Automat. Contr.*, vol. AC-18, pp. 582–588, Dec. 1973.
[20] R. K. Mehra, "Approaches to adaptive filtering," *IEEE Trans. Automat. Contr.*, vol. AC-17, pp. 693–698, Oct. 1972.
[21] D. T. Magill, "Optimal adaptive estimation of sampled stochastic processes," *IEEE Trans. Automat. Contr.*, vol. AC-10, Oct. 1965.
[22] L. Chin, "Advances in adaptive filtering," in *Advances in Control and Dynamic Systems*, Vol. 15, C. T. Leondes, Ed. New York: Academic, 1979.
[23] S. L. Fagin, "Recursive linear regression theory, optimal filter theory, and error analysis of optimal systems," in *IEEE Int. Conv. Rec.*, 1964, pp. 216–240.
[24] T. J. Tarn and J. Zaborszky, "A practical nondiverging filter," *AIAA J.*, vol. 8, pp. 1127–1133, June 1970.
[25] R. W. Miller, "Asymptotic behavior of the Kalman filter with exponential aging," *AIAA J.*, vol. 9, pp. 537–539, Mar. 1971.
[26] T. R. Benedict and G. W. Bordner, "Synthesis of optimal set of radar track-while-scan-smoothing equations," *IRE Trans. Automat. Contr.*, vol. AC-7, pp. 27–32, July 1962.
[27] S. R. Neal, "Discussion on parametric relations for α-β-γ filter predictor," *IEEE Trans. Automat. Contr.*, vol. AC-12, pp. 315–317, June 1962.
[28] R. Fitzgerald, "Design aids for simple filters," unpublished notes, Raytheon Company, Bedford, MA, 1976.
[29] C. B. Chang, "Ballistic trajectory estimation with angle-only measurements," *IEEE Trans. Automat. Contr.*, vol. AC-25, pp. 474–480, June 1980.
[30] T. S. Lee, K. P. Dunn, and C. B. Chang, "On observability and unbiased estimation of nonlinear systems," in *Proceedings of 10th IFIP Conference on Systems Modeling and Optimization*. New York: Springer-Verlag, 1982.
[31] R. Sridharan and W. P. Seniw, "ANODE: An analytic orbit determination system," M.I.T. Lincoln Lab., June 1980, Tech. Note 1980-1.
[32] M. Athans, R. Wishner, and A. Bertolini, "Suboptimal state estimation for continuous time nonlinear systems from discrete noisy measurements," *IEEE Trans. Automat. Contr.*, vol. AC-13, pp. 504–514, Oct. 1968.
[33] R. P. Wishner, J. A. Tabaczynski, and M. Athans, "A comparison of three nonlinear filters," *Automatica*, vol. 5, pp. 487–496, 1969.
[34] D. L. Alspack and H. W. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Trans. Automat. Contr.*, vol. AC-17, pp. 439–448, Aug. 1972.
[35] L. Schwartz and E. B. Stear, "A computational comparison of

several nonlinear filters," *IEEE Trans. Automat. Contr.*, vol. AC-13, pp. 83–86, Jan. 1968.

[36] A. H. Jazwinski, "Adaptive filtering," *Automatica*, vol. 5, pp. 475–485, 1969.

[37] P. S. Maybeck, R. L. Jensen, and D. A. Harnly, "An adaptive extended Kalman filter for target image tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-17, pp. 172–180, Mar. 1981.

[38] R. A. Singer, "Estimating optimal tracking filter performance for manned-maneuvering targets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-6, pp. 473–483, July 1970.

[39] B. Friedland, "Treatment of bias in recursive filtering," *IEEE Trans. Automat. Contr.*, vol. AC-14, pp. 359–367, Aug. 1969.

[40] C. B. Chang and K. P. Dunn, "On GLR detection and estimation of unexpected inputs in linear discrete systems," *IEEE Trans. Automat. Contr.*, vol. AC-24, pp. 499–501, June 1979.

[41] R. J. McAulay and E. Denlinger, "A decision-directed adaptive tracker," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-9, pp. 229–236, Mar. 1973.

[42] A. S. Willsky and H. L. Jones, "A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems," *IEEE Trans. Automat. Contr.*, vol. AC-21, pp. 108–112, Feb. 1976.

[43] A. S. Willsky, "A survey of design methods for failure detection in dynamic systems," *Automatica*, vol. 12, pp. 601–611, 1976.

[44] C. B. Chang and M. Athans, "State estimation for discrete systems with switching parameters," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-14, pp. 418–425, May 1978.

[45] M. Athans and C. B. Chang, "Adaptive estimation and parameter identification using multiple model estimation algorithms," M.I.T. Lincoln Lab., June 23, 1976, Tech. Note 1976–28.

[46] D. G. Lainiotis, "Optimal adaptive estimation: structure, and parameter adaptation," *IEEE Trans. Automat. Contr.*, vol. AC-16, Apr. 1971.

[47] —, "Partitioning: A unifying framework for adaptive systems, I estimation," *Proc. IEEE*, vol. 64, Aug. 1976.

[48] M. Athans *et al.* "The stochastic control of the F-8C aircraft using a multiple model adaptive control (MMAC) method—Part I: Equilibrium flight," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 768–779, Oct. 1977.

[49] R. L. Moose, H. F. Van Landingham, and D. H. McCabe, "Modeling and estimation for tracking maneuvering targets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-15, pp. 448–456, May 1979.

[50] N. H. Gholson and R. L. Moose, "Maneuvering target tracking using adaptive state estimation," *IEEE Trans. Aerosp. Electron. Syst.*, pp. 310–317, May 1977.

[51] R. R. Tenney, R. S. Hebbert, and N. R. Sandell, Jr., "A tracking filter for maneuvering sources," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 246–251, Apr. 1977.

[52] P. S. Maybeck and D. E. Mercier, "A target tracker using spatially distributed infrared measurements," *IEEE Trans. Automat. Contr.*, vol. AC-25, pp. 222–225, Apr. 1980.

[53] D. Willner, C. B. Chang, and K. P. Dunn, "Kalman filter algorithms for a multi-sensor system," in *Proc. 1976 IEEE Conf. on Decision and Contr.*, Clearwater, FL, Dec. 1976.

[54] C. B. Chang and L. C. Youens, "Measurement correlation for multiple sensor tracking in a dense target environment," M.I.T. Lincoln Lab., Jan. 1981, Tech. Rep. 549.

[55] C. B. Chang and L. C. Youens, "Measurement correlation for multiple sensor tracking in a dense target environment," *IEEE Trans. Automat. Contr.*, vol. AC-27, pp. 1250–1252, Dec. 1982.

[56] J. Munkres, "Algorithms for the assignment and transportation problems," *SIAM J.*, vol. 5, pp. 32–38, Mar. 1957.

[57] Y. Bar-Shalom, "Tracking methods in a multitarget environment," *IEEE Trans. Automat. Contr.*, vol. 23, pp. 618–626, Aug. 1978.

[58] R. W. Settler, "An optimal data association problem in surveillance theory," *IEEE Trans. Military Electron.*, vol. MIL-8, pp. 125–139, Apr. 1964.

[59] C. L. Morefield, "Application of 0-1 integer programming to multitarget tracking problems," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 302–312, June 1977.

[60] R. A. Singer and R. G. Sea, "New results in optimizing surveillance system tracking and data correlation performance in dense multitarget environments," *IEEE Trans. Automat. Contr.*, vol. AC-18, pp. 571–581, Dec. 1973.

[61] R. A. Singer, R. G. Sea, and K. Housewright, "Derivation and evaluations of improved tracking filters for use in dense multitarget environment," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 423–432, July 1974.

[62] Y. Bar-Shalom and E. Tse, "Tracking in a cluttered environment with probabilistic data association," *Automatica*, vol. 11, pp. 451–460, Sept. 1975.

[63] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Automat. Contr.*, vol. AC-24, pp. 843–854, Dec. 1979.

[64] M. Athans, R. H. Whiting, and M. Gruber, "A suboptimal estimation algorithm with probabilistic editing for false measurements with application to target tracking with wake phenomena," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 372–384, June 1977.

[65] J. H. Taylor, "The Cramer-Rao estimation error lower bound computation for deterministic nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. AC-24, pp. 343–344, Apr. 1979.

[66] C. B. Chang, "Tow covariance lower bounds on the covariance for nonlinear estimation problems," *IEEE Trans. Automat. Contr.*, vol. AC-26, Dec. 1981.

[67] B. Z. Bobrovsky and M. Zakai, "A lower bound on the estimation error for Markov processes," *IEEE Trans. Automat. Contr.*, vol. AC-20, pp. 785–788, Dec. 1975.

Chaw-Bing Chang (S'68–M'71) was born in Szechuan, China, on May 19, 1944. He received the B.S. degree in electrical engineering from Cheng Kung University, Taiwan, China, in 1966 and the M.S. degree in electrical engineering and the Ph.D. degree in control and communication systems from the State University of New York at Buffalo in 1969 and 1971, respectively.

From 1971 to 1974 he was with the Calspan Corporation (formerly Cornell Aeronautical Laboratory), Buffalo, NY, working in the areas of detection, estimation, and radar signal processing. Since 1974 he has been a member of the Technical Staff at M.I.T. Lincoln Laboratory, Lexington, MA. His present interests are in the areas of optimal estimation, detection, and control with applications to strategic defense systems.

Dr. Chang is a member of Sigma Xi.

John A. Tabaczynski (M'65) was born in Rochester, NY, on February 19, 1939. He received the S.B. degree in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1960, and the M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, in 1962 and 1965, respectively.

In 1965 he joined the Jet Propulsion Laboratory, Pasadena, CA, where he was a member of the Radar System Group. Since 1966 he has been a Staff Member at M.I.T. Lincoln Laboratory, Lexington, MA. In 1973 he was promoted to Assistant Group Leader of the Analysis and Systems Group and in 1974 he assumed the Associate Group Leader position of the same group. Since 1978 he has been the Group Leader of Analysis and Systems. His technical interests lie in the application of estimation and control theory, signal processing, and optimization techniques to large scale systems.